GENERALIZED TEXTURE MODELS FOR DETECTING HIGH-LEVEL STRUCTURES IN REMOTELY SENSED IMAGES

A THESIS
SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Emel Doğrusöz
June, 2007

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Selim Aksoy(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Pınar Duygulu Şahin

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. M. Volkan Atalay

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet B. Baray
Director of the Institute

ABSTRACT

GENERALIZED TEXTURE MODELS FOR DETECTING HIGH-LEVEL STRUCTURES IN REMOTELY SENSED IMAGES

Emel Doğrusöz
M.S. in Computer Engineering
Supervisor: Asst. Prof. Dr. Selim Aksoy
June, 2007

With the rapid increase in the amount and resolution of remotely sensed image data, automatic extraction and classification of information obtained from such images have been an important problem in the field of pattern recognition since remotely sensed imagery is a critical resource for diverse fields such as urban land use monitoring and management, GIS and mapping, environmental change and agricultural and ecological studies. This thesis proposes statistical and structural texture models for detecting high-level structures in remotely sensed images. The high-level structures correspond to complex geospatial objects with characteristic spatial layouts in a region. As opposed to the existing approaches that are based on classifying images using pixel level methods, we propose to use simple geospatial objects as textural primitives and exploit their spatial patterns. This representation can be viewed as a "generalized texture" measure where the image elements of interest are urban primitives instead of the traditional case of pixels. The spatial patterns we are interested in correspond to the regular and irregular arrangements of these primitives within neighborhoods.

The methodology we propose in this thesis has two steps. First, the primitives of interest are detected using spectral, textural and morphological features with one-class classifiers. Then, the spatial patterns of these primitives are modeled. At this step, either a statistical or a structural approach can be followed. In the statistical approach, analysis of the spatial arrangement of the primitives is done by co-occurrence-based spatial domain features and Fourier spectrum-based frequency domain features. These features are used to quantify the likelihood of presence of the focused object in the image region being analyzed. In the structural approach, a graph-theoretic representation is proposed where the primitives form the nodes of a graph and the neighborhood information is obtained through

Voronoi tessellation of the image scene. Next, the graph is clustered by thresholding its minimum spanning tree and the resulting clusters are classified as regular or irregular by examining the distributions of the angles between neighboring nodes.

The algorithms proposed in this thesis are illustrated with the detection of two geospatial objects: settlement areas and harbors. The first step in the modeling of these objects is the detection of primitives such as buildings for settlement areas, and boats and water for harbors. In the second step, both statistical and structural approaches are illustrated for the modeling of the spatial patterns of these objects. Results of the experiments on high-resolution Ikonos satellite imagery and DOQQ aerial imagery show that the proposed techniques can be used for detecting the presence of geospatial objects in large remote sensing image datasets.

Keywords: Pattern recognition, one-class classification, geospatial object detection, co-occurrence texture analysis, Fourier texture analysis, graph-based texture analysis.

ÖZET

UZAKTAN ALGILANAN RESİMLERDE ÜST DÜZEY YAPILARI BULMAK İÇİN GENEL DOKU MODELLERİ

Emel Doğrusöz
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Yard. Doç. Dr. Selim Aksoy
Haziran, 2007

Kentsel alan kullanımının gözetimi ve yönetimi, GIS (coğrafi bilgi sistemleri), kartografi, çevresel, zirai ve ekolojik çalısmalar için önemli bir kaynak olan uydu görüntülerinden otomatik olarak bilgi çıkarımı ve değerlendirilmesi, bu görüntülerin miktarının her geçen gün hızla artması ile örüntü tanıma alanı için popüler bir problem haline gelmiştir. Bu tez, uzaktan algılanan görüntülerden karmaşık yapıların sezimlemesini sağlamak amacıyla istatistiksel ve yapısal doku modelleri önermektedir. Uydu görüntülerindeki karmaşık yapılar, bulunduğu bölge içerisinde karakteristik uzamsal düzene sahip kompleks coğrafi nesnelere denk gelmektedir. Literatürde, görüntüleri piksel düzeyindeki yöntemlerle sınıflandıran yaklaşımların tersine, biz bu çalışmada, basit coğrafi nesneleri dokunun temel birimi olarak kullanmakta ve bu temel öğelerin uzamsal örüntülerini modellemekteyiz. Bu yaklaşım, ilgilenilen görüntü elemanlarının geleneksel yaklaşımdaki pikseller yerine kentsel temel öğelere dönüştüğü "genellenmiş doku" ölçütü olarak görülebilir. İlgilendiğimiz uzamsal örüntüler, bu temel öğelerin komşuluklar içerisindeki düzenli veya düzensiz şeklindeki dağılımlarıdır.

Bu tezde sunduğumuz yöntem iki basamaktan oluşmaktadır. İlk basamakta, temel öğeler spektral, dokusal ve morfolojik özniteliklerin tek-sınıflı sınıflandırıcılarda kullanılmasıyla bulunmaktadır. İkinci basamakta ise bu temel öğelerin uzamsal örüntüleri modellenmektedir. Bu basamakta, istatistiksel ya da yapısal bir yaklaşım izlenebilir. İstatistiksel yaklaşımda, temel öğelerin uzamsal dağılımı analizi eş oluşuma dayalı uzamsal bölge öznitelikleri ve Fourier spektrum tabanlı frekans bölgesi öznitelikleri ile yapılmaktadır. Bahsedilen öznitelikler, odaklanılan nesnenin, analizi yapılan görüntü bölgesinde var olma olasılığını nicelemek için kullanılmaktadır. Yapısal yaklaşımda ise, herbir düğümün bir temel öğeye karşılık geldiği, ve komşuluk bilgisinin görüntünün Voronoi diyagramı sayesinde bulunduğu çizge tabanlı bir gösterim sunulmuştur. Bir çizge, minimum kapsayan

ağacının belli bir eşik değere göre parçalanması yolu ile kümelere bölünmekte ve kümeler, içlerindeki düğümlerin komşularıyla yaptığı açı dağılımları dikkate alınarak düzenli/düzensiz şeklinde sınıflandırılmaktadır.

Bu tezde sunulan algoritmalar, yerleşim alanları ve liman olmak üzere iki çeşit kompleks coğrafi nesnenin bulunması problemine uygulanmıştır. Bu nesnelerin modellenmesindeki ilk adım, yerleşim alanları için bina, liman için ise gemi ve su gibi temel öğelerin bulunmasıdır. İkinci adımda ise, bu nesnelerin uzamsal örüntülerini modellemek üzere istatistiksel ve yapısal yaklaşımlar örneklendirilmiştir. Yüksek çözünürlüklü İkonos ve DOQQ görüntüleri üzerinde yapılan deney sonuçları, önerilen tekniklerin büyük uydu görüntüsü veri kümelerinde coğrafi nesnelerin varlığını sorgulamak amacıyla kullanılabileceğini göstermiştir.

Anahtar sözcükler: Örüntü algılama, tek sınıflı sınıflandırma, coğrafi nesne sezimi, eş oluşuma dayalı doku analizi, Fourier tabanlı doku analizi, çizge tabanlı doku analizi.

Acknowledgement

I would like to express my deep thanks to my advisor Selim Aksoy for his guidance and support throughout this work. It has been a valuable experience for me to work with him and get benefit from his vision and knowledge in every step of my research.

I am also very thankful to Pınar Duygulu Şahin and M. Volkan Atalay, for their suggestions on improving this work.

Besides, I would like to express my pleasure on being part of the RETINA team, and having such a nice friendship with the group members. The two years with them in EA 522 will be a very pleasant memory for me.

Certainly, I appreciate my whole family for their endless support and caring. Uğur deserves the very big part of the credit by his valuable comments, support and love. I would like to thank him for always being with me.

Finally, I would like to thank to TUBITAK, Dr. Sitaram Bhagavathy and Dr. B. S. Manjunath from the University of California, Santa Barbara, for providing the datasets used in this thesis. 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.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Remotely sensed imagery of the Earth is a critical resource for diverse fields such as geography, urban planning, cartography and monitoring applications. Spatial technology may provide great information for these fields by analyzing such images. However, for spatial technology to be effective, it has to be done automatically and it should provide reliable results when applied to a large extent.

The amount of satellite images has increased with the advances in sensing and storage technology. This resulted in the availability of images covering a large surface area on the earth [9]. For example, nearly 3 terabytes of data are being sent to Earth by NASA's satellites every day [22]. Advances in satellite technology and computing power have enabled the study of multi-modal, multi-spectral, multi-resolution and multi-temporal data sets for applications such as urban land use monitoring and management, GIS and mapping, environmental change, site suitability, agricultural and ecological studies. Automatic content extraction, classification and content-based retrieval have become highly desired goals for developing intelligent systems for effective and efficient processing of remotely sensed data sets [3].

This thesis focuses on detecting high-level structures in remotely sensed images. The high-level structures correspond to complex geospatial objects with characteristic spatial layouts in a region. The detection of geospatial objects with simple geometric or shape models such as buildings [46, 32, 33, 52] and roads [25, 26, 6] has been explored extensively in the literature. However this is not the case for complex geospatial objects such as buildings with their arrangement information in an urban setting (or an harbor with a special alignment of rows of boats). Therefore new flexible techniques are required to detect such objects and to analyze and model their structures. This task requires the use of position, scale and rotation invariant modeling techniques and this can be achieved via the use of high level structures and modeling. In this thesis, we propose such an approach which distinguishes the regular (organized) geospatial structures from the irregular (unorganized) ones. We define the basic geospatial elements of images as *primitives* (e.g. individual buildings, roads, trees, boats). We consider these object primitives as *texels* (texture elements) as opposed to traditional texture work which considers pixels as the unit elements. With object primitives as the unit elements, we analyze the relationships and properties of these primitives. This representation can be viewed as a "generalized texture" measure where the image elements of interest are urban primitives instead of the traditional case of pixels. Again in analogy with the traditional texture work which characterizes texture with either a statistical or structural method, the approaches we bring to characterize and measure the extent of regularity/irregularity of a geospatial scene can be mainly classified as statistical and structural. This approach, representing the problem as a generalized texture problem, has diverse applications such as complex geospatial object detection and urbanization modeling. In this thesis, we illustrate the use of the approach on harbor detection (see Figure 1.1) and on classifying urban areas as having regular settlement patterns and irregular settlement patterns that represent highly organized and unorganized neighborhoods, respectively (see Figure 1.2).

In the following section we discuss some of the previous work on geospatial object detection and urbanization modeling.

Figure 1.1: Example harbor object.

(a) Regular (highly organized) (b) Irregular (unorganized)

Figure 1.2: Examples of building patterns.

1.2 Related Work

1.2.1 Pixel-Level Analysis

There is an extensive literature on pixel-level analysis of remotely sensed imagery [38]. However, a recent study [72] showed that there has not been any significant improvement in the performance of methodologies over the last 15 years. The main reason is that the use of only pixel level data often does not meet the expectations as the resolution increases. Even though high success rates have been published in the literature using limited ground truth data, visual inspection of the results can show that most of the urban structures still cannot be delineated as accurately as expected [1].

Pixel-based approaches assume that similar land structures will cluster together and behave similarly in terms of pixel level features. However, the assumptions for distribution models often do not hold for high-resolution data. Even though such approaches can be successful for the detection of geospatial objects with simple shape and geometry such as roads, they fail for applications that require not only detecting geospatial objects but also analyzing the relationship of those objects. Since pixel level classification cannot capture the spatial information inherent among objects of the image, methods based on pixel level analysis are incapable of interpreting the land cover with neighboring information.

The generic techniques that can be used for modeling and detection of different types of geospatial objects require the use of spatial distribution and neighborhood information. Traditionally, this is done by the use of textural or edge-based methods. Since pixel level approaches cannot capture neighborhood information, in the rest of the thesis we focus only on techniques analyzing spatial distribution.

1.2.2 Texture-based Methods

Texture is one of the most important characteristics of images used for analyzing them. It can be characterized by textural primitives as unit elements and neighborhoods in which the organization and relationships between the properties of these primitives are defined [28]. Numerous methods, that were designed for a particular application, have been proposed in the literature. However, there seems to be no general method or a formal approach which is useful in a broad range of images [2].

Haralick and Shapiro [30] defined texture as the uniformity, density, coarseness, roughness, regularity, intensity and directionality of discrete tonal features and their spatial relationships. Although no generally applicable definition of texture exists, some common elements in the definitions found in the literature are primitives and/or properties that are defined in a neighborhood and the statistical and/or structural relationships between these primitives and/or properties that are measured at a scale of interest. In his texture survey [28], Haralick gave two kinds of approaches to characterize and measure texture: statistical approaches like autocorrelation functions, optical transforms, digital transforms, textural edgeness, structuring elements, spatial gray level run lengths and autoregressive models, and structural approaches that use the idea that textures are made up of primitives appearing in a near-regular repetitive arrangement.

In the literature, there exists a variety of methods using image texture in geospatial image analysis but they mostly focus on the classification of certain types of land cover such as terrain types and crops [71, 13, 8, 69]. Textural features have also been used to model spatial information in neighborhoods of pixels since textural statistics such as Grey-Level Co-occurrence Matrix (GLCM) [29] or Markov Random Fields (MRF) [20] consider the spatial distribution and neighborhood information within a moving window.

A GLCM tabulates the frequencies with which different gray levels occur in a certain spatial configuration (usually defined by distance and direction). The co-occurrence-based textural features such as contrast, homogeneity, entropy have been commonly used for remote sensing applications. For example in [14], co-occurrence matrices were used in the study of land cover change detection in moderate resolution imaging spectroradiometer (MODIS) imagery. More recently, Karathanassi *et al.* [35] used spectral thresholding to detect buildings and computed co-occurrence frequencies of adjacent pixels within rectangular neighborhoods to classify images into areas with low, medium and high density of buildings. A similar work was in [18] where Dell'Acqua *et al.* used co-occurrence texture measures to improve the pixel-by-pixel classification of an urban area to provide information on different building densities inside a town structure.

Another framework for texture analysis is the MRF which treats an image as a realization of a two-dimensional lattice of random variables by the Markovian assumption. In [56], MRFs were used for the labeling of hyperspectral AVIRIS image regions as urban or non-urban. In [20], Descombes *et al.* used Gaussian Markov random fields and proposed two methods to estimate textural parameters in remote sensing images. With the first estimation method, urban areas were extracted from SPOT images and the second method is applied to segment ice in polar regions from AVHRR data.

A different approach was presented in [7] and [15] to classify high-resolution remotely sensed images from urban areas by the use of mathematical morphology. Morphological features were used in the preprocessing of the neural network-based classification in the first, whereas in the latter the derivative of morphological profile (DMP) features were used as the feature vector on which the classification is based.

In the literature, frequency domain knowledge is commonly used for structural analysis of texture. Since frequency domain analysis can give information about the regular or periodic image patterns, Fourier power spectrum and wavelet-based texture analysis are ways of analyzing structural patterns in texture. In [69], Fourier spectrum based texture features were used to classify different types of vineyards in high-resolution aerial photographs. In [40], Daubechies wavelet family [17] is used to classify Landsat image regions as mountainous or flat by computing the standard deviation of the wavelet coefficients in local windows. To detect geospatial objects, Bhagavathy [9] used Gabor filters tuned to different frequencies as texture features, and performed Gaussian mixture-based clustering of pixels as texture elements. Histograms of these elements within sub-windows were used for detection of golf courses and harbors. Even though texture features are sensitive to different pixel neighborhoods, histograms ignore spatial arrangements within a window, and cannot capture the actual placement patterns of texture elements.

1.2.3 Edge-based Methods

These methods usually interpret a given scene using some amount of contextual knowledge about a scene (e.g. airport, housing development) [9]. These methods usually divide an image into spatial units (closed regions, lines, etc.) through image segmentation or edge detection/ linking. Spatial relations between units are analyzed using relational models such as production systems [49], semantic networks [51, 55], human-specified constraints or rules [47], and evidential reasoning [45]. These frameworks are essentially used for humans to specify spatial constraints among the constituents of a scene or object.

For example in [68], Unsalan and Boyer used edge information to directly model image windows for urbanization. They extracted line segments in grayscale images, constructed graphs to model the relationships of these lines, and introduced a set of measures based on various properties of these graphs to classify images as rural, residential or urban without explicitly detecting any objects such as buildings. In [32], Huertas and Nevatia used lines, corners and shadow analysis to detect buildings using the rectangularity of buildings as a simplifying constraint.

Similarly, for road detection from aerial imagery, Laptev *et al.* used a strategy mainly based on the multi-scale detection of roads in combination with geometry-constrained edge extraction using snakes in [39]. In [59], a model was presented for the extraction of road networks from images in the presence of occlusions, where high-order active contours were used with incorporating sophisticated geometric information to close the gaps between extracted networks.

1.3 Summary of Contributions

The goal of this thesis is to analyze remote sensing images and detect complex geospatial structures in terms of the regularity and irregularity of simpler primitives. This detection can give useful information for diverse application fields. For example, differentiating regularly structured buildings from the irregular ones can be used in urbanization studies to measure the degree of urbanization in a settlement area. Likewise, the approach can be used for detecting complex geospatial objects that consist of simpler image structures and where the alignment of these structures is important in the detection. Most of the complex geospatial object types (unlike roads or buildings) necessitate the use of spatial alignment information for detection, and this problem has not received much attention yet. In this work, the approach we have for the distinction of regular vs. irregular structures is to see the problem as a high-level texture analysis problem. As explored in Section 1.2.2, texture analysis can be performed using statistical or structural approaches. In this work we propose a statistical and a structural method to analyze the regularities vs. irregularities in images.

The statistical method we propose involves detection of individual object primitives such as buildings and boats using texture features, multispectral information and morphological characteristics. Then we perform the texture-based modeling of these primitives' spatial arrangements within image scenes. The spatial information we are interested in corresponds to the primitives' repetitiveness and periodicity at particular orientations, and we achieve this task by the use of co-occurrence-based spatial domain features and Fourier spectrum-based frequency domain features. Note that we are interested in the spatial arrangement of the texels (texture elements) as opposed to most of the literature work (for example, [18], [35]) that consider the spatial information in pixel level.

Seen from this aspect, the work by Bhagavathy [9] is the one most related to our work in terms of the level of the primitives used, and the general goal of model-driven detection of the spatial arrangements of geospatial objects. However different than that work, we use second order measures (co-occurrence-based spatial domain features and Fourier spectrum-based frequency domain features) to analyze the spatial arrangements of object primitives within a window as opposed to their first order approach (histograms of texture elements within a window) which ignores the spatial arrangements of primitives. In this work, we demonstrate the use of our model on two applications. In the first one, we examine the spatial arrangements of buildings and detect the regular and irregular patterns that represent highly organized and unorganized neighborhoods, respectively. With such an approach, it is possible to detect whether a settlement area has undergone planned land development or it is an informal settlement that has been affected by illegal expansion. The second application of the method is on complex object detection, and the tests are done to detect harbors in large images. In this application, the first phase of the method finds boats in the image (with possible false alarms), and the second phase of the method eliminates false alarms and finds almost exact regions of the harbors by utilizing the spatial arrangements of boats in harbors.

The structural technique we propose for modeling high-level geospatial objects and their neighborhoods in an urban setting starts with the detection of urban primitives (e.g., buildings, trees, roads, etc.). Then, the neighborhoods are modeled in terms of the spatial arrangements of these primitives. This is achieved using a graph-based model that clusters the primitives into groups that are composed of primitives with similar spatial arrangements. In this representation, Voronoi tessellation of a graph is used to determine the neighborhood information of primitives. The grouping phase can be considered as a structural pattern recognition problem that uses graph-based representations and clustering techniques. We illustrate the proposed approach in the problem of measuring the level of urbanization according to spatial building patterns where the graph nodes correspond to individual buildings, and the clusters of the graph correspond to building groups with similar arrangements. The spatial arrangements we are interested in correspond to regular patterns and irregular patterns that represent highly organized and unorganized neighborhoods, respectively as shown in Figure 1.2. The former represents urban areas that undergo planned land development whereas the latter corresponds to areas that are affected by illegal expansion mostly due to immigration.

In this work, we aim to perform the specified tasks using minimum training, and this is why we detect object primitives and operate on them. This approach provides us to just focus on the primitives of interest and use their specific arrangement information in detecting the complex geospatial objects.

1.4 Organization of the Thesis

In Chapter 2, we present the features extracted for this work and the dataset used in the experiments. Mainly, we used Gabor texture features, morphological characteristics and multispectral data of images. In the experiments, we used two datasets; one for urbanization application and one for the detection of harbors. Chapter 3 explains how we detect primitives, such as buildings and boats. In Chapter 4, we present the statistical approach for differentiating regular structures from the irregular ones. For this purpose, we mainly propose using two types of features, one from spatial domain and one from frequency domain. In Chapter 5, the structural model is explained. In this chapter, we present a graph-theoretic approach and show how it can be used for modeling urbanization. In Chapter 6, experiments and their results are given. Finally, in Chapter 7, we summarize the work and conclude with future research directions.

Chapter 2

Feature Extraction and Dataset

2.1 Overview

In this work, to extract the object primitives from images, we used information from different domains. Namely, we used Gabor texture features from the frequency domain, morphological profile features from the structural domain and RGB features from the spectral domain. Details of these features are given below.

2.2 Gabor Texture Features

We use Gabor texture features to take advantage of the frequency domain analysis. As stated in [10], frequency domain texture analysis is generally computationally less expensive than image segmentation and edge detection/linking, especially for large and highly detailed geospatial images. Furthermore, texture analysis using a Gabor filter bank provides a compact description of visual structure present in a neighborhood and has a high potential of describing high-level structures. Figure 2.1 illustrates the use of Gabor filters in localizing the patterns of objects without having to perform image segmentation or edge detection/linking.

(a) (b) (c)

Figure 2.1: (a) An instance of an harbor object, with the white line indicating the extent. The ground resolution of the image is 1m/pixel. When it is convolved with a Gabor filter bank, strong responses are observed inside the harbor at two scales. (b) Output of a filter with an orientation of 0° and scale corresponding to a filter period of 9.3 pixels. This corresponds to an approximate separation of 9.3m between individual boats. (c) Output of a filter with an orientation of 90° and scale corresponding to a filter period of 37.8 pixels. This corresponds to an approximate separation of 37.8m between rows of boats (images taken from [9].)

To analyze texture in frequency domain, it is a common practice to apply a bank of scale and orientation selective Gabor filters constructed as in [44], to an image. A 2D Gabor function *g(x,y)* and its Fourier transform *G(u,v)* can be expressed as:

$$\begin{aligned} g(x,y) &= \left(\frac{1}{2\pi\sigma_x\sigma_y}\right)\exp\left[\frac{-1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)+2\pi jW_x\right] \\ G(u,v) &= \exp\left\{\frac{-1}{2}\left[\frac{(u-W)^2}{\sigma_u^2}+\frac{v^2}{\sigma_v^2}\right]\right\} \end{aligned} \tag{2.1}$$

where $\sigma_u = 1/2\pi\sigma_x$ and $\sigma_v = 1/2\pi\sigma_y$. Considering *g(x,y)* as the mother wavelet of the Gabor wavelets, a Gabor filter bank can be derived by dilations and translations of *g(x,y)* through the generating function:

$$\begin{aligned} g_{s,k}(x,y) &= a^{-s}g(x',y'), a>1 \\ x' &= a^{-s}(x\cos\theta + y\sin\theta) \\ y' &= a^{-s}(-x\sin\theta + y\cos\theta) \end{aligned} \tag{2.2}$$

where $s \in \{0,...,S-1\}$, $k \in \{0,...,K-1\}$ and $\theta = \frac{k\pi}{K}$ is the orientation of the filter w.r.t. the vertical axis. The indices k and s indicate the orientation and scale of the filter, respectively. K denotes the total number of orientations and S is the total number of scales in the filter bank. Given the input specifications S, K, and the upper and lower center frequencies, U_h and U_l, the filter bank parameters $\{\sigma_x, \sigma_y, a, W\}$ are computed by the method described in [44]:

$$\begin{aligned}
a &= \left(\frac{U_h}{U_l}\right)^{\left(\frac{-1}{S-1}\right)} \\
W &= U_h \\
\sigma_u &= \frac{(a-1)U_h}{(a+1)\sqrt{(2\ln 2)}} \\
\sigma_v &= \tan\frac{\pi}{2k}\left[U_h - 2\ln\left(\frac{\sigma_u^2}{U_h}\right)\right]\left[2\ln 2 - \frac{(2\ln 2)^2\sigma_u^2}{U_h^2}\right]^{\frac{-1}{2}}.
\end{aligned} \tag{2.3}$$

Gabor filter-based textural features of a pixel is derived as an $S \times K$-dimensional feature vector obtained by convolving an image window with a Gabor filter bank with S scales and K orientations. Let $c(\mathbf{x})$ denote the feature vector extracted from the neighborhood of pixel $\mathbf{x} = [x\ y]^T$. This feature vector is given by

$$c(\mathbf{x}) = [F_{0,0}(\mathbf{x})F_{0,1}(\mathbf{x})...F_{s-1,k-2}(\mathbf{x})F_{s-1,k-1}(\mathbf{x})]^T \tag{2.4}$$

where $F_{s,k}(\mathbf{x})$ is the filter output at pixel $\mathbf{x}$, obtained by convolving the image $I(\mathbf{x})$ with the filter $g_{s,k}(\mathbf{x})$. In other words, $F_{s,k}(\mathbf{x}) = |g_{s,k}(\mathbf{x}) * I(\mathbf{x})|$.

In our experiments, we take $S = 5$, $K = 6$, $U_l = 0.05$ and $U_h = 0.4$ where filter kernel size is 75 pixels as suggested in [9]. Figure 2.2 illustrates Gabor texture filters at different scales and orientations.

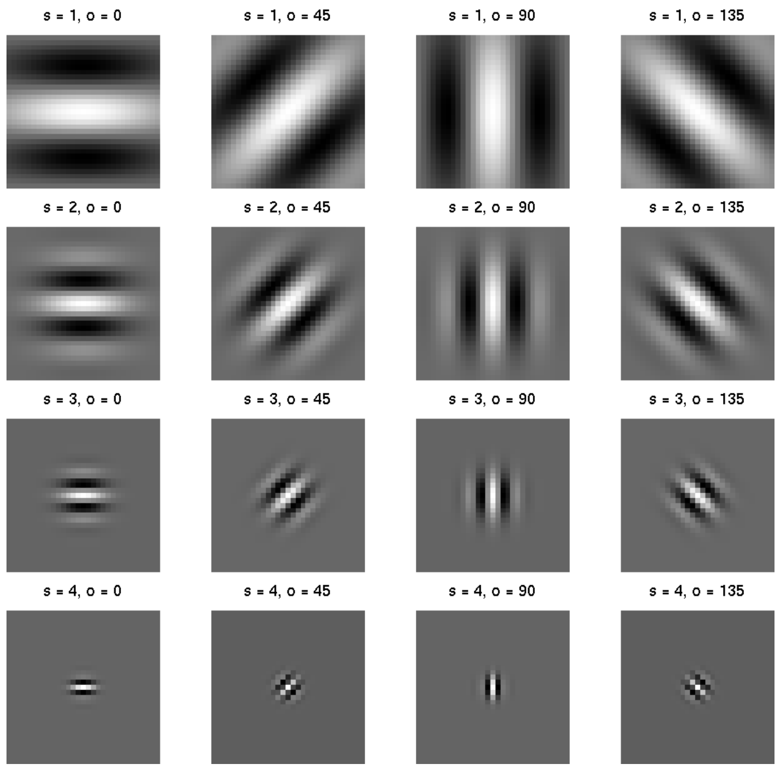


Figure 2.2: Gabor texture filters at different scales ($s = 1, \ldots, 4$) and orientations ($o = 0°, 45°, 90°, 135°$). Each filter is approximated using 31×31 pixels (image taken from [3].)

2.3 Morphological Profile Features

In the literature, morphological operators are widely used to model structural characteristics of pixel neighborhoods. In [53], morphological operators with different structuring element (SE) sizes were applied to obtain a multi-scale representation of structural information. In [53], morphological profiles (MP) for all pixels of an image are generated by successively applying opening and closing operations with increasing structuring element sizes. Furthermore, the derivative of the morphological profile (DMP) is defined as a vector where the measure of the slope of the opening-closing profile is stored for every step of an increasing SE series. A review of the concepts of the morphological profile and of the derivative of the morphological profile as defined by Pesaresi and Benediktsson [53] is given below.

Let γ_λ^* be a morphological opening by reconstruction operator using structuring element $SE = \lambda$ and $\Pi_\gamma(x)$ be the opening profile at the pixel x of the image I. $\Pi_\gamma(x)$ is defined as a vector

$$\Pi_\gamma(x) = \{\Pi_{\gamma_\lambda} : \Pi_{\gamma_\lambda} = \gamma_\lambda^*(x), \forall\lambda \in [0, n]\}. \tag{2.5}$$

Also, let φ_λ^* be a morphological closing by reconstruction operator using structuring element $SE = \lambda$ and $\Pi_\varphi(x)$ be the closing profile at the pixel x of the image I. $\Pi_\varphi(x)$ is defined as a vector

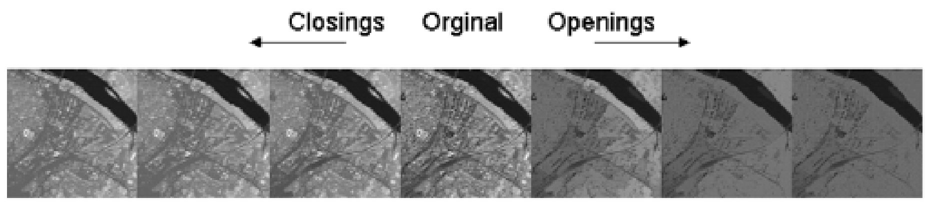
$$\Pi_\varphi(x) = \{\Pi_{\varphi_\lambda} : \Pi_{\varphi_\lambda} = \varphi_\lambda^*(x), \forall\lambda \in [0, n]\}. \tag{2.6}$$

In the above definitions, the opening and closing by reconstruction operations imply $\Pi_{\gamma_0}(x) = \Pi_{\varphi_0}(x) = I(x)$. Then, the derivative of the morphological profile is defined as a vector where the measure of the slope of the opening-closing profile is stored for every step of an increasing SE series. The derivative of the opening profile is $\Delta_\gamma(x)$ is defined as the vector

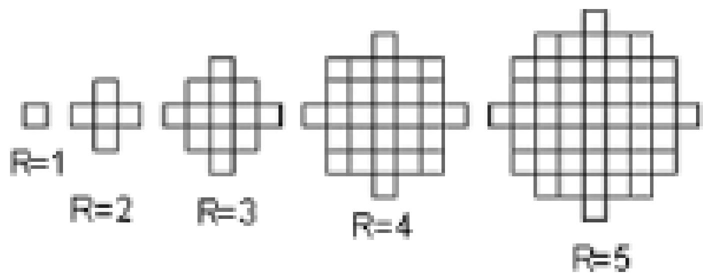
$$\Delta_\gamma(x) = \{\Delta_{\gamma_\lambda} : \Delta_{\gamma_\lambda} = |\Pi_{\gamma_\lambda} - \Pi_{\gamma_{\lambda-1}}|, \forall\lambda \in [1, n]\}. \tag{2.7}$$

The derivative of the closing profile is $\Delta_\varphi(x)$ is defined as the vector

$$\Delta_\varphi(x) = \{\Delta_{\varphi_\lambda} : \Delta_{\varphi_\lambda} = |\Pi_{\varphi_\lambda} - \Pi_{\varphi_{\lambda-1}}|, \forall\lambda \in [1, n]\}. \tag{2.8}$$



(a) Example morphological profile.



(b) Circular structuring elements with different radius sizes.

Figure 2.3: Morphological profile based on a circular structuring element, three openings, and three closings. In the shown profile, circular structural elements with R = 2, 4, and 6 were used (taken from [7]).

Figure 2.3 presents an example morphological profile, based on a circular structuring element. Figure 2.4 illustrates the derivative of the morphological profile relative to different points in a densely built-up area, whereas Figure 2.5 shows the morphological decomposition of the image in Figure 2.3.

In our work, we tested the usefulness of both MP and DMP features and concluded that they both can help in capturing the structure information of images. In our experiments, we used disk-shaped structuring elements and the sizes of structuring elements used in an MP extraction change with respect to the sizes of the connected components (objects) of that image. Details on the specific MP and DMP features can be found in Chapter 6.

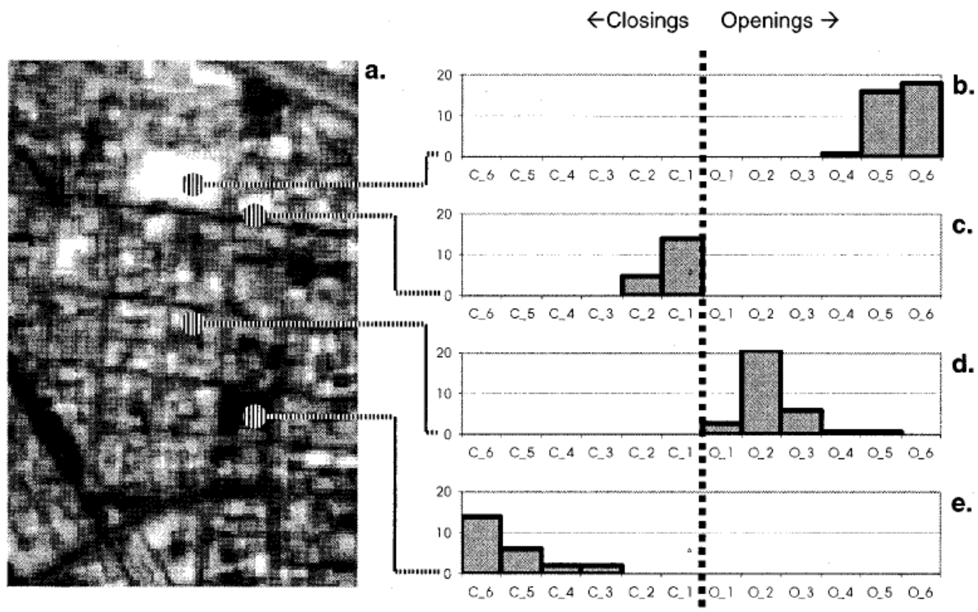


Figure 2.4: Derivative of the morphological profile relative to different points in a densely built-up area. (a) Original piece of IRS-1C satellite scene with 5 meters spatial resolution. (b) Commercial building. (c) Small street. (d) Residential building. (e) Small green area (image taken from [53].)

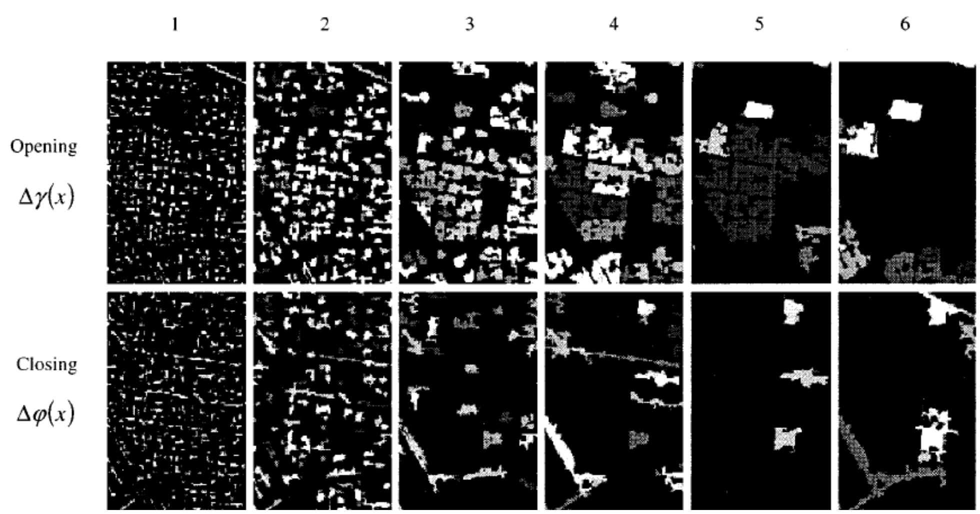


Figure 2.5: Morphological decomposition of the image in Figure 2.4 by using the derivative of the opening and closing profiles. The images have been visually enhanced. The derivative has been calculated relative to a series generated by six iterations of the elementary SE (size 3×3 pixels) (images taken from [53].)

2.4 Multispectral Features

Use of multispectral features can help in capturing the image contents; therefore, as we used in [4], we add the multispectral features of pixels into the feature vectors. These features help in obtaining additional information for classifying the image pixels whose frequency-domain or morphological features do not have enough discriminability. In all of our data sets, spectral values correspond to the red, green, and blue channels.

2.5 Dataset

The methodologies presented in this thesis will be illustrated using two different data sets.

2.5.1 Ikonos Image Set

Name of the sensor is IKONOS-2. Images are pan-sharpened multi-spectral, of 1 meter spatial resolution. There are totally 24 images, each of size approximately 11000×10000. In the experiments, we use four 2000×2000 parts of these images. We obtained the data from TUBITAK (The Scientific and Technological Research Council of Turkey). An example image is shown in Figure 2.6.

2.5.2 Digital Ortho Quarter Quads (DOQQ) Image Set

These are gray scale aerial images of 1 meter spatial resolution. There are totally 216 images, each of size approximately 6600×7600. We use five images of size 3000×4000 from this dataset. We obtained the data from the University of California, Santa Barbara, Vision Research Lab. An example image is shown in Figure 2.6.

(a) Example image from IKONOS image set.

(b) Example image from DOQQ image set.

Figure 2.6: Example images from datasets.

Chapter 3

Primitive Detection

3.1 Overview

As stated in Chapter 1, we consider the basic geospatial structures of images as *primitives* (as opposed to pixels in most of the literature work), and we aim to analyze the relationships and properties of these primitives using texture analysis. In this chapter, we first define the primitives of interest and present a methodology to detect them. The algorithms proposed in this thesis are illustrated with the detection of two geospatial objects: settlement areas and harbors. The first step in the modeling of these objects is the detection of primitives such as buildings for settlement areas, and boats and water for harbors.

On building detection, in the literature there exist techniques that are specifically designed for this task using spectral, edge and shape properties [46]. Most of the techniques that use optical data either assume that the edges representing boundaries can be successfully extracted and merged to delineate the buildings, or expect that buildings are surrounded by vegetation such as grass so that they can be separated from the background using thresholding (based on features such as Normalized Difference Vegetation Index, NDVI).

In this work, we made use of the textural, morphological and multi-spectral properties of buildings and the detection is performed with classification operations. Besides experimenting with traditional multi-class classifiers, we used one-class classifiers since the detection problem is a popular application of one-class classification phenomena. In classification problems, depending on the type of data (the sample sizes, the data distribution and how well the true distribution could be sampled), the best fitting data descriptions are sought. Unfortunately, classifiers hardly ever fit the data distribution optimally. Using just the best classifier and discarding the classifiers with poorer performance might waste valuable information [73]. To improve the performance, different classifiers (which may differ in complexity or training algorithm) can be combined. This may not only increase the performance, but can also increase the robustness of the classification [63]. During this research, one idea was to intersect the classifier boundary of one-class classifiers using the multi-class classifiers. Since multi-class classifiers try to partition the full feature space, there can be undefined regions in the feature space where there is a lack of training examples, and classification can be mostly erroraneous for data samples whose feature values belong to these undefined regions. Likewise, since one-class classifiers estimate a boundary around the target class, the boundary can be strictly covering the training examples and in many cases target objects can be classified as outliers resulting in high misdetection rates. We observed such cases in our experiments and what we concluded from those experiences is that, if it is possible to intersect the boundaries of the classifiers from different approaches, they would complement the lacking parts of each other and a better classification boundary could be obtained. With this observation, for both building and harbor primitives' detection, we conducted experiments that intersect the boundaries of one-class and multi-class classifiers that result in more robust results and better performances.

The second geospatial object of interest in this work is harbor. The detection of such complex (compound) geospatial objects necessitates new approaches to object detection since they are characterized by several parts and their spatial layout. For example, harbors contain boats, and golf courses contain trees and grass, both with a distinct spatial arrangement [9](See Figure 3.6).

As stated by [10], there are several obstacles to using strictly spatial analysis for the modeling and detection of compound objects. 1) Compound geospatial objects often contain a large number of parts, e.g., an harbor may contain hundreds of boats. 2) The structural relations among parts are often loose and vary from one object instance to another. In order to robustly recognize an object, this variation has to be accounted for. 3) Geospatial images are highly detailed, which are usually on the order of thousands of pixels in each dimension. These factors increase the computational expense of spatial-analysis methods.

To detect such geospatial objects, Bhagavathy [9] used Gabor filters tuned to different frequencies as texture features, and performed Gaussian mixture-based clustering of pixels as texture elements. Histograms of these elements within sub-windows were used for detection of golf courses and harbors. Even though texture features are sensitive to different pixel neighborhoods, histograms ignore spatial arrangements within a window, and cannot capture the placement patterns of texture elements.

For harbor detection, our work involves detection of individual object primitives (boats, water and other primitives in an harbor scene) using multi-spectral information and morphological characteristics. To detect each primitive of harbor (specifically the boat, water and a primitive representing "other" textural primitives of an harbor scene), we used textural and morphological characteristics of images, and as in the case of buildings, we combined the advantages of one-class and multi-class classifiers.

The rest of the chapter reviews the concepts in one-class vs. multi-class classification, and presents the details of building and harbor primitives' detection.

3.2 One-Class vs. Multi-Class Classification

Traditionally, many pattern recognition problems use multi-class classification techniques. These techniques train a classifier using example patterns for each class to learn a model that estimates decision boundaries in the feature space. This corresponds to a complete (exhaustive and exclusive) partitioning of the feature space where each part of the space corresponds to a particular class and is separated from the others [24]. On the other hand, the goal of one-class classification [66] is to accurately describe one class of patterns (called the target class) against the rest of the patterns (called outliers). Hence, a test sample is either detected as belonging to the target class or it is rejected. However, this is not the case in two-class (the special case of multi-class classification where number of classes is two) classifiers since they require sufficient number of training data for each of the classes; and this case is not always possible in real world problems. Therefore, to overcome this problem, one-class classifiers are proposed which model only the target class and assume a low uniform distribution for the outlier class. Another advantage of one-class classifiers is that different classifiers can use different features that are the most suitable for that target class whereas in the multi-class classifier, all classes are modeled in the same feature space. Below, we provide an overview of the one-class classification concept.

As stated in [66], the term one-class classification is believed to have originated from [48]. Other terms refering to the same or similar concepts that have been used in the literature are outlier detection [57], novelty detection [12] and concept learning [34]. One-class classification has proved valuable in a variety of research areas such as document classification [43], texture segmentation [65], image retrieval [37], ecological modeling [27] and remote sensing [60].

As stated in [66], the one-class classification problem differs in one essential aspect from the conventional classification problem. In one-class classification it is assumed that only information of one of the classes, the target class, is available. This means that just example objects of the target class can be used and that no information about the other class of outlier objects is present. The boundary between the target classes and other classes has to be estimated from data of only the normal, genuine class. The task is done using a boundary around the target class, such that it accepts as much of the target objects as possible, while it minimizes the chance of accepting outlier objects.

In all one-class classification methods two distinct elements can be identified [66]. The first element is a measure of the distance $d(z)$ or resemblance (or probability) $p(z)$ of a case z to the target class. The second element is a threshold θ on this distance or resemblance. New cases are accepted by the description when the distance to the target class is smaller than the threshold or when the resemblance is larger than the threshold. The one-class classification methods differ, however, in their optimization of $p(z)$ or $d(z)$ and thresholds with respect to the training set.

There exist different approaches in one class classification, such as the reconstruction methods [54, 74], density methods [23, 58] and boundary methods [75, 61]. Each approach has differing advantages in differing cases. The methods are summarized below.

Density methods are straightforward since they estimate the density of the training data and set a threshold on this density. These methods assume that the target data is derived from a family of known distributions (e.g., Gaussian or Parzen distributions). The probability density is then estimated from available data samples. Cases of unknown membership may then be assigned to the target class, if $p > \theta$, where θ is the threshold level chosen. Therefore, in the classification process, the choice of θ has a big impact. Several data distributions can be assumed where the Gaussian distribution assumes a unimodal and convex model of the data. A mixture of Gaussians can be used if the unimodality of a normal distribution is inappropriate. Another approach which is a linear combination of normal distributions is the Parzen-density estimation method, where a mixture of Gaussian kernels are centered on individual training points [11].

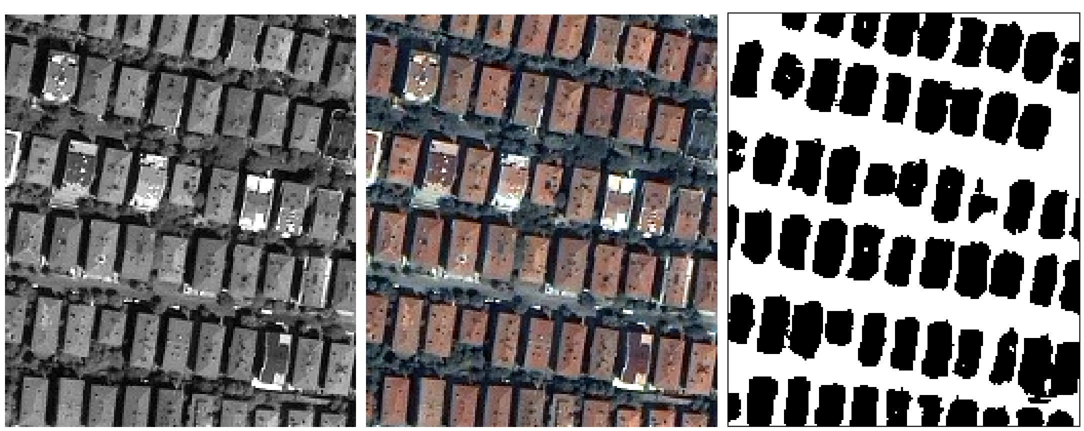
Reconstruction methods have not been primarily constructed for one-class classification but rather to model the data. By using prior knowledge about the data and making assumptions about the generating process, a model is chosen and fitted to the data. Most of these methods make assumptions about the clustering characteristics of the data or their distribution in subspaces. With the application of the reconstruction methods, it is assumed that outlier objects do not satisfy the assumptions about the target distribution. The reconstruction error of a test object is used as a distance to the target set. Because these methods were not developed for one-class classification, the empirical threshold has to be obtained using the training set [60]. The simplest method is the k-means classifier. In this method, it is assumed that the data are clustered and can be characterized by a few prototype cases. The distance d of a case z to the target set is then defined as the squared distance of that case to the nearest prototype [11]. In the self-organizing map, the placing of the prototypes is not only optimized with respect to the data but also constrained to form a low-dimensional manifold [36]. In these methods, the Euclidean distance is used in the definition of the error and the computation of the distance. Another method is the principal component analysis (PCA). The PCA mapping finds the orthonormal subspace, which captures the variance in the data as best as possible. The reconstruction error of a case z is now defined as the squared distance from the original object and its mapped version. Other reconstruction methods include those based on automatic encoders and diabolo networks [34, 31, 5].

In the boundary methods, only a closed boundary around the target set is optimized. In most cases, the distances or weighted distances d to a (edited) set of cases in the training set are computed and objects are accepted or rejected according to a threshold. For example, the k-center method covers the data set with k small balls with equal radii [76]. The ball centers are placed on training cases such that the maximum distance of all minimum distances between training cases and the centers is minimized. When the centers have been trained, the distance from a test case z to the target set can be calculated. In the nearest neighbor (NN) classifier, a test object z is accepted when its local density is larger or equal to the local density of its (first) nearest neighbor in the training set. This means that the distance from case z to its nearest neighbor in the training set is compared with the distance from this nearest neighbor to its nearest neighbor [21]. In the Support Vector Data Description (SVDD), a boundary in the form of a sphere contains all the target data within the smallest radius and all the outliers are assumed to lie outside this sphere. These outliers are identified by calculating the distance of a new case z to the center of the sphere [67].

3.3 Building Detection

To classify settlement areas as organized vs. unorganized, we first detect buildings of a scene, and by analyzing the spatial relationships between buildings of a scene, we determine an urbanization measure for that scene. For building detection, we experimented with different features: the pan-sharpened multi-spectral features, textural features and morphological profile features of Ikonos images. The classification is based on the Parzen-window estimation-based one-class classifier. We fine tuned this classifier with different two-class classifiers. An example is illustrated in Figure 3.2. Figure 3.3 examplifies intersecting the boundaries of different classifiers for fine tuning. This intersection corresponds to the combination of two classifiers using the Boolean "and" operation. Performances of each classifier with different feature sets are given in Chapter 6.

In the experiments, manually labeled pixels for buildings were used to train the target class and examples for roads, vegetation, soil, etc. were used for the non-building class in training the two-class classifiers. Example classification results in terms of a binary map that separates buildings from the background as shown in Figure 3.1.



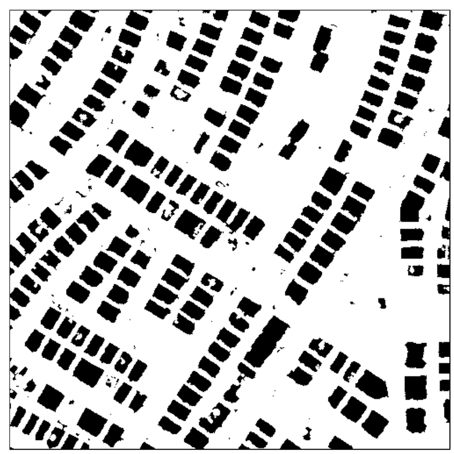
(a) Panchromatic band (b) Pan-sharpened RGB bands (c) Building map

Figure 3.1: Panchromatic and multi-spectral bands of an example scene, and the binary classification map of buildings.

(a) Pan-sharpened RGB bands of input image

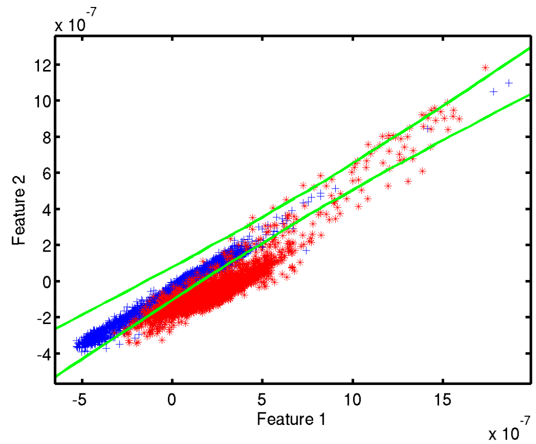
(b) Detected buildings when classification is done with two-class quadratic gaussian classifier (qdc)

(c) Detected buildings when classification is done with one-class parzen-window estimator (parzen)

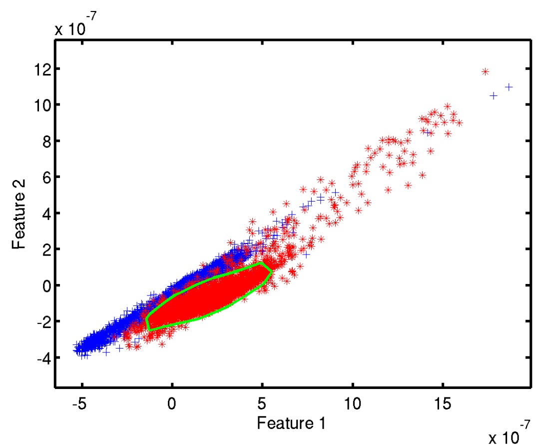


(d) Detected buildings when classification is done with intersecting the boundaries of qdc and parzen

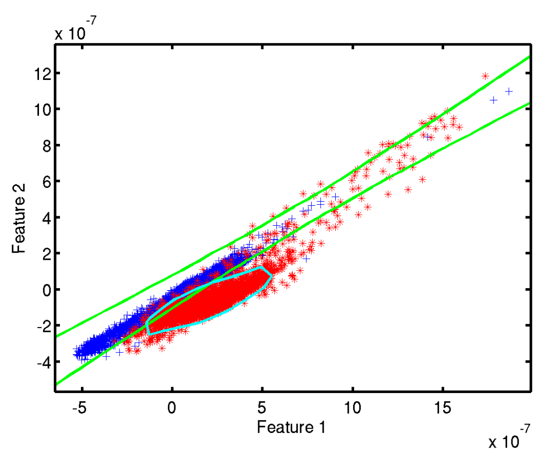
Figure 3.2: The use of combining different one-class and multi-class classifiers for better classification performance.



(a) Two-class quadratic Gaussian classifier boundary



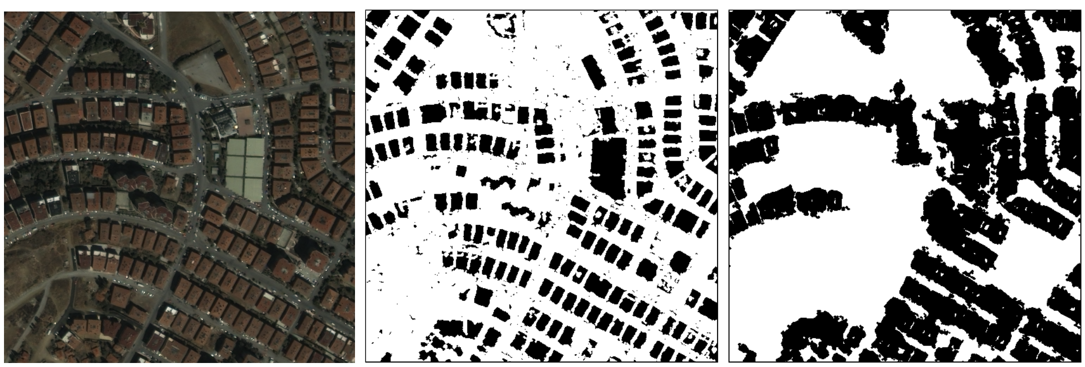
(b) Parzen classifier boundary



(c) Intersecting the boundaries to obtain a new classifier

Figure 3.3: Different classifier boundaries on R and B bands' features of the scene in Figure 3.2. Target class (building) instances are shown in red, outlier class (non-building) instances are shown in blue, and classifier boundaries are highlighted in green.

During experiments, what we observed was, even though for some feature combinations the overall pixel-based error rate was smaller than some other cases, visual examination of the classification maps showed that error rates do not always reflect the visual quality of the results as shown in Figure 3.4. This also supports our belief that high success rates achieved using pixel-based classification methods and limited pixel-based ground truth do not present semantically satisfactory performance, and more powerful structural models are needed for further improvements. Detailed experiments are given in Chapter 6, and Figure 3.4 illustrates that even though the error rate is lower when Gabor features are added (using pixel-based ground truth), individual buildings can be isolated better when only RGB bands are used.

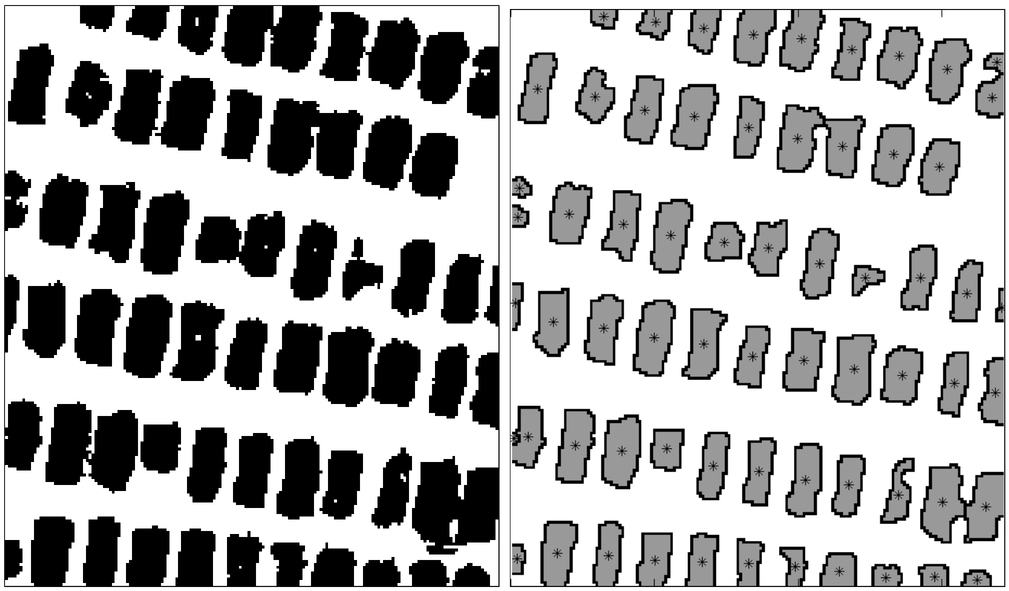


(a) Pan-sharpened RGB bands (b) Detected buildings using RGB bands (c) Detected buildings using RGB bands and Gabor features

Figure 3.4: Multi-spectral bands of an example scene and the binary classification maps of buildings. Even though the error rate is lower when Gabor features are added (using pixel-based ground truth), individual buildings can be isolated better when only RGB bands are used. (Results are shown before morphological cleaning.)

The resulting classification can have some false positives especially along some roads and soil areas. We clean noisy pixels in the background (specifically, by morphologically eroding the image with a disk structuring element of radius 1) and fill small holes inside the resulting connected components using morphological operations. We also compute the distance transform, suppress insignificant local minima, and apply the watershed transform to separate buildings as individual regions if they are touching each other with a small amount after classification.

These individual buildings are used as texture primitives in the rest of the work. Figure 3.5 illustrates the use of post processing step.



(a) Detected buildings

(b) Detected buildings after post-processing - centroids are shown with stars

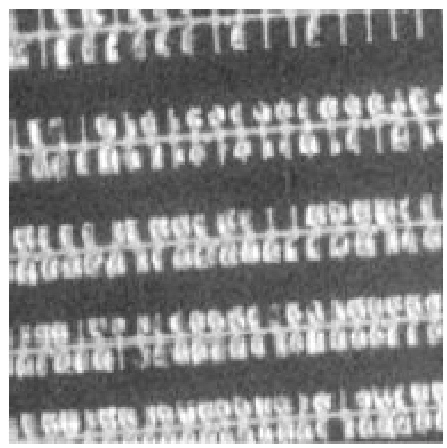
Figure 3.5: Detected buildings of an example scene, and the same scene after post processing

3.4 Boat and Water Detection

For harbor detection, our work involves detection of individual object primitives (boats, water and other primitives) using multi-spectral information, texture features and morphological characteristics. The detection of individual harbor primitives is used to perform the texture-based modeling of these primitives' spatial arrangements within image scenes (Chapter 4).

We use three one-class classifiers to detect the textural primitives of an harbor, namely the boat, water and others (see Figure 3.6). We manually label some pixels of example images to train the classifiers. For instance, to train the one-class classifier for boat primitive, sample boat pixels are used; or to train the

(a) An harbor instance in white borders



(b) The arrangement of texture primitives (boats and water) in the harbor instance

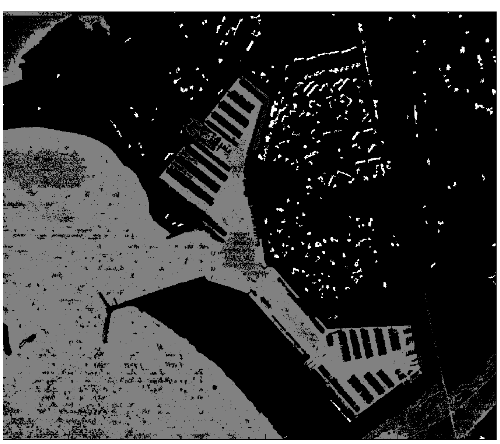
Figure 3.6: An example complex geospatial object (harbor) and its texture primitives.

(a) A scene with harbor instances

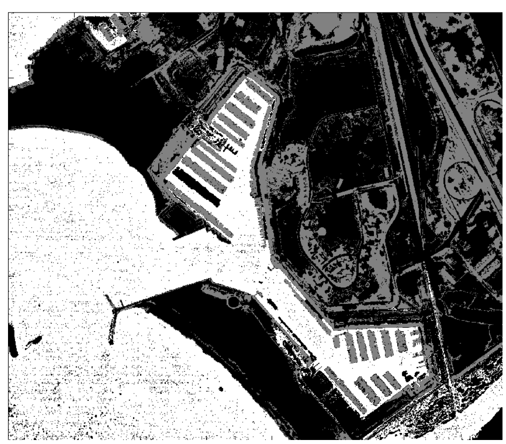
(b) Output of one-class boat classifier

(c) Output of one-class water classifier

(d) Output of one-class classifier for others class



(e) Output of multi-class quadratic gaussian classifier



(f) Output of combination of one-class classifiers with the multi-class classifier

Figure 3.7: A scene with harbor instances, output of each classifier and resulting combined classifier. Gray levels in classifier outputs represent probabilities.

one-class water primitive, sample water pixels are used. This way, three one-class classifiers, one for each primitive of an harbor, are obtained. In deciding the class labels of a pixel, that pixel is assigned to the class that has the highest target probability among all classes. In this phase, as in building detection, we perform the Boolean “and” operation on the one-class classifiers’ boundaries and the boundaries of a multi-class classifier. The multi-class classifier in this part is trained with sample pixels for boat, water and others classes. Figure 3.7 illustrates the methodology. In this example, the features used for classification form a morphological profile. In the morphological profile, the gray-scale image itself and the outputs of opening and closing by reconstruction operations are used. Disk structuring elements with the following radius sizes are used in opening and closing by reconstruction operation: 2, 5, 7, 10, 13, 14, 15, 16, 17, 18, 19, 20 pixels. These radius sizes are chosen since the structures of harbor examples are mostly of the given structuring element sizes. Further examples can be found in Chapter 6.

When we examine the output of one-class boat classifier in Figure 3.7, we see that most of the boat examples are correctly classified in the target class (pixels classified as belonging to target class are shown in white, whereas outliers are shown in black). However, besides boat instances, there are other image parts such as the buildings, soil land or roads that are also classified in the target class. Similar errors exist in the detection of other harbor primitives, but we claim that such false alarms in the primitive detection are not problematic if we use the spatial distribution information in the right way. As it can be observed from the output of combination of one-class classifiers with the multi-class classifier in this example figure, we see that in only harbor regions, rows of boats are repetitively coming side-by-side with the water pixels. We propose that the image regions where rows of boat pixels are aligned side-by-side with rows of water pixels, such regions belong to an harbor instance. Therefore, even though the detection of individual harbor primitives can have false alarms, in Chapter 4, by analyzing spatial relationships, we eliminate them and obtain almost exact regions of the harbors.

Chapter 4

Statistical Texture Model

4.1 Overview

Detection of high-level structures needs region-based analysis of remotely sensed imagery. However, segmentation techniques usually assume that regions consist of uniform feature distributions and cannot delineate areas that include several objects (for example buildings, harbor instances) and the background (trees, grass, roads, etc.). Therefore, dividing images into non-overlapping sub-windows and analyzing the individual sub-windows have been the common approach used for image partitioning [9, 68]. Given a rectangular window, features are extracted to characterize the content of this window. This characterization is usually performed using histograms or summary statistics (e.g., mean, standard deviation) of pixel level features [9]. However, histograms and statistics ignore spatial arrangements within a window and cannot capture the placement patterns of objects.

As defined in Chapter 1, texture can be characterized by the relationships of unit elements (textural primitives) within neighborhoods. To determine the relationships among textural primitives numerous methods have been suggested in the literature. However, an important problem has been the definition and detection of textural primitives. Thus, traditionally, pixels are used as the unit elements and features are extracted for pixel neighborhoods.

Three most important perceptual dimensions in natural texture discrimination are "repetitiveness", "directionality", and "granularity" [42]. Therefore, the goal of texture models is to extract features that relate to periodicity, directionality and randomness. The most popular models used toward this goal include the co-occurrence matrix [29], the Fourier transform [70], and the autocorrelation function [41]. Studies have shown that features extracted from co-occurrence matrices computed at increasing inter-pixel distances at a particular orientation can be used to detect coarseness, directionality, and periodicity at that orientation [16, 77, 64]. Similarly, peaks in the Fourier power spectrum [70] and the autocorrelation function [41] can be used to detect periodicity.

In this work, to differentiate organized settlement areas from unorganized ones, we use buildings as the unit primitives. And to detect harbor instances we used water and boat rows as the unit primitives. Then spatial domain and frequency domain texture features are used for characterizing the primitives' spatial arrangements. Details of the features are described below.

4.2 Spatial Domain Texture Analysis

Several comparative studies in both remote sensing and general computer vision literature showed that features extracted from co-occurrence matrices are very effective features for texture analysis [70]. Co-occurrence, in general form, can be specified in a matrix of relative frequencies $P(i, j; d, \theta)$ with which two texture elements separated by distance d at orientation θ occur in the image, one with property i and the other with property j. For particular values of d and θ, the resulting square matrix can be normalized by dividing each entry by the number of elements used to compute that matrix.

In order to use the information contained in co-occurrence matrices, Haralick *et al.* [29] defined 14 statistical features that measure textural characteristics such as homogeneity, contrast, organized structure, and complexity. Conners and Harlow [16] showed that the local minima of the contrast (inertia) feature among these 14 can be used to detect periodicity at a given orientation. Zucker and Terzopoulos [77] defined a χ^2 (chi-square) statistic that is also based on co-occurrence matrices to measure the amount of structure at a particular inter-pixel distance and orientation. Starovoitov *et al.* [64] compared twenty-two co-occurrence-based features for periodicity analysis, and concluded that seven of these features are useful for this purpose.

We use both the contrast feature and the χ^2 statistic to detect periodicity and directionality. The contrast feature computed from a co-occurrence matrix at particular d, θ and N (number of distinct gray levels) is [28]

$$x_i = \sum_{i,j=0}^{N-1} (i-j)^2 P(i,j) \tag{4.1}$$

and the χ^2 statistic is [77]

$$y_i = N \left(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \frac{P(i,j)^2}{r_i c_j} - 1 \right) \tag{4.2}$$

where

$$\begin{aligned} r_i &= \sum_{j=0}^{N-1} P(i,j) \\ c_j &= \sum_{i=0}^{N-1} P(i,j) \end{aligned} \tag{4.3}$$

We compute these features at 1 to 60 inter-pixel distances and eight different orientations $i\frac{\pi}{8}$, $i = 0, \ldots, 7$, shown in Figure 4.1.

Example building patterns and the corresponding features are given in Figure 4.2 through Figure 4.7. These examples show that the features at a particular orientation exhibit a periodic structure as a function of distance if the neighborhood contains a regular arrangement of buildings along that direction. On the other hand, features are very similar for different orientations if there is no particular arrangement in the neighborhood.

To extract a single feature vector, we sum the feature values along each orientation and obtain a feature vector of length 8 (one value for each direction). For this step, we tried different approaches such as the Wavelet decompositions,

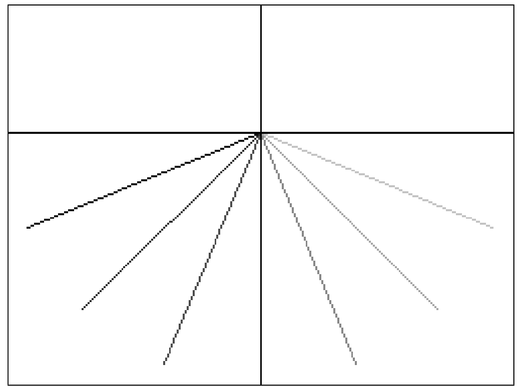


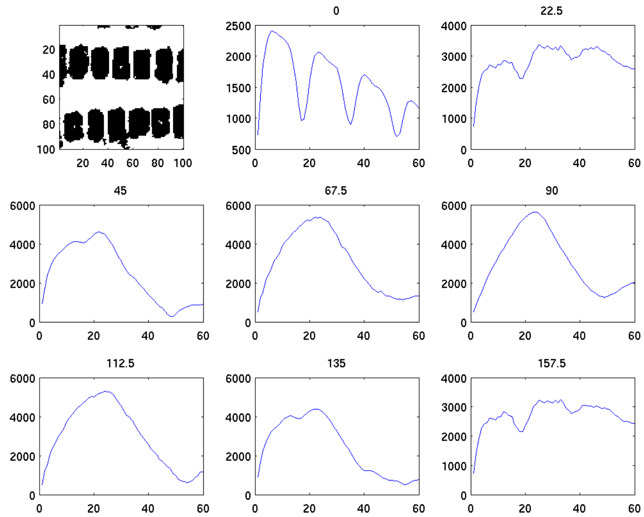
Figure 4.1: Orientations used for computing the co-occurrence matrices.

Fourier transforms and periodicity transforms [62] of each feature vector in each direction, however we did not obtain results as satisfactory as the case of summing the features of each orientation. Vectors for contrast and χ^2 features are computed separately. Finally, values in these vectors are sorted to achieve rotation invariance.

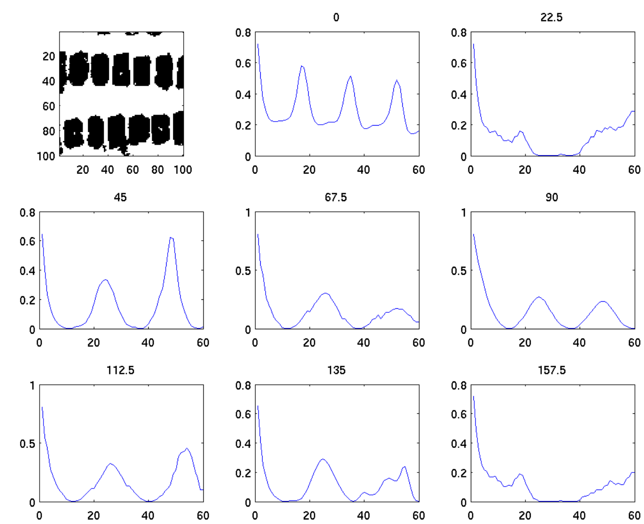
4.3 Frequency Domain Texture Analysis

It is well known that the radial distribution of values in the Fourier spectrum of an image (which is analogous to spatial autocorrelation) is sensitive to texture coarseness in that image. A coarse texture will have high values concentrated near the origin of the spectrum, while a fine texture will cause the values to be spread out [70]. It is also well known that the angular distribution of values in the spectrum is sensitive to the directionality of the texture in the image. A texture with many edges in a given direction θ will have high values of the spectrum concentrated around the perpendicular direction $\theta + \pi/2$, while for a non-directional texture, the spectrum is also non-directional [70]. Therefore, an analysis of the Fourier spectrum can provide information about the spatial periodicity and the principal direction of the texture patterns.

Given the spectrum function $S(r, \theta)$ expressed in polar coordinates, features

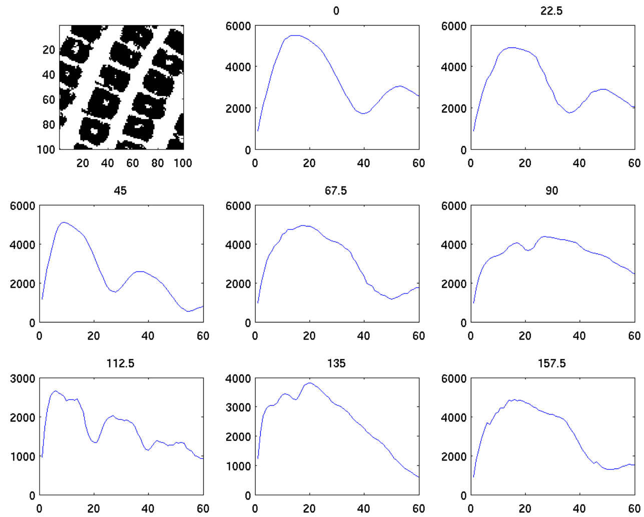


(a)

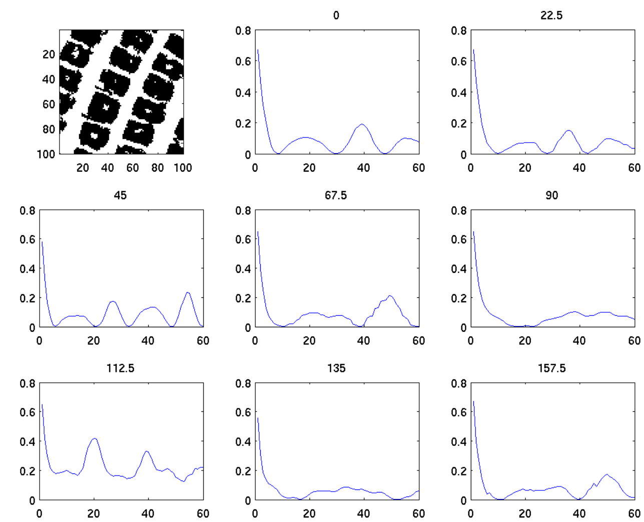


(b)

Figure 4.2: Example building pattern with corresponding co-occurrence-matrix based textural features. x-axes in the feature plots represent inter-pixel distances of 1 to 60. (a) For the given example building pattern, the contrast features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. (b) For the given example building pattern, the χ^2 features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. The buildings in the example have periodic structure along horizontal direction, and features in $0°$ have regular peaks illustrating this fact.

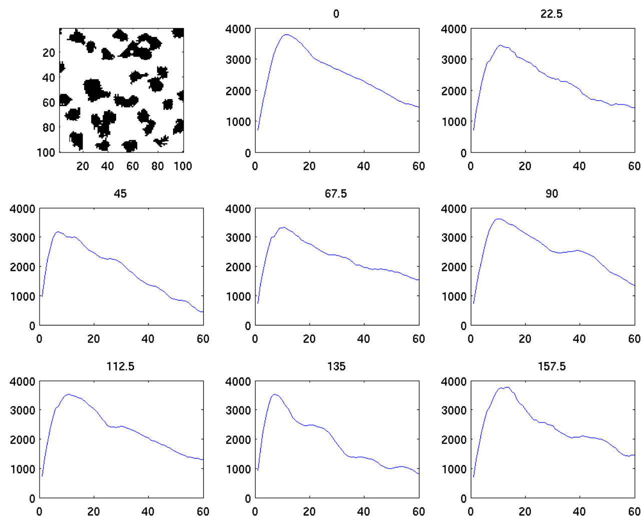


(a)

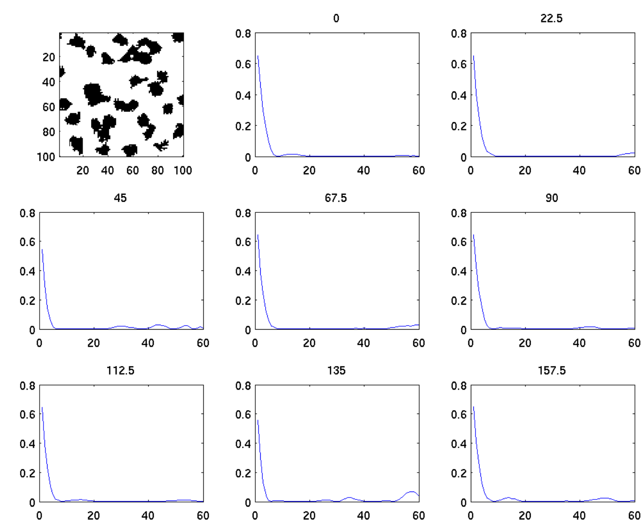


(b)

Figure 4.3: Example building pattern with corresponding co-occurrence-matrix based textural features. x-axes in the feature plots represent inter-pixel distances of 1 to 60. (a) For the given example building pattern, the contrast features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. (b) For the given example building pattern, the χ^2 features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. The buildings in the example have periodic structure along diagonal direction, and features in 112.5° have regular peaks illustrating this fact.

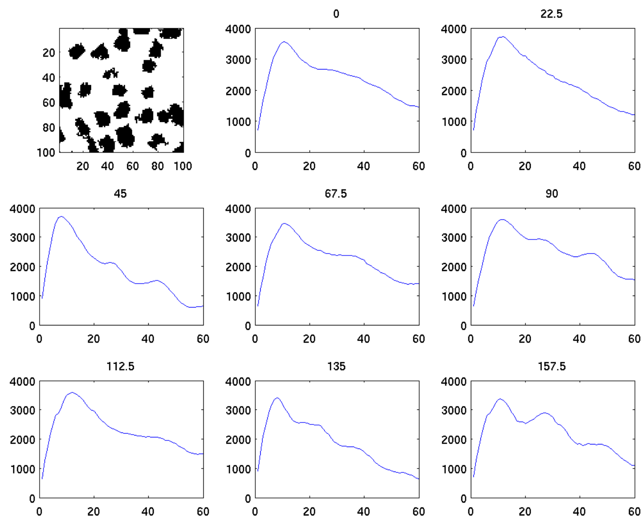


(a)

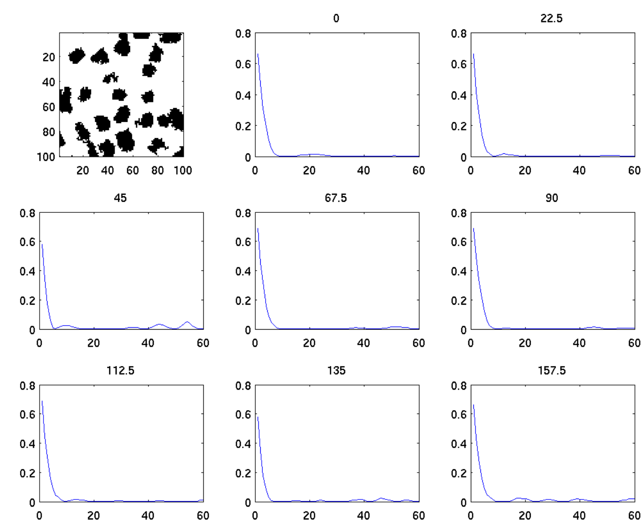


(b)

Figure 4.4: Example building pattern with corresponding co-occurrence-matrix based textural features. x-axes in the feature plots represent inter-pixel distances of 1 to 60. (a) For the given example building pattern, the contrast features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. (b) For the given example building pattern, the χ^2 features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. The buildings in the example have random alignment in that neighborhood, and there occurs no periodic peaks in any of the directions examined.

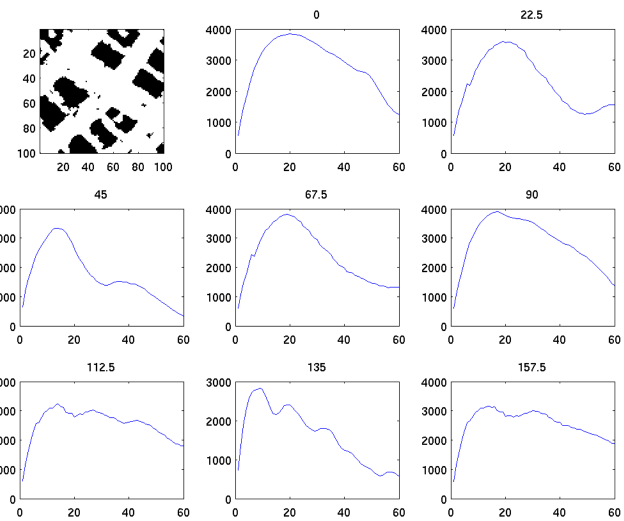


(a)

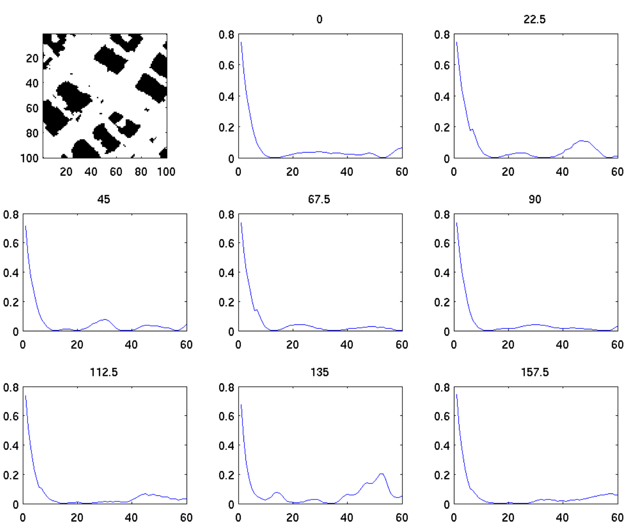


(b)

Figure 4.5: Example building pattern with corresponding co-occurrence-matrix based textural features. x-axes in the feature plots represent inter-pixel distances of 1 to 60. (a) For the given example building pattern, the contrast features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. (b) For the given example building pattern, the χ^2 features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. The buildings in the example have random alignment in that neighborhood, and there occurs no periodic peaks in any of the directions examined.

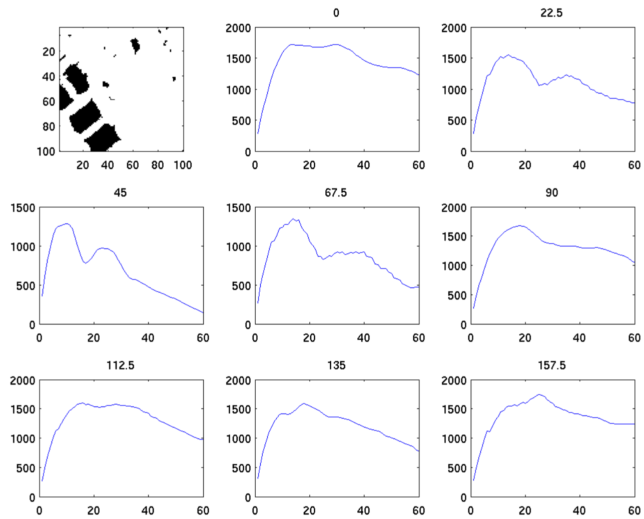


(a)

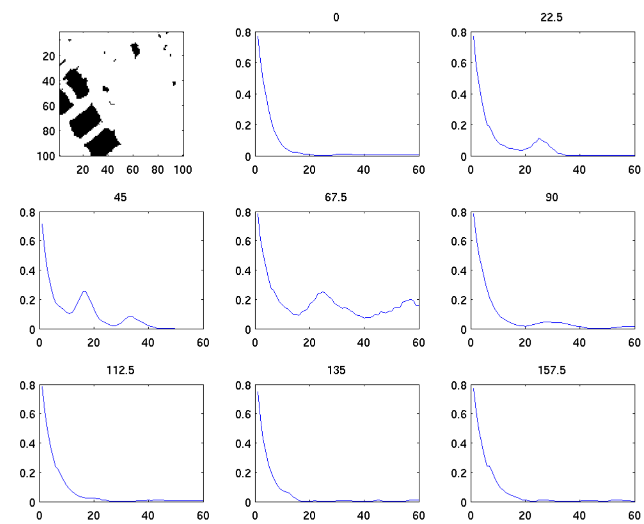


(b)

Figure 4.6: Example building pattern with corresponding co-occurrence-matrix based textural features. x-axes in the feature plots represent inter-pixel distances of 1 to 60. (a) For the given example building pattern, the contrast features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. (b) For the given example building pattern, the χ^2 features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. The buildings in the example are almost aligned in diagonal direction, and features in 135° have some peaks illustrating this fact.



(a)



(b)

Figure 4.7: Example building pattern with corresponding co-occurrence-matrix based textural features. x-axes in the feature plots represent inter-pixel distances of 1 to 60. (a) For the given example building pattern, the contrast features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. (b) For the given example building pattern, the χ^2 features for [0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5] degree orientations. The buildings in the example have periodic structure along 45° direction although the window does not contain buildings homogeneously, and features in 45° have some peaks illustrating this fact.

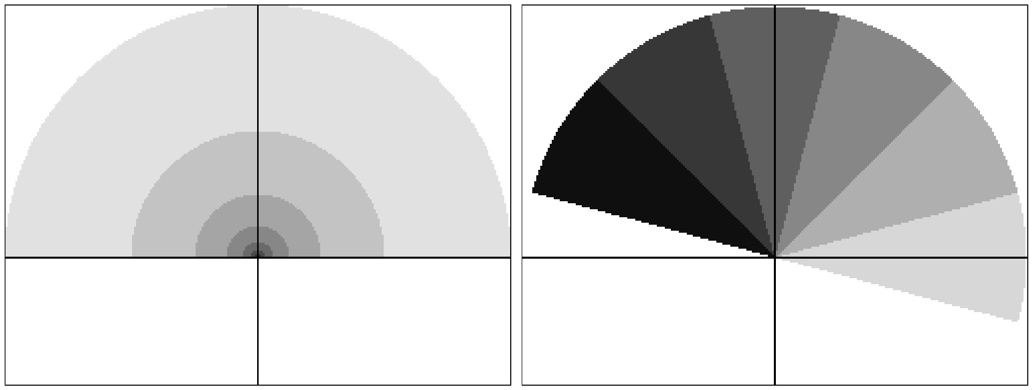


Figure 4.8: Rings (left) and wedges (right) for computing the features based on the Fourier spectrum. Seven rings and six wedges are shown as example.

that capture texture periodicity and directionality can be computed by integrating (summing in the discrete case) S over ring-shaped and wedge-shaped regions centered at the origin, respectively. The ring-based (periodicity) features have the form

$$x_i = \sum_{r=r_i}^{r_{i+1}} \sum_{\theta=0}^{\pi} S(r,\theta) \tag{4.4}$$

where r_i and r_{i+1} represent the inner and outer radii of the ring. The values of r used in the experiments are computed as powers of 2 (e.g., $[0,2)$, $[2,4)$, $[4,8)$, etc.) as suggested in [70]. The wedge-based (directional) features have the form

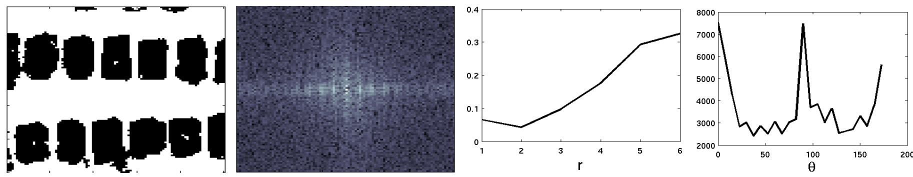
$$y_i = \sum_{\theta=\theta_i}^{\theta_{i+1}} \sum_{r=1}^{r_{\max}} S(r,\theta) \tag{4.5}$$

where θ_i and θ_{i+1} represent the slope of the wedge and $r_{\max}$ is the radius of a circle centered at the origin. The DC component is omitted since it is common to all wedges. The values of θ used in the experiments are computed as $\theta_i = (2i-1)\frac{\pi}{2n}$, $i = 0, \ldots, n$, where n is the number of wedges and is set to 24. Example rings and wedges are shown in Figure 4.8.

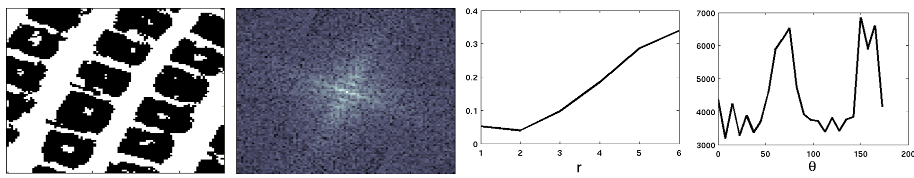
Example building patterns and the corresponding feature vectors are given in Figure 4.9. These examples show that the peaks in the features correspond to the periodicity and directionality of the buildings, whereas no dominant peaks can be found when there is no regular building pattern. Since we are interested only in the periodicity (i.e., organization) but not the directionality, the wedge-based feature vectors are circularly shifted so that the largest value is at the origin for rotation invariance.

4.4 Scene Classification

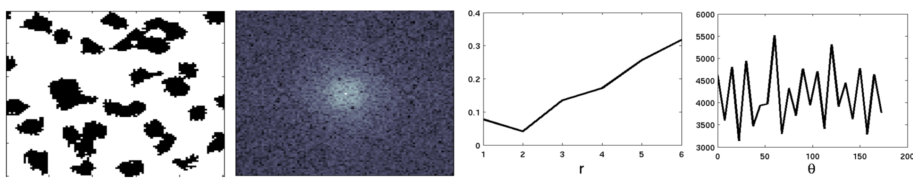
Given a large scene, first, its primitives of interest are detected as explained in Chapter 3. Then, the resulting detection map is partitioned into overlapping sub-windows and textural features are computed for each sub-window as described in Sections 4.2 and 4.3. Finally, each sub-window is classified using binary decision tree classifiers trained by manual labeling of several sub-windows as regular (organized) or irregular (unorganized). In this part, to reduce the effect of window boundaries and to add the neighboring windows' information into decision of labeling the considered window, we follow a *voting* schema on the overlapping windows. For each window, we slide the window boundaries by half, horizantically, vertically and both horizontal-vertically, and this way, for a window we obtain three slided versions (except the windows of the upper and leftmost side of the image). In this schema, a region is classified four times (again except the regions in the explained sides), and in each of these cases, this region is neighbored with different parts of the image. This way, we consider the different neighbors of a region. In deciding the label of this region, we consider the most frequent label that region obtains from the four windows. Besides, in the classification, decision trees were chosen because they do not require any assumptions about neither the distributions nor the independence of features, and they also automatically perform feature selection. Examples of scene classification are illustrated in Chapter 6.



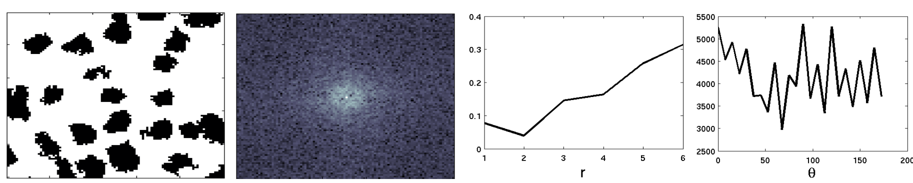
(a)



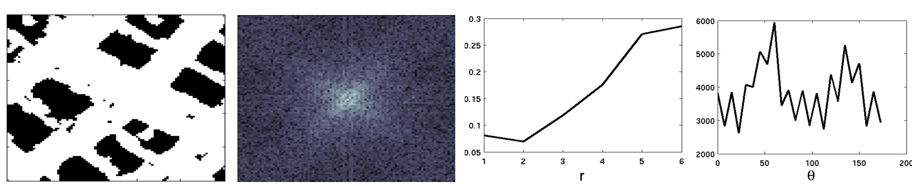
(b)



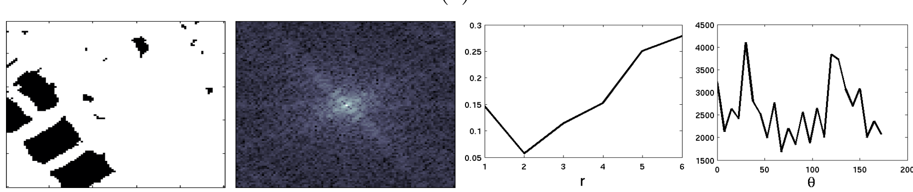
(c)



(d)



(e)



(f)

Figure 4.9: Example building patterns (first column), Fourier spectrum of these patterns (second column), and the corresponding ring- and wedge-based features (third and fourth columns). x-axis for ring-based feature plot represents the ring and for wedge-based feature plot it represents orientations of wedges. In (a) and (b), peaks in feature plots are observed since the buildings have organized patterns. In (c) and (d), no dominant peaks in feature plots are observed since the buildings have no organized patterns. In (e) and (f), peaks in feature plots are observed since the buildings have almost organized patterns.

Chapter 5

Structural Texture Model

5.1 Overview

In this part, we propose a structural technique for modeling high-level geospatial objects and their neighborhoods in an urban setting. The approach starts with the detection of urban primitives (e.g., buildings, trees, roads, etc.). Then, the neighborhoods are modeled in terms of the spatial arrangements of these primitives. This is achieved using a graph-based model that clusters the primitives into groups that are composed of primitives with similar spatial arrangements. This representation can be viewed as a "generalized texture" measure where the image elements of interest are urban primitives instead of the traditional case of pixels. The grouping phase can be modeled as a structural pattern recognition problem that uses graph-based representations and clustering techniques.

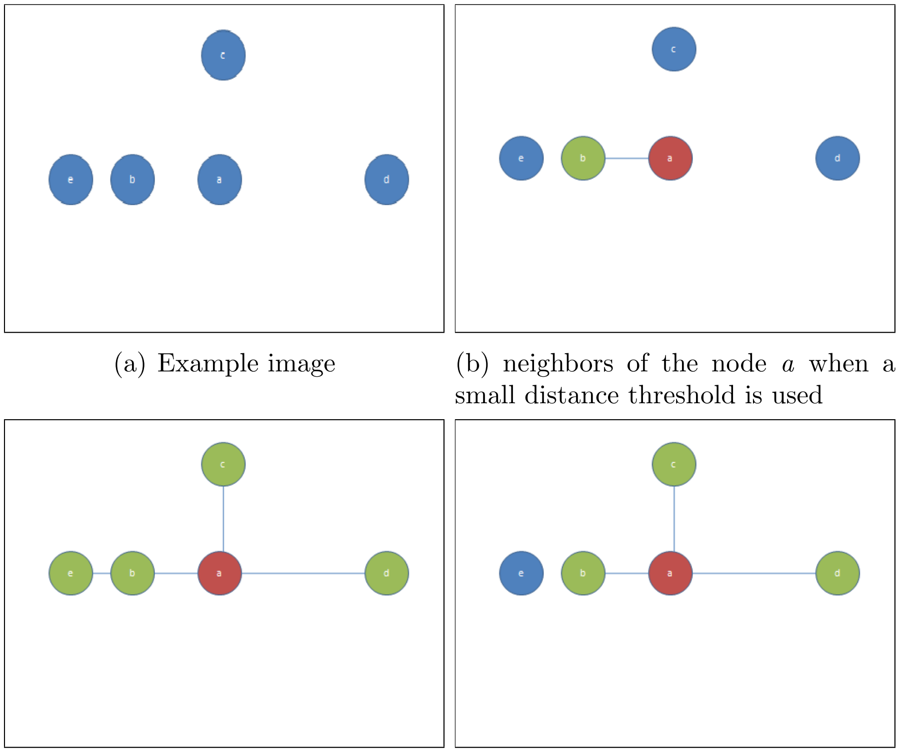
We illustrate the proposed approach in the problem of measuring the level of urbanization according to spatial building patterns where the graph nodes correspond to individual buildings, and the clusters of the graph correspond to building groups with similar arrangements.

5.2 Graph-Theoretic Scene Model

For this work, we detect the buildings inside input images with the methodology explained in Chapter 3. To analyze the spatial arrangements of individual buildings, we propose a graph-theoretic approach. In the literature, dividing images into non-overlapping sub-windows and analyzing the individual sub-windows has been the common approach used for image partitioning [68, 4, 9]. Such partitions assume that the window contents have uniform feature distributions. However, it is often difficult to select a window size that is small enough to have uniform content but is large enough to cover complex geospatial structures. Hence, window-based modeling cannot handle geospatial objects at multiple scales. In this work, we propose a graph-theoretic image partitioning technique that captures the spatial arrangements of objects (which cannot be done by histogram or summary-statistics-based analysis) and that is successful even for image regions containing different types of objects (e.g., multiple buildings).

The scene model involves the construction of a graph-based representation where the graph nodes correspond to the primitives and edges model their spatial arrangements. First, neighboring primitives are found using the Voronoi tessellation corresponding to the primitive centroids. An edge is created between two nodes if they are assigned as neighbors in this decomposition. In the related literature on graph-based pattern proximity analysis, the distance between nodes (e.g., buildings) is compared to a threshold that is used to define the edges. Such approaches may be successful in some specific applications where distance between the primitives is almost constant and can give sufficient information about their spatial proximity. However, such thresholds are scale dependent and can produce poor results in modeling the spatial arrangement in applications that involve high amount of variations in terms of the structures of the primitives and the relationships between these structures. The insufficiency of distance-based neighborhood analysis is illustrated in Figure 5.1.

After finding the neighbors of each primitive, the goal is to group these primitives into clusters so that they can be automatically classified as regular or irregular. Voronoi tessellation of primitives can assign some nodes that are considerably



(a) Example image

(b) neighbors of the node a when a small distance threshold is used

(c) neighbors of the node a when a large distance threshold is used

(d) neighbors of the node a when Voronoi tessellation is used

Figure 5.1: This example illustrates that distance is not a good metric to define neighbors of node a. The rational neighbors are obtained by Voronoi tessellation.

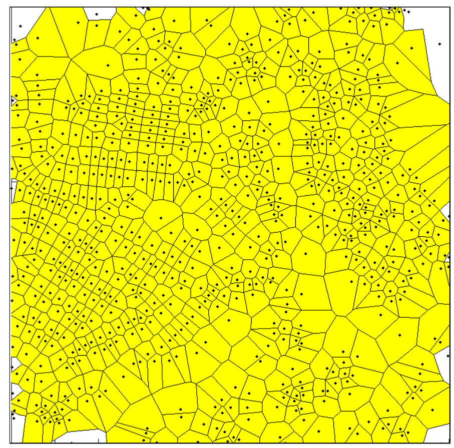
distant from each other as neighbors, and this is not a desired property in analyzing the relationships of neighboring primitives. Therefore, to determine the most important neighbors of each primitive, the minimum spanning tree of the graph is constructed using the distances of Voronoi neighbors. This way, in the minimum spanning tree, a node is connected to its most important and most related neighbors and its relationship with the far away neighbors can be ignored (Figures 5.2 and 5.3 illustrates the steps in graph construction on different examples).

To cluster the primitives into groups, some edges of the minimum spanning tree should be removed. The edges that are removed are selected as the ones that are at least 50% longer than the average edge length in the minimum spanning tree. (The threshold was selected empirically after a search procedure using the ground truth described in Section 6.4.) As a result, such long edges are removed and nodes that are spatially close enough remain in the same cluster.

5.3 Labeling Graph Regions

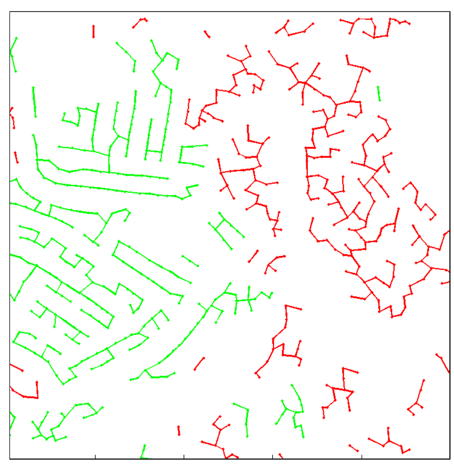
The spatial arrangements of interest in this work are the organization of buildings. After the graph representing the scene is formed and the clusters of buildings are found, the next step is the classification of these clusters as regular (organized) or irregular (unorganized). As illustrated in Figure 5.2, in organized neighborhoods, buildings are mostly aligned linearly or they have a regular grid-like arrangement. Consequently, when the angles between buildings in a cluster are examined, it can be seen that in organized neighborhoods the angle distribution has peaks around 90 and 180 degrees, whereas for irregularly aligned areas (where there is no specific arrangement of primitives), random angle distributions are observed with no considerable peaks. Therefore, the angles between connected nodes of a cluster are computed (three nodes are used for each angle) and a histogram of these angles are formed for each cluster. Then, a cluster is labeled as organized if in its histogram the count of angles in the two bins including 90 and 180 degrees is greater than the total count of angles in the rest of the bins. Note that, the measures used for both clustering and labeling are scale and rotation invariant because neighborhoods are computed from Voronoi tessellations and relative angles are computed between these neighbors, respectively.

(a) Example image with different building patterns



(b) Voronoi tessellation of the image with centroids on it.

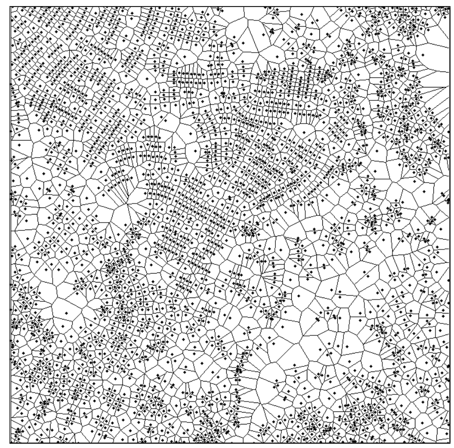
(c) Minimum spanning tree formed using Voronoi neighborhood distances.



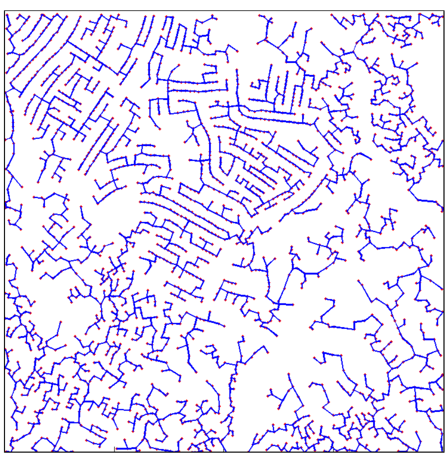
(d) Groups formed after cutting minimum spanning tree and result of labeling them as organized/unorganized.

Figure 5.2: Phases of graph construction and labeling each group of nodes (clusters labeled as organized and unorganized are shown in green and red, respectively).

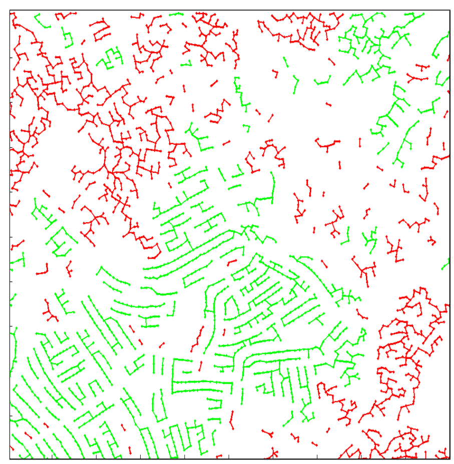
(a) Example image with different building patterns



(b) Voronoi tessellation of the image with centroids on it.



(c) Minimum spanning tree formed using Voronoi neighborhood distances.



(d) Groups formed after cutting minimum spanning tree and result of labeling them as organized/unorganized.

Figure 5.3: Phases of graph construction and labeling each group of nodes (clusters labeled as organized and unorganized are shown in green and red, respectively).

Chapter 6

Experiments And Results

6.1 Overview

In this part, we present the experiments conducted for the proposed work in this thesis and results of these experiments. The experiments for settlement area detection were conducted using four Ikonos images (2000×2000 each) of Ankara, Turkey. One DOQQ image (3000×4000, with multispectral features) was used for the harbor detection experiments (We have five gray-scale DOQQ images, however we have just one multi-spectral DOQQ image).

Firstly, primitive detection is illustrated on building, boat and water primitives. For each primitive, the ground truth, the features used and the classification methodology is explained. Error rates and example results are shown. Then, scene classification using the statistical texture model is examplified. Ground truth and evaluation methodology is presented on examples. Then, the structural approach for scene classification is illustrated, with discussions on the results obtained.

In the last part, we applied two of the methods recently proposed in the literature to our datasets, with some small variations due to the difference in implementation. To compare our method with the existing ones, the selected methods are applied since they were also supposed to be successful in finding complex geospatial objects.

6.2 Primitive Detection

6.2.1 Building Detection

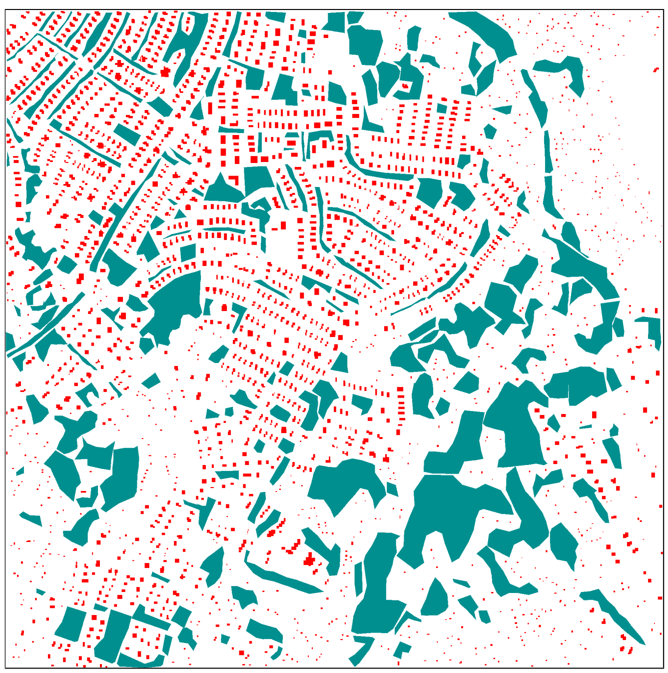
For building the ground truth of building detection, pixels within parts of four test scenes were manually labeled as buildings vs. others. These pixels (summarized in Table 6.1) form independent training and testing sets for evaluation of building detection. Note that the one-class classifier (based on Parzen window estimation) was trained by only 41,138 samples of the target class (building class); the non-building samples were merged in the training of multi-class classifiers. Figure 6.1 through 6.4 shows the test images and their corresponding ground truth.

Table 6.1: Training and testing data used in evaluating building detection.

Datasets	# building pixels	# non-building pixels
Training	305,937	2,839,316
Testing	203,958	1,892,878

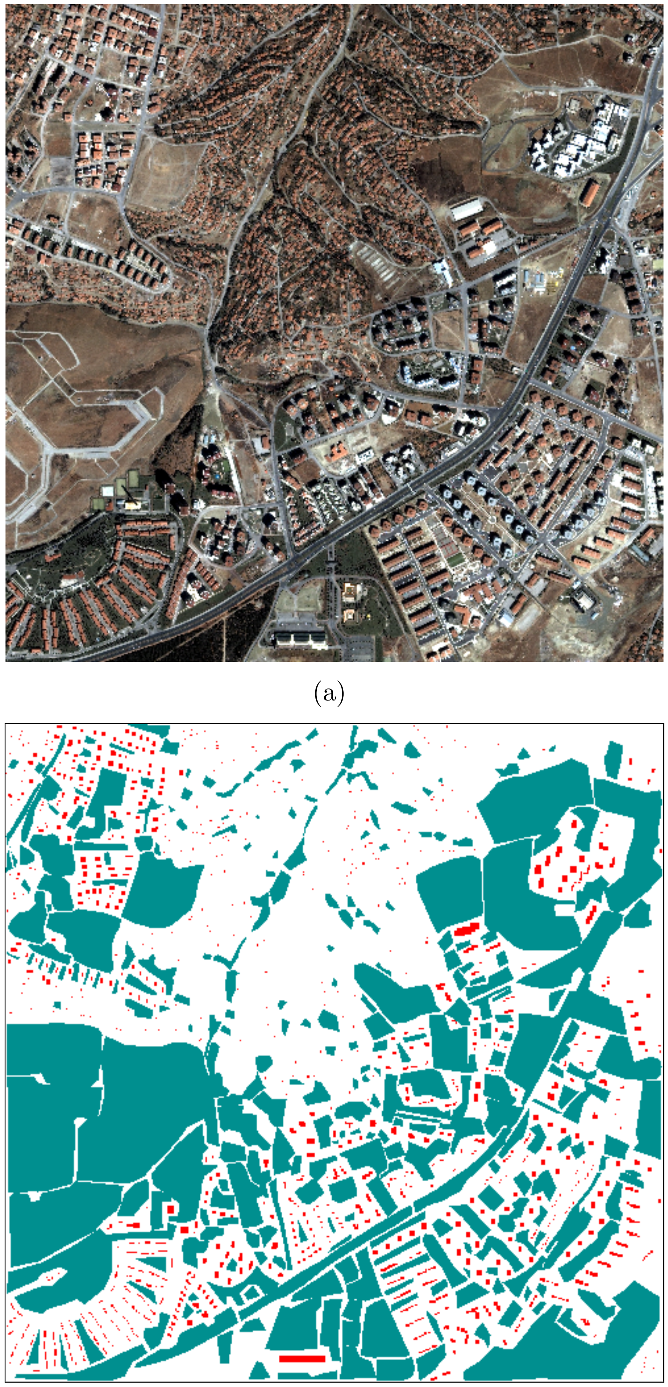
In building detection, different combinations of spectral (RGB) features, Gabor texture features, and morphological profile features were considered (Table 6.2). Table 6.3 summarizes the percentage success rates obtained using different features with different classifiers. Some of these classifiers are multi-class whereas some of them are one-class. In this table, *QDC* stands for the two-class Quadratic Gaussian classifier, *MOGC(a,b)* represents the Mixture of Gaussians with *a* mixtures for building and *b* mixtures for the non-building class. The *MOGC* method performs comparable with *QDC* in low feature dimensions, however as the feature dimension increases, the method experiences the convergence problem. This is why some entries of the table are empty. *GOC_QDC* stands for the classifier obtained by intersecting (by Boolean and operation) the data description boundary of Gaussian one-class classifier with the quadratic Gaussian

(a)



(b)

Figure 6.1: Test image 1 and the ground truth extracted from this image. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The red pixels are examples of buildings, cyan pixels are examples of the non-building class and white pixels are the non-labeled parts.

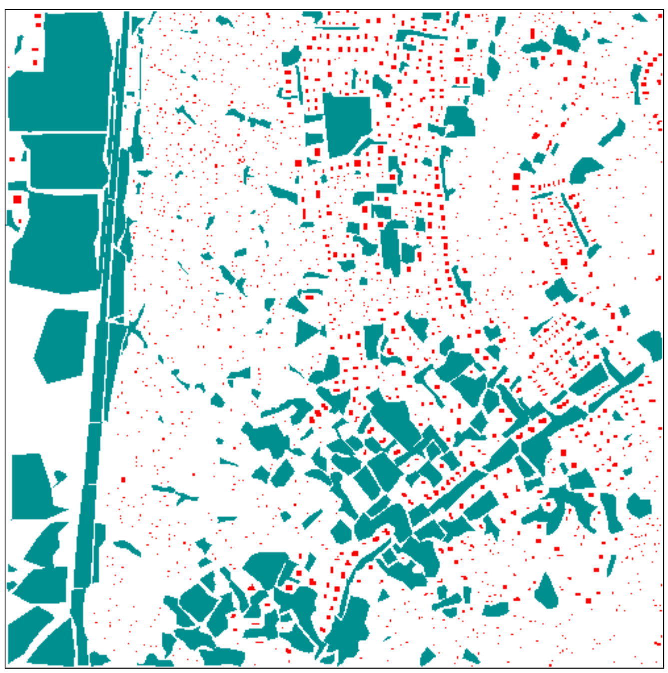


(a)

(b)

Figure 6.2: Test image 2 and the ground truth extracted from this image. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The red pixels are examples of buildings, cyan pixels are examples of the non-building class and white pixels are the non-labeled parts.

(a)



(b)

Figure 6.3: Test image 3 and the ground truth extracted from this image. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The red pixels are examples of buildings, cyan pixels are examples of the non-building class and white pixels are the non-labeled parts.

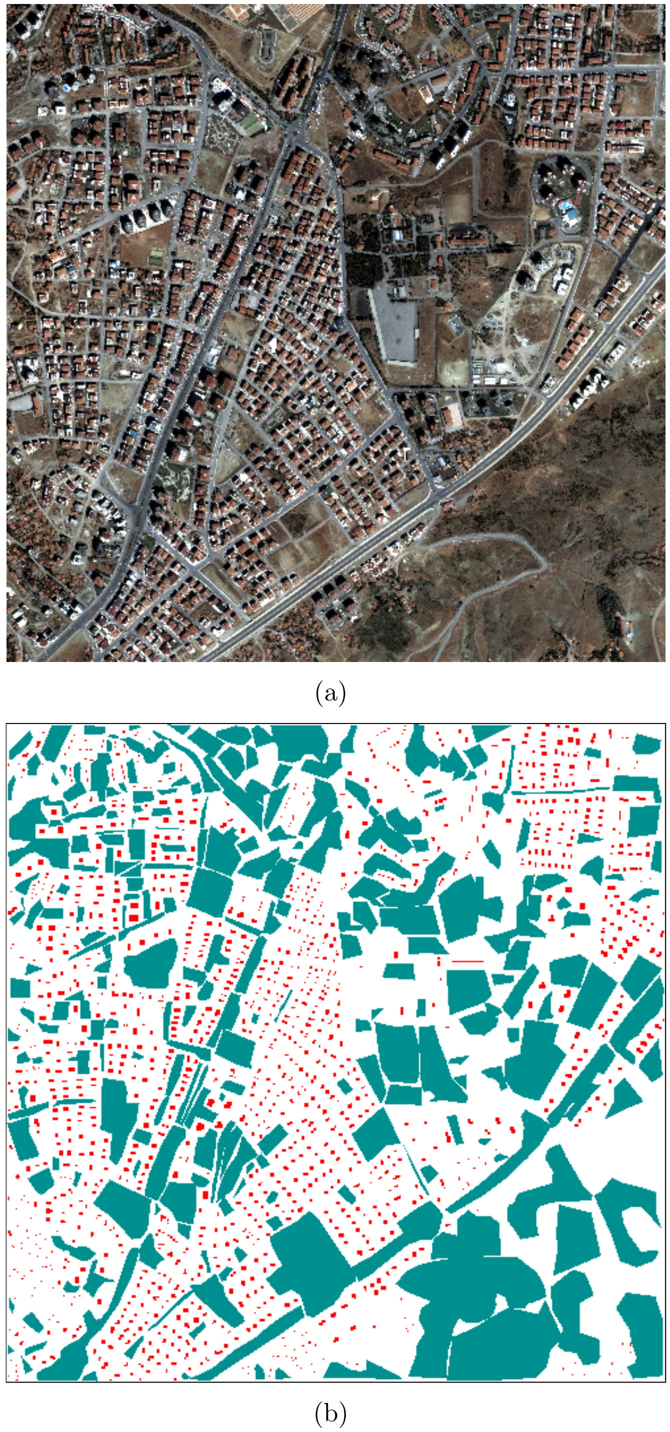


Figure 6.4: Test image 4 and the ground truth extracted from this image. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The red pixels are examples of buildings, cyan pixels are examples of the non-building class and white pixels are the non-labeled parts.

classifier. Even though the success rates for several feature combinations (e.g., F8 and F9 for QDC classification) were higher than the rates for others (e.g., F1), visual examination of the results showed that the success rates for pixel level ground truth did not always reflect the visual quality of the results. We decided to use the RGB features with both the one-class classifier and the two-class classifier since these features achieved a comparably high accuracy and a successful detection of individual buildings.

Table 6.2: Different feature combinations used for detecting buildings.

Feature Code	Features in the combination
F1	RGB (3 bands)
F2	Gabor 2. scale features' mean (1 band)
F3	Gabor 3. scale features' mean (1 band)
F4	F2 + F3 (2 bands)
F5	Morphological Opening Features (5 bands)
F6	Morphological Closing Features (5 bands)
F7	F5 + F6 (10 bands)
F8	F1 + F2 (4 bands)
F9	F1 + F3 (4 bands)
F10	F1 + F2 + F3 (5 bands)
F11	F1 + F5 (8 bands)
F12	F1 + F6 (8 bands)
F13	F1 + F5 + F6 (13 bands)
F14	F1 + F2 + F5 + F6 (14 bands)
F15	F1 + F2 + F3 + F5 + F6 (15 bands)

In building detection, we use the *PARZEN_QDC* classifier that is obtained by intersecting the one-class parzen data descriptor boundary with the classifier boundary of quadratic Gaussian classifier. Table 6.4 shows the performance of this classifier on the test data. Examples of detected buildings are shown in Figures 6.10 through 6.12 and in Figures 6.13 and 6.14.

6.2.2 Boat and Water Detection

For boat and water detection, manully labeled masks that sample image regions as water, boat and others were used. For each of these primitives, there exist

Table 6.3: Performance of different classifiers on different feature combinations in detecting buildings. The numbers are the percentage success rates. PARZEN_QDC outperforms other classifiers in most of the cases, despite the limited number of training samples used.

Feature Code	QDC	MOGC(2,2)	MOGC(3,3)	MOGC(6,3)	GOC_QDC	PARZEN_QDC
F1	94.57	94.65	94.33	94.45	94.68	**96.38**
F2	**86.09**	77.72	75.42	83.84	85.79	84.87
F3	**84.53**	72.54	68.97	65.52	84.51	83.85
F4	**84.95**	69.36	72.98	75.20	85.34	53.53
F5	**89.85**				77.35	41.43
F6	81.18				88.19	**90.13**
F7	88.90				85.89	**89.67**
F8	95.44	95.54	94.61	94.96	95.37	**95.74**
F9	94.76	94.74	93.91	94.59	94.77	**95.71**
F10	94.55	95.01	95.24	95.23	95.31	**95.86**
F11	**93.16**				88.01	92.90
F12	93.52				92.44	**94.21**
F13	92.56				91.70	**92.92**
F14	92.56				92.41	**94.03**
F15	93.40				92.28	**93.76**

Table 6.4: Confusion matrix for building detection on test images. Success rate is 96.38%.

		Detected	
		building	non-building
True	building	164,628	39,330
	non-building	36,521	1,856,357

one-class classifiers that learn the patterns of the corresponding class. Figure 6.5 shows an image that has harbor instances and the masks used for each primitive of harbor class.

Table 6.5 shows the training and testing data used to evaluate the detection of harbor primitives. Table 6.6 illustrates the accuracy on detecting harbor primitives.

Table 6.5: Training and testing data used in evaluating harbor primitives' detection.

Datasets	# boat pixels	# water pixels	# others pixels
Training	14,431	5,948	21,706
Testing	24,198	19,833	44,106

Table 6.6: Confusion matrix on the detection of harbor primitives. Success rate is 83.70%.

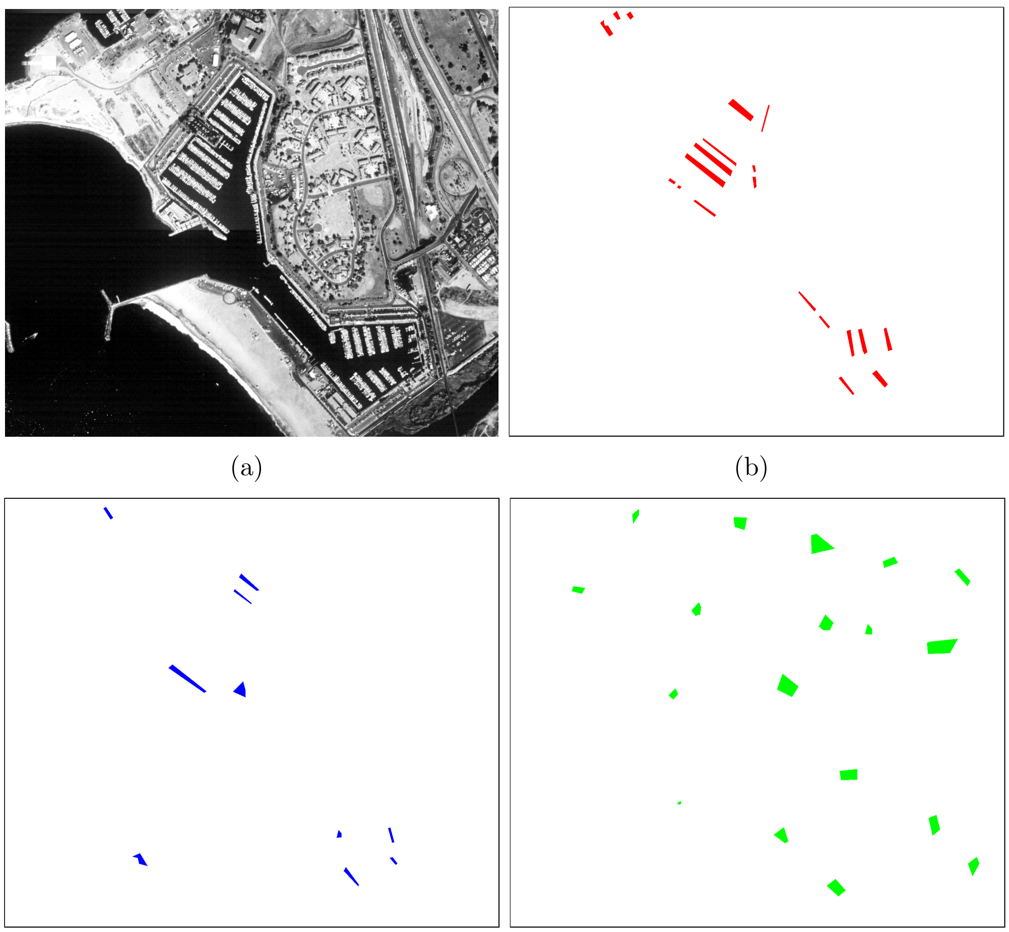
		Detected		
		boat	water	others
True	boat	20,566	41	3591
	water	0	19,173	660
	others	9,045	1,026	34,035

Table 6.7: Number of buildings used in the evaluation of classification performances.

Buildings in organized regions	# building in unorganized regions
3,020	7,614

Table 6.8: Number of windows used in the training of the decision tree classifier. There are totally 1600 windows in test images.

# of Organized	# of Unorganized	# of Fuzzy
242	290	49



(a) (b) (c) (d)

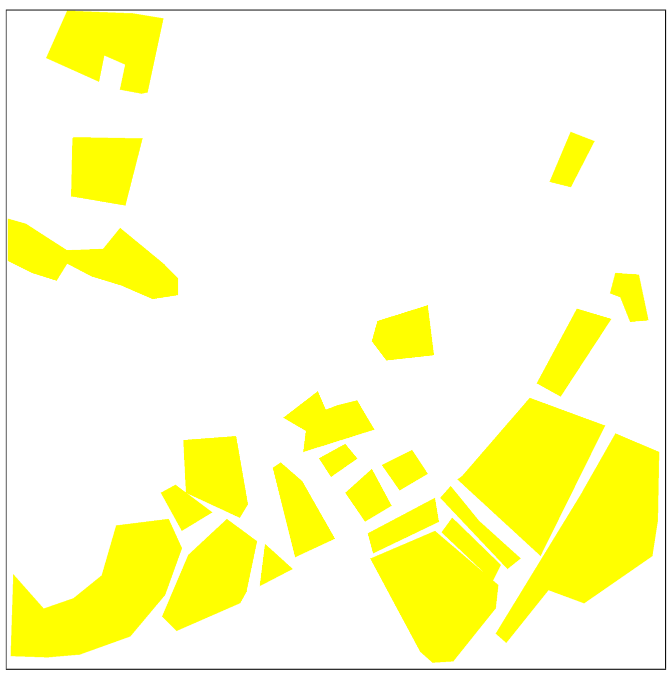
Figure 6.5: Example harbor image and masks used for detecting the primitives of harbor. (a) Example scene with harbor instances. (b) Mask for boat primitive, boat regions are shown in yellow. (c) Mask for water primitive, labeled water regions are shown in yellow. (d) Mask for others primitive, labeled others regions are shown in yellow.

(a)

(b)

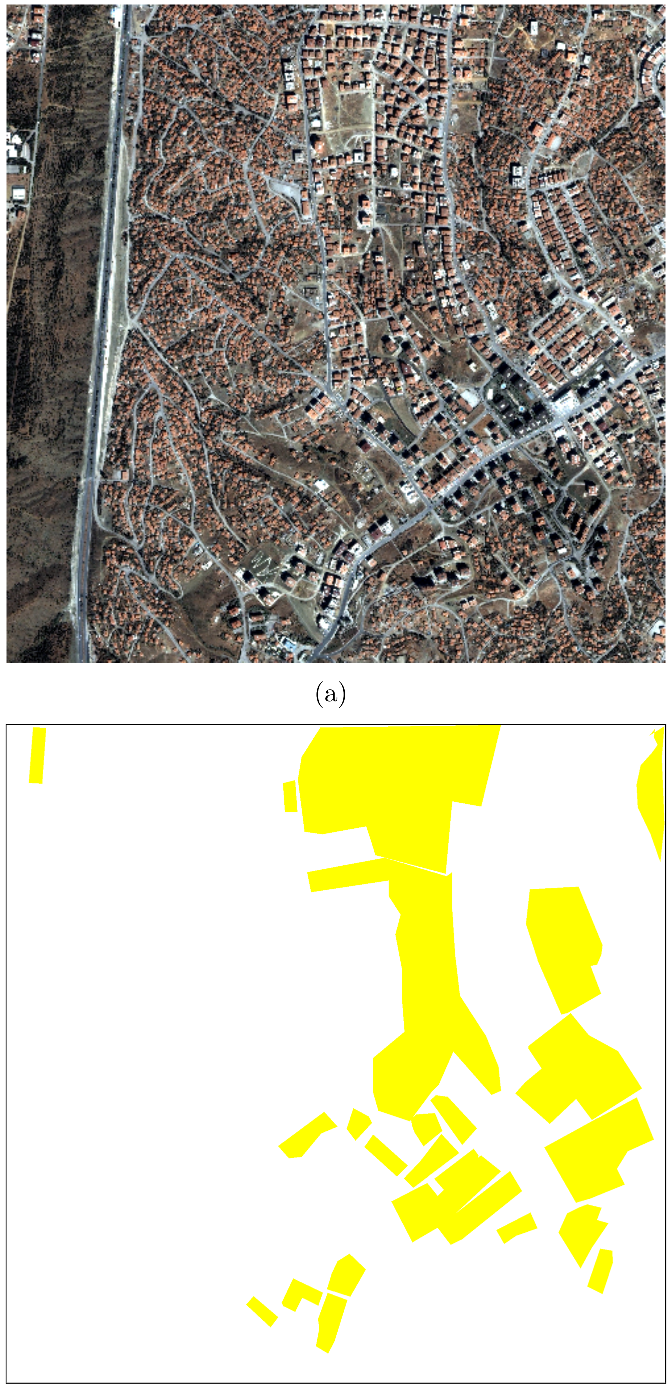
Figure 6.6: Test image 1 and the ground truth extracted from this image for settlement area detection. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The settlement areas inside yellow regions are extracted as organized settlement areas whereas the settlement areas inside white part are the unorganized regions.

(a)



(b)

Figure 6.7: Test image 2 and the ground truth extracted from this image for settlement area detection. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The settlement areas inside yellow regions are extracted as organized settlement areas whereas the settlement areas inside white part are the unorganized regions.

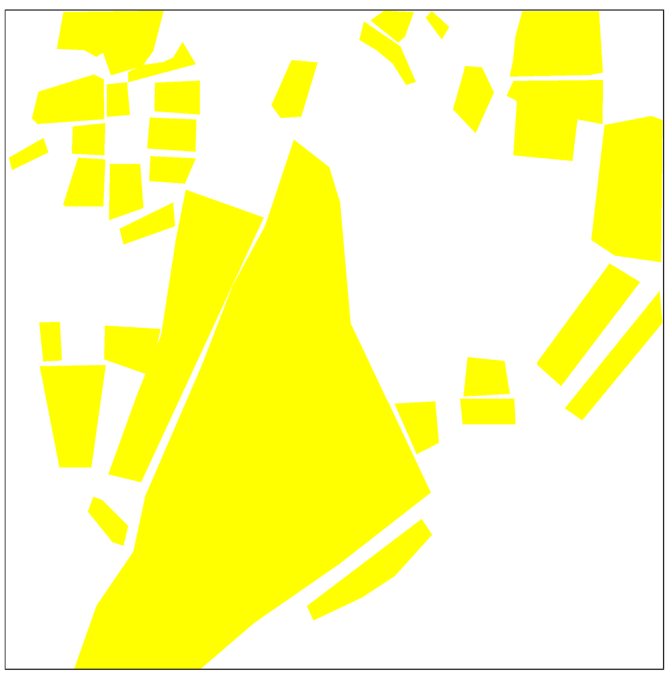


(a)

(b)

Figure 6.8: Test image 3 and the ground truth extracted from this image for settlement area detection. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The settlement areas inside yellow regions are extracted as organized settlement areas whereas the settlement areas inside white part are the unorganized regions.

(a)



(b)

Figure 6.9: Test image 1 and the ground truth extracted from this image for settlement area detection. (a) RGB bands of one of the test images used in settlement area detection (b) The ground truth extracted from the image in (a). The settlement areas inside yellow regions are extracted as organized settlement areas whereas the settlement areas inside white part are the unorganized regions.

6.3 Scene Classification Using Statistical Approach

In this part, we evaluated the performances of co-occurrence-based contrast and chi-square features and wedge-based Fourier features when classified with a decision tree. For each of the test images, we manually extracted masks that show the boundaries of organized settlement areas. For the training of the decision tree, we randomly chosed some image windows, and automatically learned their labels (organized, unorganized or fuzzy for windows that lie on the boundaries of different settlement types) from the masks that show the boundaries of organized settlement areas (details can be found in Table 6.8). The classification result of image regions (100×100 windows, each) consists of labels as organized/unorganized for each window. To evaluate the classification accuracy, we determined the centroids of each of the buildings, and extracted the labels each building centroid has according to in which window they lie in. Then, the true labels of each building is taken from the manually extracted mask, and compared with the classification map. The result is a confusion matrix that shows the number of buildings classified as organized/unorganized, for each test image. Figure 6.6 through Figure 6.9 shows the test images with corresponding masks. Table 6.7 shows the total number of buildings used in evaluating the classification performances.

We performed classification using two spatial domain and frequency domain features. Common to all of these features, errors were mostly caused by different densities of buildings in different neighborhoods. In particular, some neighborhoods were incorrectly classified as unorganized when they contained a low density of regularly placed buildings. Some errors were also due to sub-windows that were on the boundary between organized and unorganized neighborhoods. All these cases are the disadvantages of window-based approaches, therefore as an alternative, we developed the structural approach explained in Chapter 5.

Example classification results are given in Figures 6.10 through 6.12. Visual evaluation of the results showed that most of the neighborhoods were classified correctly. Most of the errors occured in the classification of test image 4, since in this image, the buildings are far away from each other and their distribution in a window is not sufficient to determine their regularity information. Results for classification of each building using different features are given in Tables 6.9 through 6.11.

Table 6.9: Confusion matrix for classifying windows of the test images using χ^2-based spatial domain features as organized/unorganized. Success rate is 81.54%.

		Detected	
		organized	unorganized
True	organized	2,034	986
	unorganized	977	6,637

Table 6.10: Confusion matrix for classifying windows of the test images using contrast-based spatial domain features as organized/unorganized. Success rate is 72.90%.

		Detected	
		organized	unorganized
True	organized	1,925	1,095
	unorganized	1,787	5,827

Table 6.11: Confusion matrix for classifying windows of the test images using wedge-based frequency domain features as organized/unorganized. Success rate is 76.86%.

		Detected	
		organized	unorganized
True	organized	2,040	980
	unorganized	1,482	6,132

Since we do not have enough data on harbor, we could not evaluate the performance of the proposed methods on harbor classification.

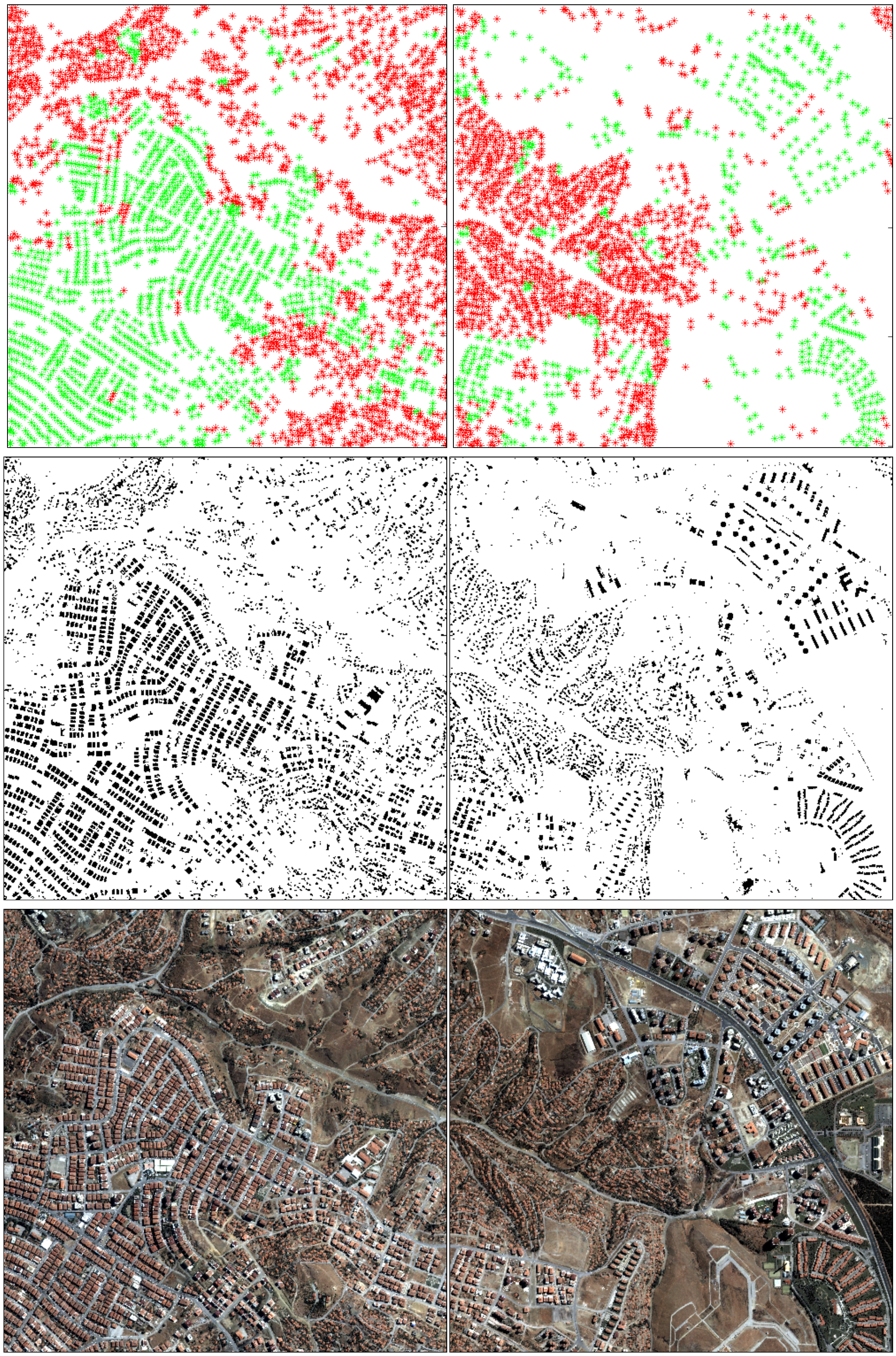


Figure 6.10: Example scene classification results for test images 1 and 2. The left column shows the original data, the middle column shows the detected buildings, and the right column shows the neighborhoods classified using the χ^2-based spatial domain features. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

Figure 6.11: Example scene classification results for test images 1 and 2. The left column shows the original data, the middle column shows the detected buildings, and the right column shows the neighborhoods classified using the contrast-based spatial domain features. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

Figure 6.12: Example scene classification results for test images 1 and 2. The left column shows the original data, the middle column shows the detected buildings, and the right column shows the neighborhoods classified using the fourier domain features. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

6.4 Scene Classification Using Structural Approach

To evaluate the performance of labeling building groups as organized or unorganized, the neighborhood masks shown in Figures 6.6 through 6.9 are used. Then, given all labels resulting from the procedure described in Section 5.3, the labels of buildings are compared to the ground truth to check whether a building automatically labeled as belonging to an organized/unorganized neighborhood is inside a mask manually labeled as organized/unorganized. Table 6.12 shows the resulting confusion matrix (success rate was 81.60%). (Note: The total number of buildings in Table 6.12 is less than the one given in the experiments of statistical approach. We used the same primitive maps and considered the same centroids in the experiments, however in the structural approach, to consider a cluster as organized/unorganized, this cluster should have at least three primitives (since otherwise angle distributions cannot be analyzed). Therefore, the clusters with less than three primitives are ignored in analyzing the performance of the method and this is why different numbers are seen in the tables.)

Table 6.12: Confusion matrix for labeling neighborhoods of the test images as organized/unorganized (graph-based approach). Success rate is 81.60%.

		Detected	
		organized	unorganized
True	organized	2,073	458
	unorganized	949	4,167

Example classification results are given in Figures 6.13 and 6.14. It can be seen that most of the buildings were correctly labeled. Large organized groups were found in the upper part of the first image and lower left part of the second image. Many unorganized building groups were also classified correctly. Some organized building groups in the lower left part of the first image and the upper left part of the second image were wrongly labeled as unorganized. Most of the errors were caused by the limitations of the graph clustering procedure using the minimum spanning tree. Since the only edge weight used to construct the minimum spanning tree and the subsequent clustering was the distance between neighboring nodes (buildings), some unorganized components that are too close to organized components affected the latter during clustering.

In this methodology, to obtain the clusters, some edges of the minimum spanning tree are removed, and as we explained in Section 5.2, the edges that are removed are selected as the ones that are at least 1.5 times the average edge length in the minimum spanning tree. This cut value is obtained after some experiments and Figure 6.15 presents the clustering accuracy obtained for different cut values.

6.5 Comparison With The Existing Methods

In order to support the efficiency of the methods we proposed, in this part we apply some literature work that is accepted to be successful on similar applications.

Firsly, we considered the work by Bhagavathy and Manjunath [10]. Similar to their work, we implemented a two-layered framework, where the first layer learns the constituent "texture elements" of a texture motif and the second learns the spatial distribution of the elements. We used Gabor texture features as they did (the specifications of the Gabor filter bank used for extracting the features are the same), and applied the same sampling methodology in training the first layer of the framework. In this part, to learn the texture element model they used a Gaussian mixture model (GMM) framework which is rotation invariant by the use of an orientation-normalized GMM [50]. Differently, we applied a K-means clustering process. Since we did not use the orientation-normalized GMM, we used two approaches to eliminate the rotation invariance problem. In the first approach, as the texture features, we used the mean of considered orientations for each scale of the Gabor texture filter (corresponds to a feature set of 5 bands since scale number is 5 and orientation number is 6 in the Gabor texture filter bank). In the second approach, we considered all the scales and orientations and

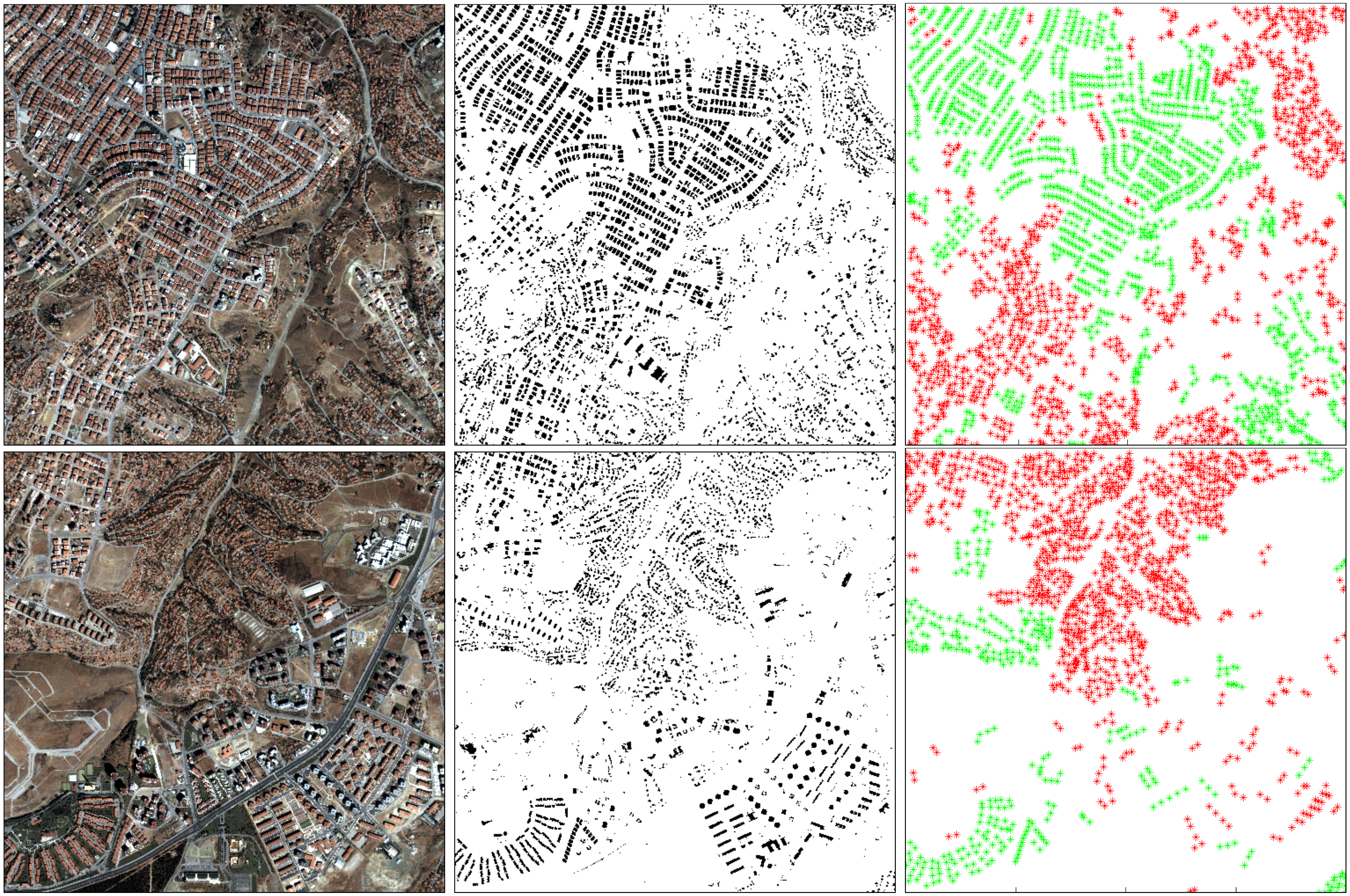


Figure 6.13: Scene classification results for two of the Ikonos test scenes. The left column shows the original data, the middle column shows the detected buildings, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

Figure 6.14: Scene classification results for two of the Ikonos test scenes. The left column shows the original data, the middle column shows the detected buildings, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

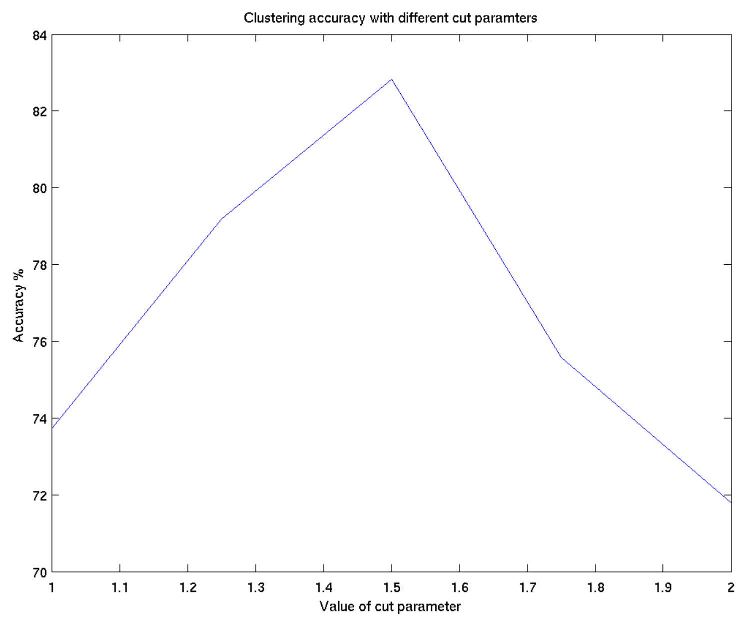


Figure 6.15: Scene clustering performance for different cut values. Best result is obtained for 1.5.

for each image we used 30 bands as the textural features.

In the second layer, to learn the spatial distribution of texture elements, they applied GMM on the spatial histogram of texture element labels. We described the spatial distribution similarly, with spatial histograms formed in the same way and applied GMM on these features. When the feature set was composed of Gabor filters of 5 bands, we applied 3 mixtures for the organized class and 2 mixtures for the unorganized class. When Gabor features of 30 bands were used as the feature set, we applied 2 mixtures for the organized class and 1 mixture for the unorganized class (since divergence was a problem, we could not apply the same procedure as we applied for features composed of 5 bands). For texture element components, we used the values by which they obtained their best results (3 components for texture elements). In determining the window size for spatial histogram, we took 101×101 pixels for each window as in our method. We performed the experiments on urbanization modeling application and with this methodology, we differentiated the organized settlement areas from the unorganized ones. In the experiments, we applied the same procedure that we applied in our statistical approach. The training and testing windows we considered in the statistical approach were used in training and testing the GMM classifier. The performance evaluation method is also the same as we applied to our methods, by counting the number of buildings correctly classified according to its neighborhood type (organized or unorganized). Table 6.13 and Table 6.14 present the performance of this method when 5 and 30 bands of Gabor texture filter outputs are considered as the textural features, respectively. Figures 6.16 and 6.17 show the results when 5 bands are considered. Figures 6.18 and 6.19 illustrate the results when 30 bands are considered.

Besides the work presented in [10], we implemented the work of Dell'Acqua *et al.* in [19], where informal settlement area analysis is performed. In this work, Fuzzy Artmap neural classifier is used to discriminate the urban and desert areas by co-occurrence matrix-based textural features. It is stated in the work that best performances were obtained with the mean, second moment, variance combination as the feature set and when features are computed using a 21×21

Table 6.13: Confusion matrix for labeling neighborhoods of the test images as organized/unorganized, with a method similar to the one proposed in [10]. 5 bands of Gabor texture filter outputs are considered as the textural features. Success rate is 64.82%.

		Detected	
		organized	unorganized
True	organized	2,039	981
	unorganized	2,760	4,854

Table 6.14: Confusion matrix for labeling neighborhoods of the test images as organized/unorganized, with a method similar to the one proposed in [10]. 30 bands of Gabor texture filter outputs are considered as the textural features. Success rate is 70.02%.

		Detected	
		organized	unorganized
True	organized	894	2,126
	unorganized	1,062	6,552

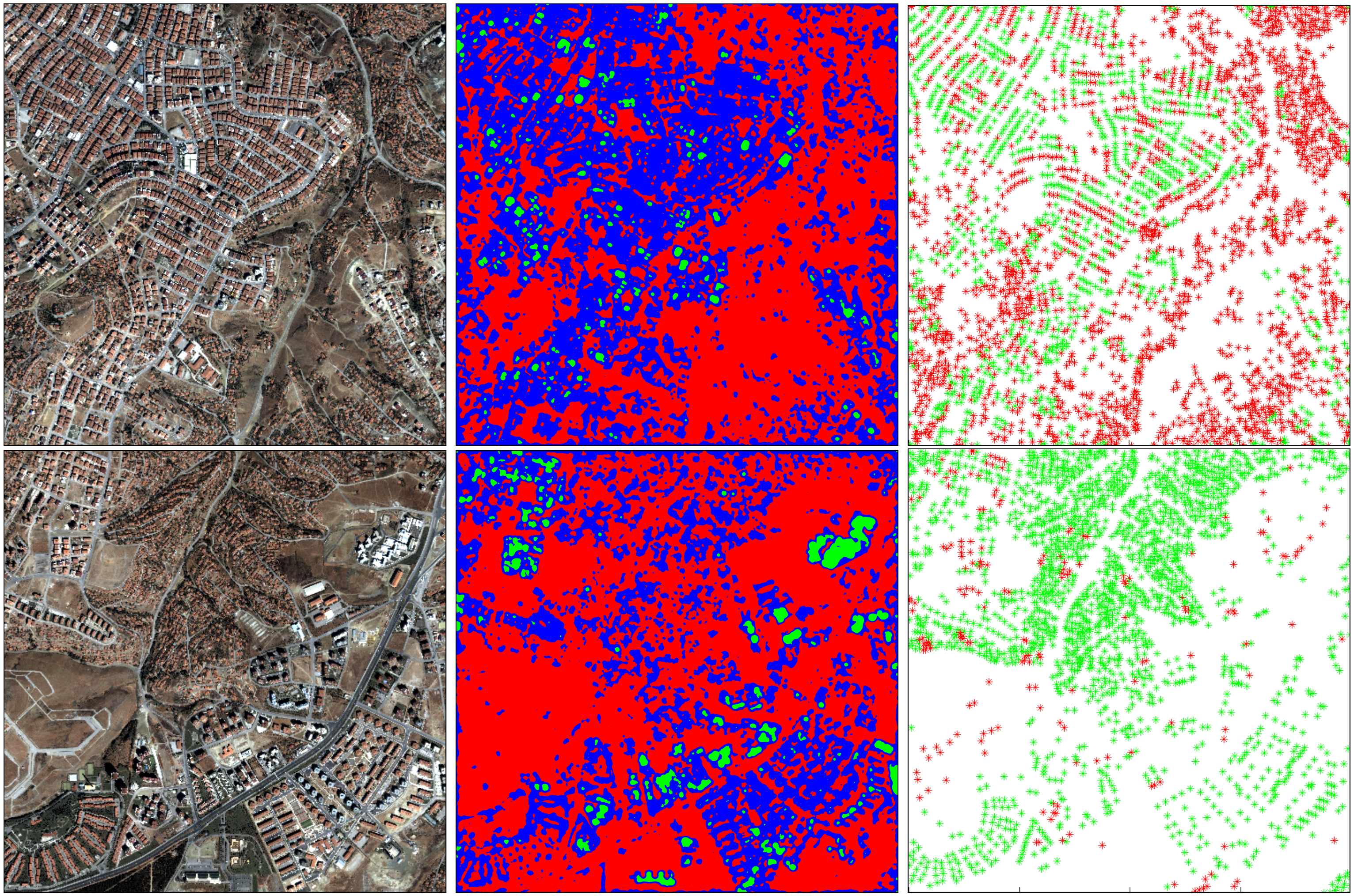


Figure 6.16: Scene clustering results for two of the Ikonos test scenes, with the method similar to the one proposed in [10] and when 5 bands are considered as the Gabor texture features. The left column shows the original data, the middle column shows the texture elements after K-means clustering, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

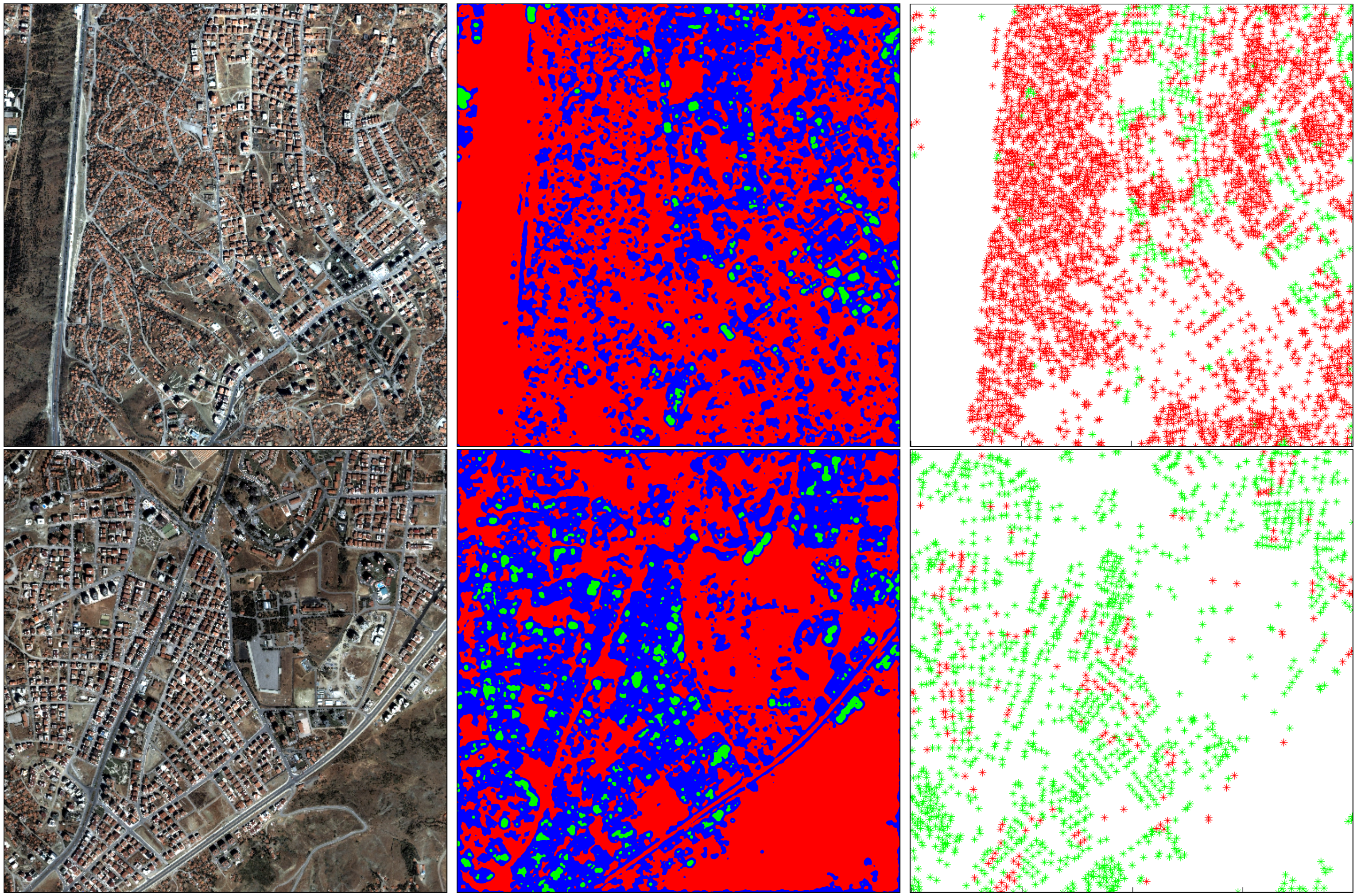


Figure 6.17: Scene clustering results for two of the Ikonos test scenes, with the method similar to the one proposed in [10] and when 5 bands are considered as the Gabor texture features. The left column shows the original data, the middle column shows the texture elements after K-means clustering, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

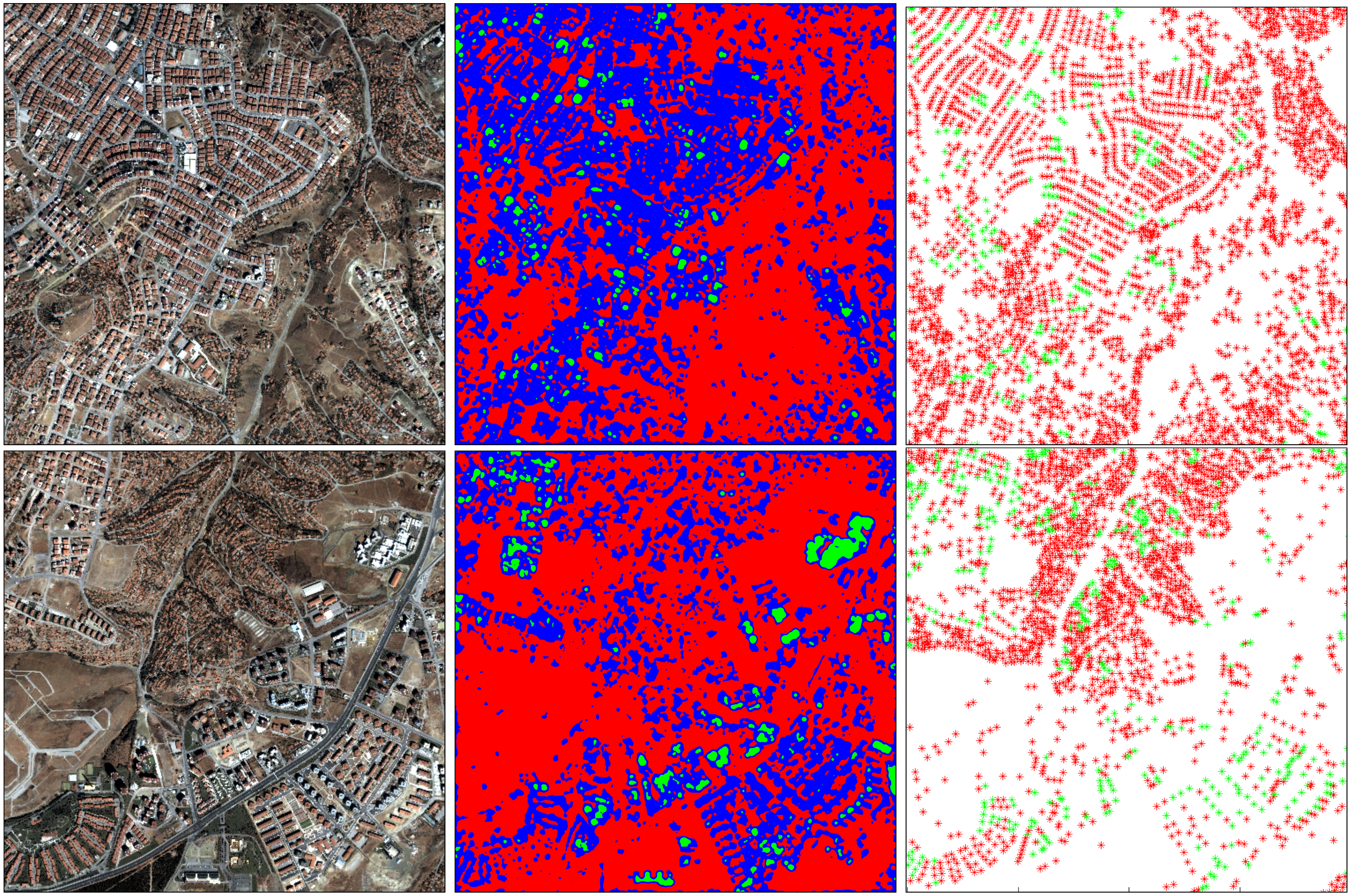


Figure 6.18: Scene clustering results for two of the Ikonos test scenes, with the method similar to the one proposed in [10] and when 30 bands are considered as the Gabor texture features. The left column shows the original data, the middle column shows the texture elements after K-means clustering, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

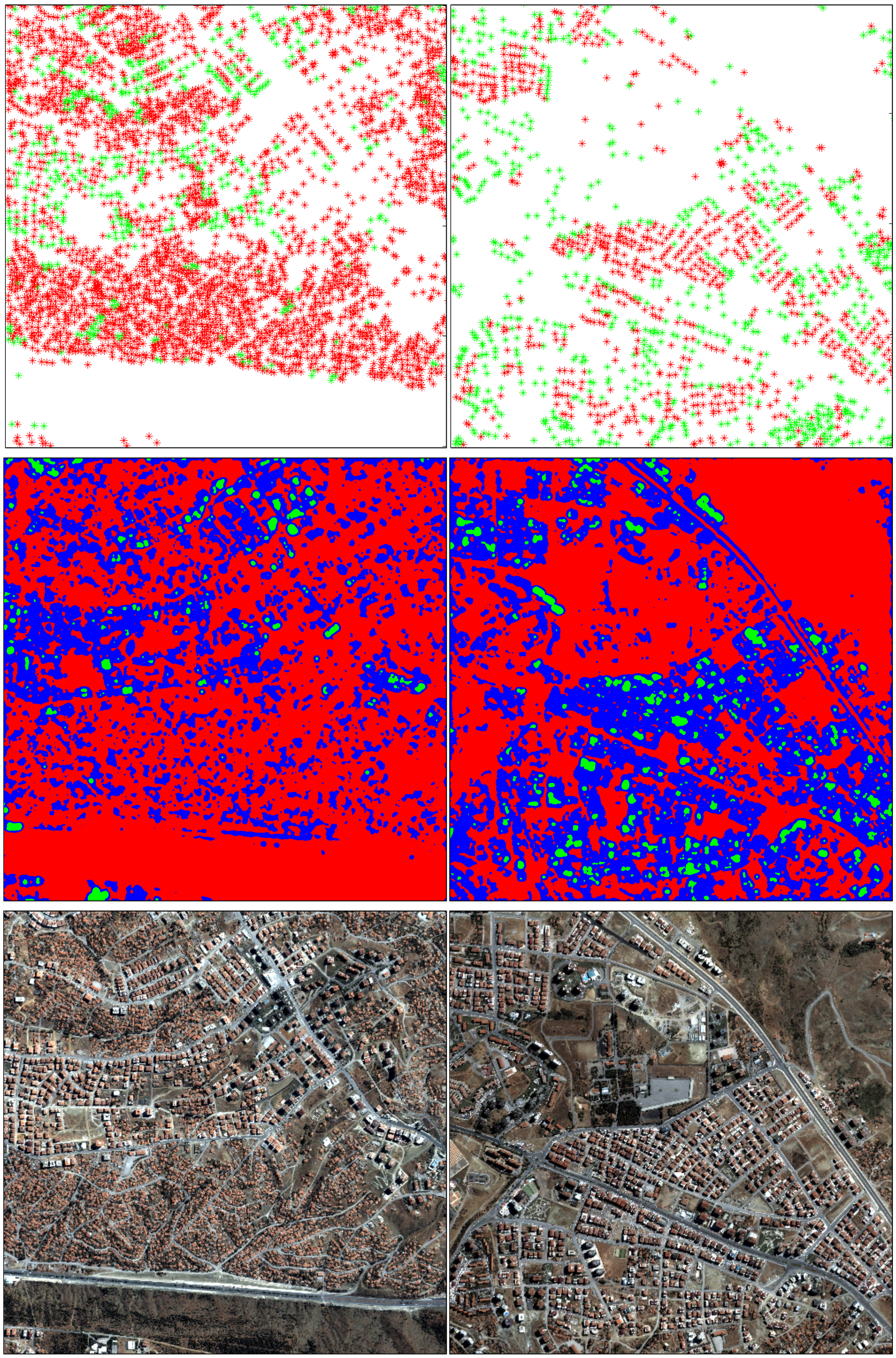


Figure 6.19: Scene clustering results for two of the Ikonos test scenes, with the method similar to the one proposed in [10] and when 30 bands are considered as the Gabor texture features. The left column shows the original data, the middle column shows the texture elements after K-means clustering, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

window. We considered the same features, and used the two-layered framework that we employed in implementing the work in [10]. To evaluate the performance, we applied the same procedure as we performed on our method and counted the correctly classified primitives. Table 6.15 illustrates the results and Figures 6.20 and 6.21 demonstrate the clustering results on the test set.

Table 6.15: Confusion matrix for labeling neighborhoods of the test images as organized/unorganized, with a method similar to the one proposed in [19]. Success rate is 72.48%.

		Detected	
		organized	unorganized
True	organized	1,846	1,174
	unorganized	1,753	5,861

The classification results of both of the approaches illustrate that these approaches cannot capture the neighborhood information. When one examines the classification results, it is clear that neighborhoods cannot be identified with a histogram based approach. The obtained success rates even do not reflect this fact, since these approaches mostly classify regions into just one class (for instance unorganized) and visual inspection suggests that the results are even worse that the given success rates. These results show that our method performs considerably successfully and gives promising results in terms of the applicability to other problems and its robustness.

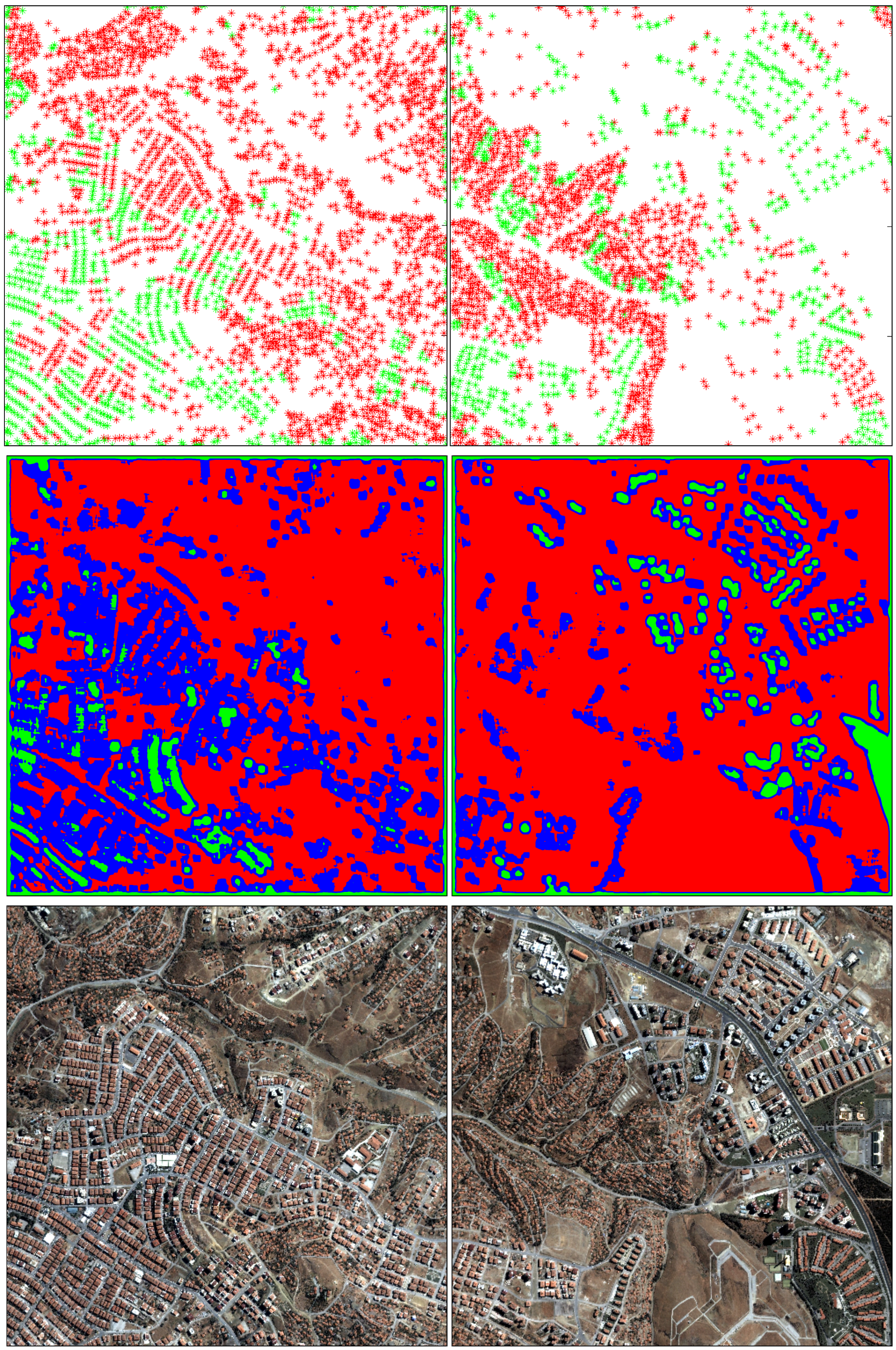


Figure 6.20: Scene clustering results for two of the Ikonos test scenes, with the method similar to the one proposed in [19]. The left column shows the original data, the middle column shows the texture elements after K-means clustering, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

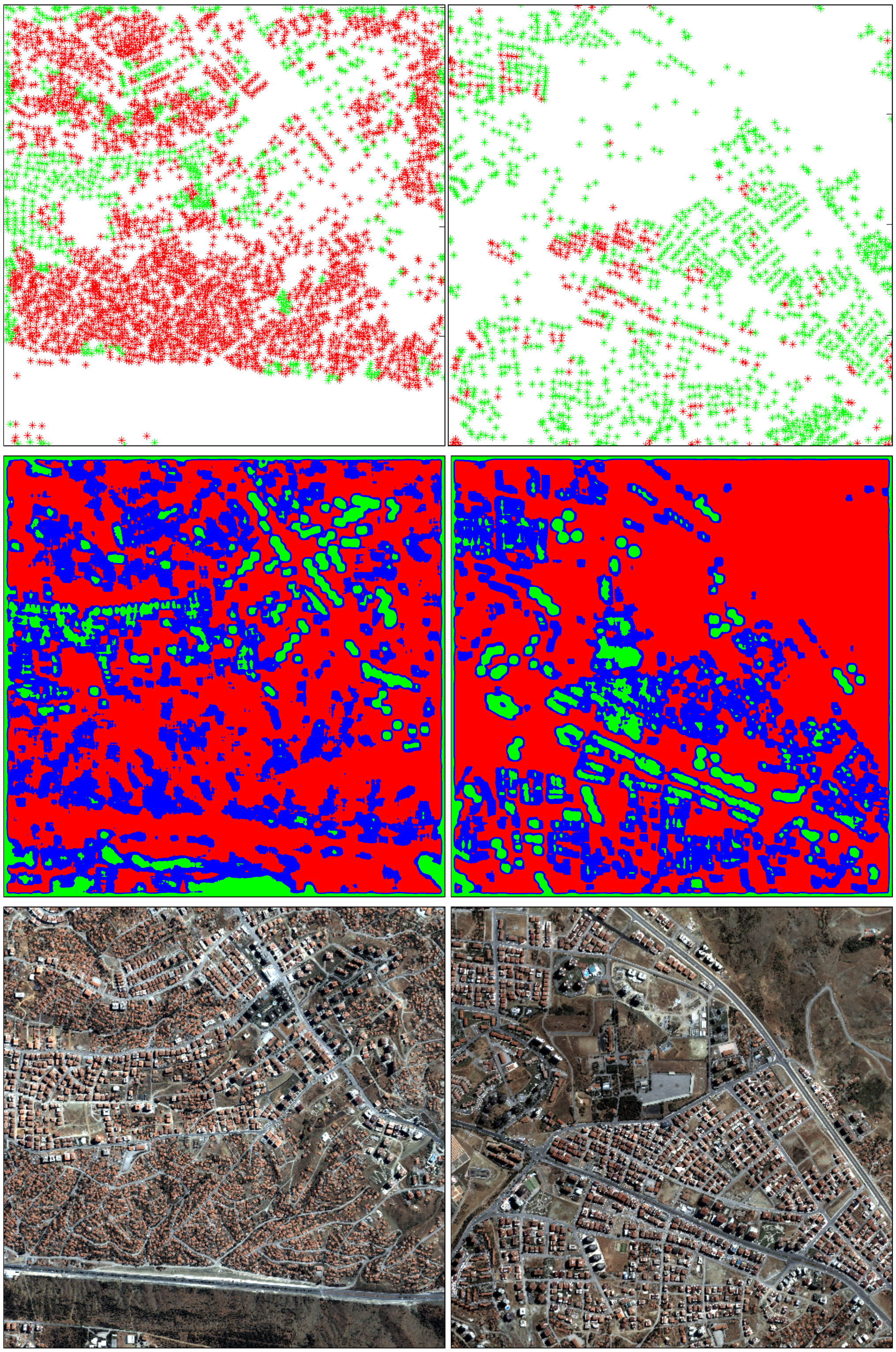


Figure 6.21: Scene clustering results for two of the Ikonos test scenes, with the method similar to the one proposed in [19]. The left column shows the original data, the middle column shows the texture elements after K-means clustering, and the right column shows the results of clustering neighborhoods. Buildings that belong to neighborhoods classified as organized are shown as green and buildings in unorganized neighborhoods are shown in red.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we proposed methodologies to analyze remote sensing images and detect complex geospatial structures in terms of the regularity and irregularity of simpler primitives. We mentioned the importance of spatial arrangement information of geospatial structures and claimed that without integrating this information, it is impossible to detect complex geospatial objects in whose definition there exists specific alignment information (for instance, the alignment of rows of boats and water in an harbor). In modeling the geospatial objects, we proposed that some geospatial object primitives could be used as the unit elements of textural analysis methods, and this analysis can give useful information in terms of the characterization of spatial patterns of these unit elements. We developed such methodology which first detects the objects primitives and then characterizes the spatial patterns in terms of the statistical and structural texture analysis methods. We expressed our belief that such kind of a system can give useful information for diverse application fiels, and examplified this belief on the detection of harbor instances and measuring the degree of urbanization in a settlement area.

The proposed methodology firstly detects the primitives of interest. In primitive detection, we discussed some important features of images, and illustrated the use of Gabor texture features, morphological profile features and multi-spectral information for accurate detection. We utilized the advantages of both the one-class and multi-class classification approaches and suggested fine tuning the classifier boundary of a one-class classifier with the use of multi-class classifier boundary. With this approach, we argued that we can eliminate the disadvantages of each classifier approach; where the disadvantage of one-class classifiers is the possible high rate in false alarms due to the strict data description boundaries and the disadvantages of multi-class classifiers are the complexity in the outlier class data and very probable insufficient sampling of outliers that result in undefined regions in the feature space. In this work, we gave an example of intersecting the boundary of a parzen density estimation-based classifier with the boundary of a quadratic gaussian classifier. On the usage of different feature types, we concluded that in primitive detection although some feature combinations can give lower error rates in some other combinations, the error rates do not always have a corresponding visual inspection, and it may be better to use feature combinations with higher error rates.

In the proposed methodology, the second step in high-level object detection is the analysis of spatial patterns of primitives. In this step, we proposed a statistical and a structural approach. In the statistical approach, the spatial information we are interested in corresponds to the primitives' repetitiveness and periodicity at particular orientations, and we achieved this task by the use of co-occurrence based spatial domain features and Fourier spectrum-based frequency domain features. What we observed when we used our rotation invariant co-occurrence based features is that if a neighborhood contains a regular arrangement of primitives along a specific direction, the features at this orientaton exhibit a periodic structure. We used this observation in classifying image regions into regular/irregular with a decision tree. Knowing that Fourier spectrum-based features can capture texture periodicity and directionality, we computed the ring-based and wedge-based features for each pattern, respectively. What we observed when we used this features was, the features had significant peaks if the neighborhood of interest had periodic and directional patterns.

In the structural spatial analysis approach, we used a graph-theoretic modeling schema which groups the primitives into clusters of primitives with similar spatial arrangements. In this representation, the primitives form the nodes of the graph where the neighborhood information is obtained through Voronoi tessellation of the image scene. We performed the clustering of graph by thresholding its minimum spanning tree and the resulting clusters were classified as regular or irregular by examining the distributions of the angles between neighboring nodes. What made this approach valuable and successful for us was its position, scale and rotation invariance nature, with applicability to application areas other than urbanization modeling.

In the experiments, we evaluated the performances of primitive detectors and spatial pattern analysis methods. In building detection, we evaluated the performances of different feature combinations and different classifiers. What we observed was, although some feature combinations can perform better in terms of the error rates, visual inspection of these results showed that the resulting detection could reflect the nature of data better with some other feature combinations that result in higher error rates. Besides, we observed that classification by the classifiers obtained by intersecting the boundaries of one-class and multi-class classifiers can give successful results. The *PARZEN_QDC* classifier we used achieved 96.38% success in detecting the buildings of the test images. Similar procedure was followed for the detection of harbor primitives. The results of classifying scenes with the statistical texture model showed that the proposed textural features can capture the spatial information of the primitives. We observed the disadvantages of using window-based approaches in this part, because some windows can be on the boundaries between organized and unorganized neighborhoods and there can be misclassifications in such cases. The performances of the contrast-based, χ^2-based and Fourier-based features were comparable. In the structural scene classification, a success rate of 81.60% was obtained, and most of the settlement areas were correctly classified as organized or unorganized. Most of the misclassifications were in regions where organized components were too close to unorganized components. We believe that such cases can be overcome by the use of new edge weights in addition to distance.

7.2 Future Work

In the proposed work, we could not evaluate the performances of the methodology on harbor instances since we do not have enough data and the data in hand do not include the features we would like to use (such as the multi-spectral information). Therefore, as the next step, we will evaluate the method on harbors better. Besides, in the statistical texture model, the features were obtained by summing the co-occurrence features at each orientation. We aim to find a better solution that represents the periodicity information of the data in this part. In the structural approach, in detecting the edges to be removed from the minimum spanning tree of the graph, we used the distance as the edge weight, and we plan to add more metrics for edge weights, such as the road information.

We believe that the methodologies proposed in this work can be used in the detection of various other complex geospatial objects, therefore the next step will be to apply the methods on new geospatial objects such as golf courses, parking lots and specific crop regions, or forestry lands.

Bibliography

[1] H. G. Akcay and S. Aksoy. Morphological segmentation of urban structures. In *Proceedings of 4th IEEE GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, Paris, France, April 11-13, 2007.

[2] S. Aksoy. Textural features for content-based image database retrieval. Master's thesis, University of Washington, 1998.

[3] S. Aksoy. Spatial techniques for image classification. In C.H. Chen, editor, *Signal and Image Processing for Remote Sensing,*. Taylor & Francis, 2006.

[4] S. Aksoy and E. Dogrusoz. Modeling urbanization using spatial building patterns. In *Proceedings of 4th IAPR International Workshop on Pattern Recognition in Remote Sensing*, Hong Kong, August 20, 2006.

[5] P. Baldi and K. Hornik. Neural networks and principal component analysis: learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.

[6] E.P. Baltsavias, A. Gruen, and L. V. Gool. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III).* A. A. Balkema, 2001.

[7] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson. Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Transactions on Geoscience and Remote Sensing*, 43(3):480–491, March 2005.

[8] S. Berberoglu, C. D. Lloyd, P. M. Atkinson, and P. J. Curran. The integration of spectral and textural information using neural networks for land cover

mapping in the mediterranean. *Computers & Geosciences*, 26(4):385–396, May 2000.

[9] S. Bhagavathy. *Modeling and Detection of Geospatial Objects Using Texture Motifs*. PhD thesis, University of California, Santa Barbara, CA, December 2005.

[10] S. Bhagavathy and B.S. Manjunath. Modeling and detection of geospatial objects using texture motifs. *IEEE Transactions on Geoscience and Remote Sensing*, 44(12):3706–3715, December 2006.

[11] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, November 1995.

[12] C.M. Bishop. Novelty detection and neural network validation. *IEEE Proceedings of Vision, Image and Signal Processing*, 141(4):217–222, 1994.

[13] J.R. Carr. Spectral and textural classification of single and multiple band digital images. *Computers & Geosciences*, 22(8):849–865, 1996.

[14] J. C.-W. Chan, R. S. DeFries, X. Zhan, C. Huang, and J. R. G. Townshend. Texture features for change detection at 250m resolution-an application of machine learning feature subset selection. *Proceedings of the IEEE*, VII:3060–3062, 2000.

[15] J. Chanussot, M. Fauvel, and J.A. Benediktsson. Classification of remote sensing images from urban areas using a fuzzy possibilistic model. *IEEE Transactions on Geoscience and Remote Sensing*, 3(1):40–44, 2006.

[16] R. W. Conners and C. A. Harlow. Toward a structural textural analyzer based on statistical methods. *Computer Graphics and Image Processing*, 12:224–256, March 1980.

[17] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, 1990.

[18] F. Dell'Acqua and P. Gamba. Texture-based characterization of urban environments on satellite sar images. *IEEE Transactions on Geoscience and Remote Sensing*, 41(1):153–159, January 2003.

[19] F. Dell'Acqua, M. Stasolla, and P. Gamba. Unstructured human settlement mapping with SAR sensors. In *Proceedings of IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2006)*, Denver, Colorado, August 2006.

[20] X. Descombes, M. Sigelle, and F. Preteux. Estimating Gaussian Markov random field parameters in a nonstationary framework: application to remote sensing imaging. *IEEE Transactions on Image Processing*, 8(4):490–503, April 1999.

[21] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley Interscience, 2000.

[22] S. S. Durbha and R. L. King. Knowledge mining in earth observation data archives: a domain ontology perspective. *Proceedings of the IEEE International Conference Geoscience and Remote Sensing Symposium*, 1, September 2004.

[23] G. Fumera, F. Roli, and G. Giacinto. Reject option with multiple thresholds. *Pattern Recognition*, 33(12):2099–2101, 2000.

[24] D. Gokalp and S. Aksoy. Scene classification using bag-of-regions representations. In *Proceedings of 4th IEEE International Conference on Computer Vision and Pattern Recognition*, Minneapolis, Minnesota, June 23, 2007.

[25] A. Gruen, P. Agouris, and O. Kuebler. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (I)*. Birkhauser Boston, 1995.

[26] A. Gruen, E.P. Baltsavias, and O. Henricsson. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*. Birkhauser Verlag, 1997.

[27] Q. Guo, M. Kelly, and C. H. Graham. Support vector machines for predicting distribution of sudden oak death in california. *Ecological Modelling*, 182(1):75–90, 2005.

[28] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, May 1979.

[29] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621, November 1973.

[30] R. M. Haralick and L. G. Shapiro. Glossary of computer vision terms. *Pattern Recognition*, 24(1):69–93, January 1991.

[31] J. Hertz, A. Krogh, and R. Palmer. *Introduction to the theory of neural computation*. Addison-Wesley, 1991.

[32] A. Huertas and R. Nevatia. Detecting buildings in aerial images. *Computer Vision, Graphics, and Image Processing*, 41(2):131–152, February 1988.

[33] R.B. Irvin and D.M. McKeown. Methods for exploiting the relationship between buildings and theirshadows in aerial imagery. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1564–1575, November 1989.

[34] N. Japkowicz, C. Myers, and M. A. Gluck. A novelty detection approach to classification. In *International Joint Conference on Artificial Intelligence*, pages 518–523, 1995.

[35] V. Karathanassi, C. Iossifidis, and D. Rokos. A texture-based classification method for classifying built areas according to their density. *International Journal of Remote Sensing*, 21(9):1807–1823, September 2000.

[36] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, December 2000.

[37] C. Lai, D.M.J. Tax, R.P.W. Duin, E. Pekalska, and P. Paclk. On combining oneclass classifiers for image database retrieval, June 2002.

[38] D. A. Landgrebe. *Signal Theory Methods in Multispectral Remote Sensing*. John Willey & Sons, 2003.

[39] Ivan Laptev, Helmut Mayer, Tony Lindeberg, Wolfgang Eckstein, Carsten Steger, and Albert Baumgartner. Automatic extraction of roads from aerial images based on scale space and snakes. *Machine Vision and Applications*, 12(1):23–31, 2000.

[40] S. Leguizamon. Characterization of texture in remotely sensed images by using the wavelet transform. *Proceedings of the International Symposium on High Mountain Remote Sensing Cartography*, pages 129–139, 1996.

[41] H.-C. Lin, L.-L. Wang, and S.-N. Yang. Extracting periodicity of a regular texture based on autocorrelation functions. *Pattern Recognition Letters*, 18(5):433–443, May 1997.

[42] F. Liu and R. W. Picard. Periodicity, directionality, and randomness: Wold features for image modeling and retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):722–733, July 1996.

[43] L. M. Manevitz and M. Yousef. One-class svms for document classification. *Journal of Machine Learning Research*, 2:139–154, 2001.

[44] B. S. Manjunath and W.Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, August 1996.

[45] T. Matsuyama and V. Hwang. *SIGMA: A knowledge-based aerial image understanding system.* Plenum Press, 1990.

[46] H. Mayer. Automatic object extraction from aerial imagery — a survey focusing on buildings. *Computer Vision and Image Understanding*, 74(2):138–149, May 1999.

[47] J. David M. McKeown, J. Wilson A. Harvey, and J. McDermott. Rule-based interpretation of aerial imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(5):570–585, September 1985.

[48] M. Moya, M. Koch, and L. Hostetler. One-class classifier networks for target recognition applications. In *World Congress on Neural Networks*, pages 797–801, 1993.

[49] M. Nagao and T. Matsuyama. A structural analysis of complex aerial photographs. In *Advanced Applications in Pattern Recognition.* Plenum Publishing, 1980.

[50] S. Newsam and B. S. Manjunath. Normalized texture motifs and their application to statistical object modeling. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Workshop on Perceptual Organization in Computer Vision*, June 2004.

[51] B. Nicolin and R. Gabler. A knowledge-based system for the analysis of aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, 25(3):317–329, 1987.

[52] S. Noronha and R. Nevatia. Detection and modeling of buildings from multiple aerial images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5):501–518, May 2001.

[53] M. Pesaresi and J. A. Benediktsson. A new approach for the morphological segmentation of high-resolution satellite imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 39:309–320, February 2001.

[54] N. J. Pizzi, R. A. Vivanco, and R. L. Somorjai. Evident: a functional magnetic resonance image analysis system. *Artificial Intelligence in Medicine*, 21(1):263–269, January 2001.

[55] F. Quint. Recognition of structured objects in monocular aerial images using context. In *Mapping buildings, roads and other man-made structures from images*, pages 213–228. Birkhauser Verlag, Munchen, 1997.

[56] G. Rellier, X. Descombes, J. Zerubia, and F. Falzon. A gauss-markov model for hyperspectral texture analysis of urban areas. *Proceedings of the International Conference on Pattern Recogition*, 1:692–695, 2002.

[57] G. Ritter and M. T. Gallegos. Outliers in statistical pattern recognition and an application to automatic chromosome classification. *Pattern Recognition Letters*, 18(6):525–539, 1997.

[58] S. Roberts and L. Tarassenko. A probabilistic resource allocating network for novelty detection. *Neural Computation*, 6(2):270–284, 1994.

[59] M. Rochery, I. H. Jermyn, and J. Zerubia. Gap closure in (road) networks using higher-order active contours. In *Proceedings of IEEE International Conference on Image Processing*, Singapore, October 2004.

[60] C. Sanchez-Hernandez, D. S. Boyd, and G. M. Foody. One-class classification for mapping a specific land-cover class: SVDD classification of Fenland. *IEEE Transactions on Geoscience and Remote Sensing*, 45(4):1061–1073, April 2007.

[61] B. Scholkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. Williamson. Estimating the support of a high-dimensional distribution. Technical report, Microsoft Research, 1999.

[62] W. Sethares and T. Staley. Periodicity transforms. *IEEE Transactions on Signal Processing*, 47(11):2953–2964, 1999.

[63] A. Sharkey and N. Sharkey. How to improve the reliability of artificial neural networks. Technical report, Department of Computer Science, University of Sheffield, 1995.

[64] V. V. Starovoitov, S.-Y. Jeong, and R.-H. Park. Texture periodicity detection: features, properties, and comparisons. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(6):839–849, November 1998.

[65] D. Tax and R. Duin. Uniform object generation for optimizing one-class classifiers, 2002.

[66] D. M. J. Tax. *One-class classification*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2001.

[67] D.M.J Tax and R.P.W Duin. Data domain description using support vectors. In M. Verleysen, editor, *Proceedings of the European Symposium on Artificial Neural Networks 1999*, pages 251–256. D.Facto, Brussel, April 1999.

[68] C. Unsalan and K. L. Boyer. A theoretical and experimental investigation of graph theoretical measures for land development in satellite imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):575–589, April 2005.

[69] T. Wassenaar, J. Robbez-Masson, P. Andrieux, and F. Baret. Vineyard identification and description of spatial crop structure by per-field frequency analysis. *International Journal of Remote Sensing*, 23(17):3311–3325, September 2002.

[70] J. S. Weszka, C. R. Dyer, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(4):269–285, April 1976.

[71] J.S. Wezska, C. R. Dyer, and Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 6(4):269–285, 1976.

[72] G. G. Wilkinson. Results and implications of a study of fifteen years of satellite image classification experiments. *IEEE Transactions on Geoscience and Remote Sensing*, 43(3):433–440, March 2005.

[73] D. H. Wolpert. Stacked generalization. Technical Report LA-UR-90-3460, Sante Fe Institute, Los Alamos, NM, 1990.

[74] Yiming Yang, Tom Pierce, and Jaime Carbonell. A study on retrospective and on-line event detection. In *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 28–36, Melbourne, AU, 1998.

[75] Dit-Yan Yeung and Calvin Chow. Parzen-window network intrusion detectors. In *Proceedings of the Sixteenth International Conference on Pattern Recognition*, volume 4, pages 385–388, August 11–15, 2002.

[76] A. Ypma and R. Duin. Support objects for domain approximation. In *Proceedings of International Conference on Artificial Neural Networks ICANN'98*, Skovde,Sweden, September 1998.

[77] S. W. Zucker and D. Terzopoulos. Finding structure in co-occurrence matrices for texture analysis. *Computer Graphics and Image Processing*, 12:286–308, March 1980.