AUTOMATED DESCRIPTION OF 2-D BUILDING BOUNDARIES FROM A SINGLE COLOR AERIAL ORTHO-IMAGE

Ali Özgün Ok

Dept. of Geodetic and Geographic Information Technologies,
Middle East Technical University, Ankara, Turkey 06531 - oozgun@metu.edu.tr

Commission IV, WG IV/2

KEY WORDS: Color Aerial Imagery, Building Detection, Segmentation, Description, Photometric Quasi-invariants

ABSTRACT:

In this study, an automated methodology for the detection and description of 2-D building footprints from a single color aerial ortho-image is developed. The methodology is initialized with the mean-shift segmentation algorithm. Next, a vector-valued canny edge detection algorithm which relies on the photometric quasi-invariant gradients is utilized to detect edges from the color segmented image. Several morphological operations are applied to the edge image and two raster datasets are automatically generated from the morphologically reconstructed edge image, (i) the edge pixels that form closed-boundary shapes, and (ii) the edge pixels that do not form closed-boundary shapes. The first dataset, the edge pixels that form closed-boundary shapes, are vectorized using boundary tracing followed with the douglas-peucker simplification algorithm. On the other hand, a minimum bounding convex-hull algorithm followed with gradient vector flow (GVF) snake is used to generate polygons from the second dataset. Two vegetation indices are combined to mask out the polygons that belong to healthy vegetated areas. Finally, a detailed post-processing operation is utilized to purify the polygon boundaries obtained from the detection step. First, within the borders of the each detected polygon, a K-means unsupervised classification is performed. For each polygon, the pixels that belong to the largest class are selected and utilized as seed points in a color region growing process along with several morphological operations. Next, a distance transform (DT) snake and a final simplification process is performed to locate the purified boundary descriptions of the buildings. Among the available 251 apartment buildings in the site, 97.2% of them were detected. Furthermore, among the detected 244 apartments, 212 of them were almost completely detected. The algorithm provided 62.6% accuracy for the buildings that are in a dense environment, 691 out of 1104 neighboring house settlements were detected. The pixel based accuracy analyses revealed that only %9.24 of the total number of pixels were wrongly stated as buildings (incorrect building pixels percentage) and the correct non-building pixels percentage was computed to be 98.48%.

1. INTRODUCTION

Until now, a large number of studies have been performed in the context of automatic building extraction from aerial images. A wide variety of approaches have been developed and integrated into several systems; however, in the meanwhile although the algorithms and the extraction strategies may differ from each other, the accuracy and the performance of those systems are far from reaching the standards. The previous work in the context of aerial imagery until the mid-1999 was reported in an exceptional review performed by Mayer (1999). Later, an extended version of Mayer's review (until the late-2003) bounded in his format was conducted by Ünsalan and Boyer (2005). The knowledge-based image analysis for object extraction and the different aspects of knowledge that can be used for building extraction were reviewed by Baltsavias (2004). The trends followed within the state of art of building extraction can be found in elsewhere (Grün et. al., 1995; Grün et. al., 1998; Baltsavias et. al., 2001).

In this study, an automated methodology for the detection and description of 2-D building boundaries from a single color aerial ortho-image is developed. During this study, the color information available in the aerial ortho-image is fully utilized during the different steps of the processing including the segmentation, edge detection, post-processing etc. Thus, the accuracy and the performance of the detection and description of the 2-D building boundaries are exploited.

2. METHODOLOGY

The main steps followed in the proposed methodology are given in Figure 1. The methodology is initialized with the mean-shift segmentation algorithm. Next, a vector-valued canny edge detection algorithm which relies on the photometric quasi-invariant gradients is utilized to detect edges from the color segmented image. Several morphological operations are applied to the edge image and two raster datasets are automatically generated from the morphologically reconstructed edge image, (i) the edge pixels that form closed-boundary shapes, and (ii) the edge pixels that do not form closed-boundary shapes. The first dataset, the edge pixels that form closed-boundary shapes, are vectorized using boundary tracing followed with the douglas-peucker simplification algorithm. On the other hand, a minimum bounding convex-hull algorithm followed with gradient vector flow (GVF) snake is used to generate polygons from the second dataset. The polygon results of both datasets are joined together in a unification step. Based on the fact that the ortho-images used do not have a near-infrared band, the verification of the generated polygons with respect to the healthy vegetation cover, two vegetation indices, namely the Ratio Index and the Visible Atmospherically Resistant Index are combined to mask out the polygons that only belong to vegetated areas. Finally, a detailed post-processing operation that includes the following steps is utilized to purify the polygon boundaries obtained from the detection step. First, within the borders of the each detected polygon, a K-means unsupervised classification is performed. In each polygon,

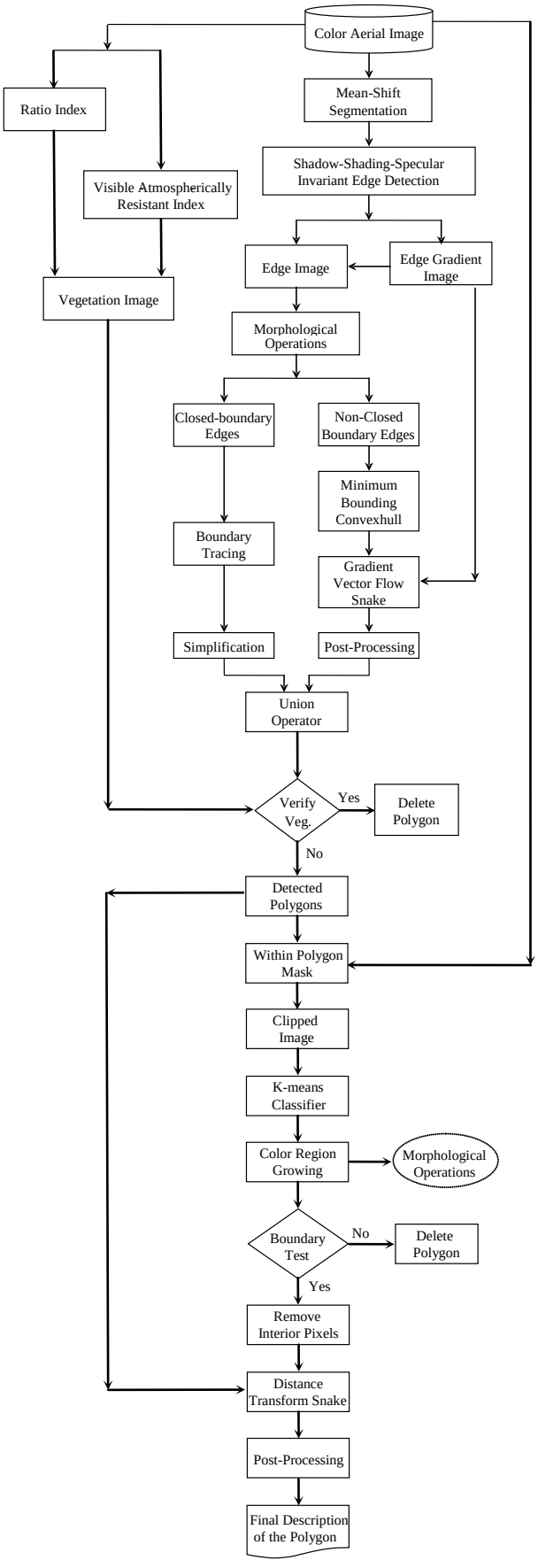


Figure 1. The proposed 2-D building detection methodology

the pixels that belong to the largest class are selected and utilized as seed points in a color region growing process along with several morphological operations. Next, a distance transform (DT) snake and a final simplification process is performed to finalize the boundary descriptions of the buildings.

2.1 Mean-Shift Segmentation

In general, the mean-shift based image segmentation is composed of two successive steps that are namely (i) the discontinuity preserving smoothing of the original image, and (ii) performing the segmentation based on the smoothed image (Comaniciu and Meer, 2002). Discontinuity preserving smoothing technique adjusts the degree of smoothing; i.e., the amount of smoothing significantly reduced in close proximity of the edges. On the other hand, the segmentation is an extension of the discontinuity preserving smoothing algorithm; each pixel is associated with a significant mode of the joint domain density located in its neighborhood, after nearby modes were pruned as in the generic feature space analysis technique. Since an image is composed of pixels with proper gray level, color, or spectral information the incorporation of the spatial coordinates of a pixel into its feature space representation is required. This is accomplished by a joint domain representation which takes into account both the spatial and spectral information of an image. Thus, the multivariate kernel is defined as the product of two radially symmetric kernels and a single bandwidth parameter is allowed for each domain

$$K_{h_s,h_r}(x) = \frac{C}{h_s^2 h_r^p} k\left(\left\|\frac{x^s}{h_s}\right\|^2\right) k\left(\left\|\frac{x^r}{h_r}\right\|^2\right) \quad (1)$$

In equation (1) x_s is the spatial part, x_r is the range part of a feature vector, $k(x)$ the common profile used in both two domains, h_s and h_r the employed kernel bandwidths, and C the corresponding normalization constant. In practice, an Epanechnikov or a normal kernel types always provides satisfactory performance, so the user only has to set the bandwidth parameter $h = (h_s, h_r)$ in equation (1), and the minimum number of pixels (area) that allowed within a single region M.

2.2 Color Edge Detection

The low level object extraction performances of the detectors used so far are always limited and mostly defective since they work on a single band or component. To correct this shortcoming, a vector-valued technique which relies on the photometric quasi-invariants is utilized to detect edges from color aerial images. The method was developed by Gevers and Smeulders (1999), and relies on the classification of edges into meaningful classes by means of the derivatives of different color spaces. The classification used is based on the generation of different color space derivatives which are quasi-invariant to specific type of edges, such as shadow-shading, specular, or both shadow-shading and specular. Once these derivatives are accurately found, they are combined using a structure tensor in a color version of the Canny algorithm (Weijer et. al., 2006). In this study, two minor improvements are made to their algorithm, (i) the output of the final gradient map is scaled between zero-and-one before further processing which significantly reduced the remaining noise edges and (ii) a two level hysteresis thresholding is designed to have a better control

on the final edge contours. After performing the color Canny algorithm with shadow-shading-specular quasi-invariance, a single gradient image is obtained. The hysteresis thresholding is used to convert the gradient image to a binary edge image which consists of white pixels forming the edges and the black pixels representing the background.

2.3 Morphological Operations

The first step of this part is to bridge the unconnected edge pixels after the color edge detection. This is performed using morphological operations in which each edge pixel in the output image is generated based on a comparison of the corresponding edge pixel in the input image with its very close neighbors. Three morphological operations are successively applied to the original edge image. First, morphological closing (dilation followed with erosion) is applied using a square structuring element of a 3-by-3 matrix. Next, morphological bridging operator is utilized. This operator sets 0-valued pixels to 1 if they have two non-zero neighbors that are not connected. Finally, a thinning operator is used to generate one-pixel wide edges in the morphologically reconstructed edge image.

Although the morphological operations linked the broken edges that are in a very close neighbor, many edges still remain unconnected (non-closed boundary shapes). However, some of the edges in the morphologically reconstructed image inherently resulted in closed-boundary shapes which do not require a particular processing as compared to their non-closed counterparts. To identify and separate these two shape formations, a morphological shrinking operator followed with a cleaning operator is used. The shrinking operator removes pixels so that objects that do not form closed-boundaries shrink to an isolated edge point, and the cleaning operator removes those isolated points from the edge image. Thus, the cleaned edge image is only composed of the edge pixels that are closed-boundary shapes. Finally, the cleaned edge image is subtracted from the previously reconstructed edge image to find the edge pixels that do not form closed-boundary shapes.

2.4 The Construction of the Vector Polygons

The aim is to generate accurate vector polygons from the generated two binary raster datasets, (i) the edge pixels that form closed-boundary shapes, and (ii) the edge pixels that do not form closed-boundary shapes. First, the processing of the closed-boundary shapes is carried out. As previously stated, closed-boundary shapes do not require a particular processing and inherently ready to be an input to a well-known vectorization algorithm, boundary tracing. During the boundary tracing processing, the exterior boundary of each closed-boundary shape is traced and converted to vector polygons. A douglas-peucker simplification algorithm with a threshold ε is used to reduce the number of vertices generated from boundary tracing.

A more complicated processing is involved during the generation of vector polygons from the edge pixels that do not form closed-boundary shapes. Each non-closed-boundary shape in the binary edge image is converted to objects using a method called connected-component labeling. This method assigns a unique integer value to each object based on a selected connectivity measure. Next, for each object, a Convex-Hull algorithm is applied. This algorithm computes the smallest convex polygon that contains the minimum bounding region defined by each object. The convex polygons generated using the convex-hull algorithm for each object is used to initialize the Gradient Vector Flow (GVF) snake. GVF is a dense vector field derived from images by minimizing certain energy functions in a variational framework and a GVF snake is an active contour that uses the GVF field as its external force (Xu and Prince, 1998). The initialization of the GVF snake is performed using the convex polygons; however, the initialization may not be always accurate enough to recover the correct polygons from the binary edge image. Moreover, the hysteresis thresholding applied to the gradient image lost some of the edges available in the gradient image. Although very-close-neighbor information is reconstructed using morphological processing, there are still many gaps and missing parts in the processed edge image. However, these potential problems are minimized by the nature of the GVF snake itself. First, the GVF snake is insensitive (or less sensitive) to the initialization and has a large capture range. Therefore, the polygons computed with the convex-hull algorithm are mostly sufficient to initialize the GVF snake. Moreover, the GVF field of the snake is computed by using the original gradient image to enhance the progression of the GVF snake. As a result, all the information available in the original gradient image influenced the final configurations of the polygons for each object, and the ability to move the objects' boundary concavities of the GVF snake has resulted in better final configurations. Once the final configurations of all object polygons are computed, a post-processing operation is performed to remove the incorrectly configured parts. This operation mainly includes two criteria; (i) multiple-buffering (margins at a pixel distance) created around each polygon object and the size of the (ii) minimum and (iii) maximum area allowed by a single polygon object. A douglas-peucker simplification algorithm with the same threshold (ε) is used to reduce the number of vertices of the post-processed polygons.

In the last step, the generated vector polygons from the two binary raster datasets are combined to produce a final polygon dataset. This is performed using a simple union operator.

2.5 Removal of Vegetation Polygons

In this part, the generated vector polygons that belong to vegetated areas in the color (RGB) aerial image are masked out. It is a well-known fact that a common way to extract vegetated areas is to use various indices which are principally dimensionless radiometric measures that point towards the vegetation. To eliminate healthy green vegetation, many of the indices utilize the apparent inverse relationship between the red and near-infrared bands (Jensen, 2005). However, the aerial image used is only composed of visible bands in which many available indices become useless. Therefore, two different indices each of which utilize only the information obtained from the visible bands are used. The first index used is computed as the ratio between the green reflected radiant flux (ρ_{green}) and the blue radiant flux (ρ_{blue}):

$$RI = \frac{\rho_{green}}{\rho_{blue}} \tag{2}$$

The second index used is proposed by Gitelson et. al. (2002) and called *Visible Atmospherically Resistant Index* (VARI):

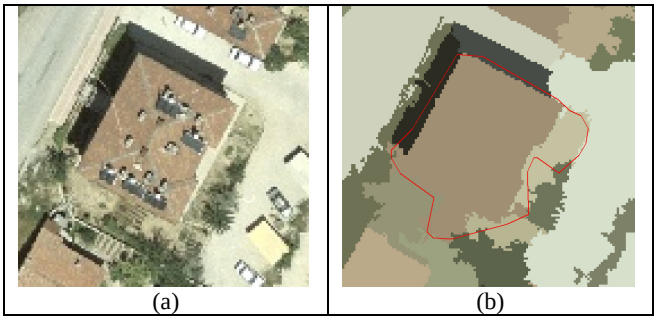
$$VARI = \frac{\rho_{green} - \rho_{red}}{\rho_{green} + \rho_{red} - \rho_{blue}} \tag{3}$$

The first index expresses the relationship between the green and the blue bands; however, the second index is mostly devoted to the green and red band interaction. The results of both indices are scale to 0-and-255 and converted to binary vegetation images (0-negative evidence and 1-positive evidence) using a certain threshold. Next, two vegetation images a combined using a logical array operator *AND* (&) in a final vegetation image. The generated polygon dataset is overlaid to the vegetation image and the number of positive evidences under each polygon is counted and divided by the total number of pixels (ratio) under that polygon. During the verification, if the ratio is found to be higher than a certain value for a specific polygon, it is automatically labeled as a vegetation object and deleted from the dataset.

2.6 Post-Processing of the Detected Polygons

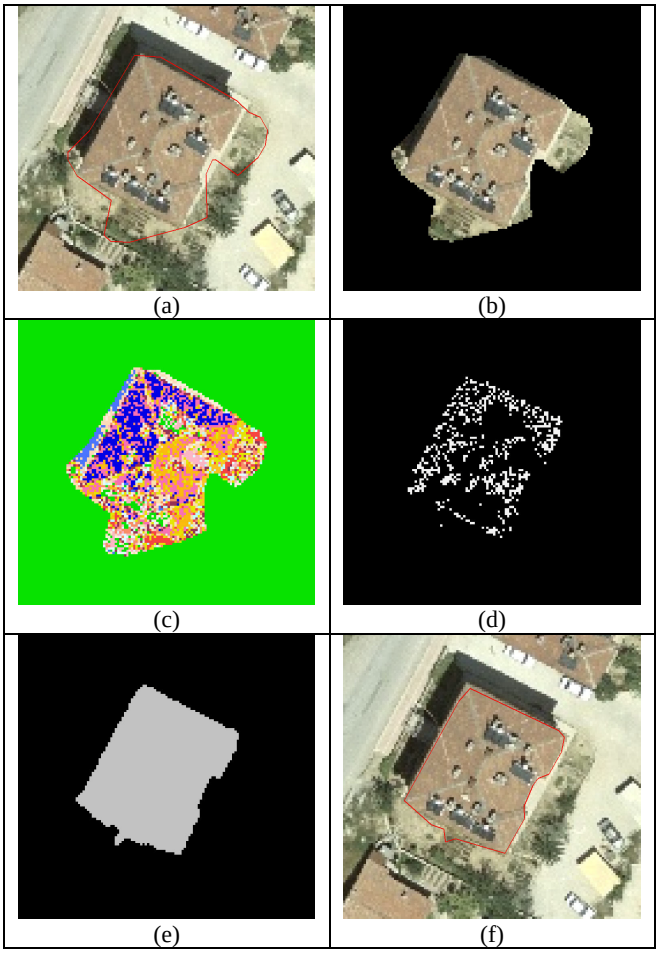
Generally, the shapes of the detected polygons are poorly affected from the segmentation step. After the segmentation, most of the observed details on a building roof (chimneys, ventilators, or solar energy panels, etc.) are removed. However, the segmentation also disturbs the original shape of the building, especially if the buildings have similar surrounding objects. An example of this problem is shown in Figure 2. It is clear that, although the detected polygon successfully covers the building, it also encloses some parts that do not belong the building. As a result, the number of pixels incorrectly labeled as building increases dramatically. In a previous study performed by Ok (2008), the computed percentage for the incorrect building pixels is computed as 24.75% and one of the reasons for this relatively high number was due to this problem.

To overcome this problem, a post-processing step to the detected polygons is applied. First, for each polygon, a minimum bounding rectangle is computed. By using a certain value, the bounding rectangle is enlarged and the original image is clipped (Figure 3.a). Since the only interested is the inside of the detected polygon, a within polygon mask is generated (Figure 3.b), and a well-known k-means clustering algorithm is performed. In order to capture every detail (object) inside the polygon, the maximum number of clusters is defined as 16. In order to prevent the algorithm stuck on to local minimums, the algorithm has been iterated n times with randomly selected initial centroids. The k-means solution that has the lowest value for the total sum of distances to the centroids is selected as the final clustering result. The final k-means image is shown in (Figure 3.c), and note that the large green class that encloses the polygon boundary belongs to the null class. In the next step, the pixels that belong to the largest class are selected (Figure 3.d). The pixels selected are assumed to be distributed across the parts of the roofs within the polygon. Actually, this assumption



(a) (b)

Figure 2. (a) Original image, and (b) the detection result overlaid to the segmented image



(a) (b) (c) (d) (e) (f)

Figure 3. (a) original image and the detected polygon, (b) masked image, (c) k-means classification result, (d) the pixels that belong to the largest class, (e) color region growing result, and (f) post-processed final polygon.

holds for the cases in which the buildings are mostly have non-steep roof surfaces and the sun elevation during the image acquisition is sufficiently high. Next, the selected pixels are utilized as seed points for a color region growing process. The building region is iteratively segmented by comparing all unallocated pixels inside the polygon. For the measure of similarity, a single threshold value is selected; however, the threshold constraint is also applied to all available bands (RGB). Several morphological operations are also applied to improve the boundaries of the region. The result of the color region growing process is shown in (Figure 3.e). In some cases, especially when the detected polygon does not belong to any building object, the growth region may grow unexpectedly within the clipped region. In this study, a simple boundary test is applied to locate these regions. If the final growth region has reached at least two sides of the clipped region, the polygon is marked as non-building and deleted. For sure, the success of this test is simply related with the size of the clipping region and the value of the region growing threshold. Both parameters must be well tuned in order to prevent wrong polygon eliminations. After that, the interior pixels of the region are removed and the edge pixels are utilized to generate the distance potential forces of the DT snake. There are several reasons for choosing the DT snake. The DT snake has a large capture range similar to the GVF snake. However, in contrast to the GVF snake, it does not have ability to move into boundary concavities (Xu and Prince, 1998). By using the DT snake, this

shortcoming turns into an advantage, since the region growing results are not accurate most of the time. There may be small gaps within the growth region due to several reasons, for example very dense small objects on a part of the roof may break down the growing. In that kind of situation, the aim should not be to move into these small concavities of the region, but only trace the outer boundary. Thus, the DT snake produces useful and accurate results in these cases. For the initial position of the snake, the previously detected polygon is given and once the final configuration of the snake is found, two post-processing operations that are previously explained (multiple buffering and douglas-peucker simplification) are applied to obtain the final configuration of the polygon (Figure 3.f).

The proposed methodology was implemented in Matlab ® v. 7.5 environment. Only the mean-shift segmentation is performed using a stand-alone system (EDISON) developed by Comaniciu and Meer (2002).

3. STUDY AREA AND DATA SETS

The study area selected is over a residential part of Konya, Turkey. The set of raw color (RGB) images was acquired with Zeiss RMK TOP 30 camera with typical photogrammetric overlaps (60% end - 20% side) for mapping. The calibrated focal length of the camera was 305.536 mm and the flying height was approximately 1200 m above the ground. The images were scanned at 14-μm resolution with a photogrammetric film scanner and it corresponds to a final ground sampling distance of approximately 6-cm. The camera was accurately pre-calibrated and passed through a rigorous simultaneous bundle block adjustment procedure. Around 350 3-D vector layers were generated from the available stereo-pairs in a digital photogrammetric workstation environment by qualified human operators. The ortho-image used in this study was generated using a digital terrain model (DTM) which was produced from the specific layers (contours, roads etc.) of the 3-D vector dataset. The area covered by the ortho-image was around 1.5 km^2, and the spatial resolution was determined to be 30 cm during the orthorectification process.

The ortho-image includes buildings with different shapes, sizes, and orientation. In the area, the total number of buildings is 1403. 96.6% of those buildings are residential and the remaining is a mixture of under-construction, government, educational, and religious buildings. Among the available 1403 buildings, 251 of them are residential apartment buildings and mostly built in a regular pattern. Other residential buildings are mostly have heights less than 6 m and are densely neighboring house settlements (detached, semi-detached or terraced buildings).

4. RESULTS AND DISCUSSION

The following accuracy measures proposed in (Lin and Nevatia, 1998) are used to test the quality of the results of the proposed methodology. Five different measures are computed:

- Detection Percentage = 100 x TP / (TP + TN)
- Branch Factor = 100 x FP / (TP + FP)
- Correct Building Pixels Percentage
- Incorrect Building Pixels Percentage
- Correct Non-building Pixels Percentage

In the first two measures, TP (True Positive) denotes the number of buildings that exist in the reference vector dataset and detected by the method, TN (True Negative) denotes the number of buildings that only exist in the reference vector dataset but not detected by the method, and FP (False Positive) denotes the number of buildings detected by the method but not exist in the reference vector dataset. The last three measures are determined by counting the correct building and non-building pixels (Lin and Nevatia, 1998).

In this study, the percentages of the detection are also computed based on four different building types, (i) apartment, (ii) detached or terraced, (iii) under-construction, and (iv) other. It is important to emphasize that a building is considered to be detected if any part of the building is detected; however, for each building, the detected portion is calculated and classified into two detection categories, (i) complete or (ii) partial detection. A building is classified as complete detection if the detected region pixels cover more than 75% of its area (Fradkin et. al., 2001); otherwise it is classified as partial detection. In order to give an idea about how the buildings are partially detected, they are further classified into three sub-classes (see Table 1 for the portions of the sub-classes).

Table 1 summarizes the results of the proposed methodology based on four different types of buildings. The detection rates among the different types of buildings ranged between 11% and 97%. The best detection results (97%) are achieved for the apartment buildings. 212 out of 251 apartment buildings are completely detected and the remaining 32 are detected partially. Figure 4 demonstrates a small part of the study area used. It is very important to state that the locations of the apartment buildings are successfully found independent from their size, shape and orientation. In addition, note that the descriptions of the apartment buildings are very close to the reference data. Most of the buildings in the study area (1104 out of 1403) are densely neighboring house settlements (detached, semi-detached or terraced buildings). For those buildings, the algorithm provided 63% detection rate. Among the detected dense buildings, approximately 69% of them are completely detected. The worst results are achieved for the buildings that are under-construction at the time of image acquisition. For those buildings, the detection rate was computed to be 11%. In fact, this is an *expected* result due to two explicit reasons. First, some of those buildings are in their very early construction stage. Therefore, there is not enough color difference between the building object and its background. Thus, an over-segmentation was observed for those buildings. The second reason is due to the color Canny edge detection used. Most of the under-construction buildings are in a condition of very high reflectance. Since the color edge detection used suppresses the specular edges, the bright nature of those buildings is treated as specular edges and inherently suppressed by the edge detector.

Table 2 shows the overall results computed for the whole ortho-image. The overall detection percentage (66.7%) is slightly higher than the detection percentage computed for the densely neighboring house settlements (detached, semi-detached or terraced buildings). This is not surprising because the buildings in the study area are mostly dominated by the detached, semi-detached or terraced buildings. It is also evident that the computed percentage for the incorrect building pixels is 9.24%. If this result is compared with the previous study (Ok, 2008), it is clear that there is a substantial improvement for the computed percentage of the incorrect building pixels (24.75%). Actually, this result is mostly related with the post-processing stage of the

detected polygons. Finally, as can be expected to be high as most pixels in an image are non-building pixels, the percentage for the correct non-building pixels are computed to be 98.48%.

5. CONCLUSIONS AND FUTURE WORK

In this study, an automated methodology for detecting buildings from a single color aerial image is presented. The methodology is tested for a large number of buildings with different shapes, sizes, and types in a complex environment. The algorithm provides high detection rates (97%) for the apartment building. The performance of the algorithm decreases to 63% in a dense environment (detached, semi-detached or terraced buildings). The modification of the methodology to handle stereo and/or multiple color aerial images is currently under development and will further improve the detection and description rates.

6. REFERENCES

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

Baltsavias, E. P., 2004, Object Extraction and Revision by Image Analysis Using Existing Geodata and Knowledge: Current Status and Steps Towards Operational Systems, *ISPRS Journal of Photogrammetry & Remote Sensing*, 58, pp. 129– 151

Comaniciu, D., and Meer, P., 2002, Mean Shift: A Robust Approach toward Feature Space Analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), pp. 603-619.

Gevers, T., and Smeulders A. W. M., 1999, Color-based Object Recognition, *Pattern Recognition*, 32, pp. 453-464.

Gitelson, A. A., Kaufman, Y. J., Stark, R., and Rundquist, D., 2002, Novel Algorithms for Remote Estimation of Vegetation Fraction, *Remote Sensing of Environment*, 80, pp.76-87.

Grün, A., Kübler, O., and Agouris, P., 1995, Automatic Extraction of Man-Made Objects from Aerial and Space Images, Birkhauser, Basel, Switzerland.

Grün, A., Baltsavias, E. P., and Henricsson, O., 1998, Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Birkhauser, Basel, Switzerland.

Jensen, J. R., 2005, Introductory Digital Image Processing: A Remote Sensing Perspective - 3rd ed., Prentice Hall, pp. 310-323.

Mayer, H., 1999, Automatic Object Extraction from Aerial Imagery - A Survey Focusing on Buildings, *Computer Vision and Image Understanding*, 74(2), pp. 138-149.

Ok, Ali Özgün, 2008, Robust Detection of Buildings from a Single Color Aerial Image, Proceedings of GEOBIA 2008 - Pixels, Objects, and Intelligence, Calgary, Alberta, Canada.

Ünsalan, C., and Boyer, K. L., 2005, A System to Detect Houses and Residential Street Networks in Multispectral Satellite Images, *Computer Vision and Image Understanding*, 98, pp. 423-461.

Weijer, J., Gevers, T., and Smeulders, A.W.M., 2006, Robust Photometric Invariant Features from the Color Tensor, *IEEE Transactions on Image Processing*, 15(1) pp. 118-127.

Xu, C., and Prince, J. L., 1998, Snakes, Shapes, and Gradient Vector Flow, *IEEE Transactions on Image Processing*, 7(3), pp. 359-369.

7. ACKNOWLEDGEMENTS

The author would like to thank MNG-Bilgisayar Company (Ankara, Turkey) for providing both the aerial ortho-image and the reference vector dataset.

Table 2. Overall results of the proposed methodology

	Detection Percentage tp / (tp+tn)	Branch Factor fp / (tp+fp)	Correct building pixels	Incorrect building pixels	Correct non-building pixels
Ortho-Image	66.5%	8.76%	62.54%	9.24%	98.48%

Figure 4. A small part of the ortho-image. The red polygons show the available reference data, the yellow polygons shows the results of the proposed methodology.

Table 1. The detection results of the proposed methodology based on the types of buildings

Detection	Detected Portion of Buildings	Building Type			
		Apartment	Detached or Terraced	Under-Construction	Other
Complete	(≥ 75%)	212	476	3	7
Partial	(50% ≤ x < 75%)	23	103	-	2
	(25% ≤ x < 50%)	7	44	-	3
	(< 25%)	2	68	-	-
Non-Detected		-	413	25	8
Detected Total		244	691	3	12
Total Number of Buildings		251	1104	28	20
Detection Percentage (%)		***97%***	***63%***	***11%***	***60%***