

# 3D Polygonal Building Model Estimation from Single Satellite Images

Mohammad Izadi, *Student Member*, and Parvaneh Saeedi, *Member, IEEE*

**Abstract**—This paper introduces a novel system for automatic detection and height estimation of buildings with polygonal shape roofs in singular satellite images. The system is capable of detecting multiple flat polygonal buildings with no angular constraints or shape priors. The proposed approach employs image primitives such as lines, and line intersections, and examines their relationships with each other using a graph-based search to establish a set of rooftop hypotheses. The height (mean height from rooftop edges to the ground) of each rooftop hypothesis is estimated using shadows and acquisition geometry. The potential ambiguities in identification of shadows in an image and the uncertainty in identifying true shadows of a building have motivated for a fuzzy logic-based approach that estimates buildings heights according to the strength of shadows and the overlap between identified shadows in the image and expected shadows according to the building profile. In order to reduce the time complexity of the implemented system, a maximum number of 8 sides for polygonal rooftops is assumed. Promising experimental results verify the effectiveness of the presented system with overall mean shape accuracy of 94% and mean height error of 0.53 meter on QuickBird satellite (0.6 meter/pixel) imageries.

**Index Terms**—Building detection, 3D building reconstruction, height estimation, satellite image processing.

## I. INTRODUCTION

**A**UTOMATIC 3D map reconstruction from optical images has been an active research subject with a wide range of applications such as urban environmental planning, military assessment simulations, resource management and control for disaster preparedness. 3D building reconstruction perhaps is the most prominent component of this research subject that includes two main tasks: building roof detection and height estimation. For many years 3D building reconstruction is performed through semi-automatic approaches where an operator identifies building boundaries in a set of stereo aerial images. Using acquisition geometry, image displacement, and perspective projection, the height of buildings are determined. This process is time consuming and tiresome with low update rate and high cost. The abundance of inexpensive frequently updated satellite imageries has initiated much work toward fully automated systems for 3D building reconstruction. While numerous semi-automatic systems have been developed, only a limited number of automated systems are reported in the literature. Some of these works present instances of limited good results and are especially developed for simple buildings in higher resolution imageries. They however are still far from being capable of coping with existing complexities of urban structures.

M. Izadi (mia4@sfu.ca) and P. Saeedi (psaeedi@sfu.ca) are with the Laboratory for Robotic Vision at the School of Engineering Science at Simon Fraser University, Burnaby, BC, Canada.

The work in this paper aims for automatic detection and mean height estimation of buildings with flat or flat looking polygonal roofs in panchromatic satellite images.

### A. Related Work

Proposed approaches for solving the problem of 3D building reconstruction vary based on sensor modalities, complexity of buildings, the level of human supervision/interaction, and additional input resources such as digital elevation models (DEM) or digital surface models (DSM).

Some of these methods address the problem of 2D rooftop detection only. In this group, curve evolution-based methods including deformable boundaries, active contour models, and segmentations, have been very popular [1]. Peng *et al.* [2] proposed an improved snake model based on radiometric and geometric behaviors of buildings. Mayunga *et al.* [3] proposed a semi-automatic approach using radial casting algorithm to initialize snake-based contours. Ruther *et al.* [4] reported a semi-automatic approach using DSM to generate initial raised structure hypotheses that were later refined via active contour method. One of the attractive characteristic of these methods is their adaptivity to topological variations that obviously exist in rooftop shapes. They also incorporate local features such as surface and boundary information that inherently tends toward good localization. These methods however are subject to inaccuracies caused by occlusion and illumination variation. Moreover most of these approaches are semi-automatic.

Model-based approaches also have been reported for the rooftop detection problem. Liu *et al.* [5] developed a semi-automatic rooftop detection algorithm for rectilinear models using Hough transform. Bailloeuil *et al.* [6] introduced a system for building recognition using specific geometrical information derived from prior knowledge of building models combined with active contours models. Karantzas and Paragios [7] proposed a recognition-driven building detection method using templates combined with energy minimization approach. Model-based methods seem to be capable of dealing with partial occlusion. However, since they are model dependent, they either assume simple model profiles, or require a large number of prior models. Each one of the above two conditions can impact the quality and efficiency of the recognition.

Other works present complete solutions for the 3D reconstruction problem. In this type of work, the rooftops are identified first and their heights are estimated next. The general solution for detecting rooftops could fall into the above mentioned categories. Proposed solutions for estimating building height and the 3D reconstruction vary according to the number of scene images (single or multiple) and additional supplementary data resources.

Systems with single-view input images generally use casting shadows to measure the height of the buildings. Irvin and McKeown [8] utilized relationships between shadows and man-made structures from aerial images to firstly predict simple structures shapes, to secondly group related structures, to thirdly verify individual structures and to finally estimate the structure height. Lin *et al.* [9], [10] employed shadows to verify building hypotheses and estimate their heights. These methods have the advantage of estimating height using single images. The accuracy of these methods however could be affected by the quality of detected shadows and the interference by nearby buildings or objects. Moreover, all previous works in this group assume simple building models such as square and rectilinear.

Some approaches used multiple views and stereo images to reconstruct 3D model of buildings. Noronha and Nevatia [11] proposed a method that reconstructed 3D models of rectilinear buildings or compositions thereof (L, T, and I shapes) from multiple view (non-stereo) aerial images. They utilized local features to verify rectilinearly constrained rooftop hypotheses. They estimated the height using the overlap between walls, shadow boundaries and corner supporting lines with image edges. They also used visible shadow junctions on the shadow boundaries and the intensity information inside the shadow boundary. Cord and Declercq [12] presented a method for high-resolution monochromatic aerial image pairs for creating DEM and modeling multi-slope rooftops. Kim and Nevatia [13] utilized multiple overlapping images of a scene to describe complex buildings. Baillard *et al.* [14], [15] proposed methods based on 3D lines and planes in multiple view images. Stereo-based methods could be robust approaches for estimating 3D height values in high resolution imageries; however, several issues makes them disadvantageous: for instance, required calibration with a rigid stereo platform, potential ambiguity in matching uniform areas, extra computation per frame, and proportionality of the height estimation error with the square of the depth. Moreover, stereo based methods require at least two exploitable optical images (for instance, without cloud cover), which is not necessarily the case in operational conditions (sometimes, only one of the available data can be fully exploited). 3D reconstruction using multiple views are technically more challenging due to issues such as match correspondence establishment under different views, sensitivity to extraction of features, and association of building components.

The use of additional data resources such as DEM, DSM, rangefinders (laser-based distance estimators) and LiDAR sensors have been reported for the problems of building detection and height estimation. Fujii and Arikawa [16] proposed a method that utilized airborne laser elevation maps with aerial images for the 3D building reconstruction. Jaynes *et al.* [17] used a DEM, registered to a corresponding optical view of an urban site, to reconstruct variety of building types. Xie *et al.* [18] proposed utilizing electro-optic and range images to reconstruct buildings with flat rooftops with various shapes. Sohn and Dowman [19] employed IKONOS images with LiDAR data for 3D reconstruction of the buildings with convex polygonal rooftops. Lafarge *et al.* [20] introduced an automatic

system using DEMs to model polygonal flat and symmetrical two-plane gabled rooftops (in the absence of DEM discontinuities) at a high computational cost. Incorporating DEMs with resolutions often within several meters could assist only for 3D modeling of large scale buildings. Therefore most LiDAR/DEM/DSM-based methods include optical images to increase the accuracy and handle potential discontinuities which increases the computational cost.

In this paper we present a set of methodologies for automatic extraction and 3D modeling of buildings with polygonal flat or flat looking rooftops in monocular satellite images. The suggested approach is based on existing and potential line intersections in the image. These line intersections act as potential vertices of rooftop candidate hypotheses. With each intersection point, two directions defined by the intersecting lines, are associated. Using a graph representation, the relationships between intersection points and lines are examined. Finding a polygonal shape in the image corresponds to finding a closed loop in the above graph. Local image properties are used to assess the quality of the found shapes as rooftop hypotheses. Heights (mean height from rooftop edges to the ground) of buildings are estimated using related shadow evidences in the image. The relevance of shadows is determined using geometrical shape constraints and the similarity of the existing and expected shadows (for a range of height values). To predict expected shadows, the wall and shadow vectors are computed by projecting the rooftop vertices on the ground. At each candidate height, a fitness score is computed using a set of fuzzy rules that represents the matching quality between existing and expected shadows at that height. The performance of the proposed system is assessed using 20 QuickBird images.

The proposed method in this work has some similar aspects to the work presented by Nevatia in [11]. Both methods are proposed for detecting rooftops and estimating their heights using shadows. There are several differences between the two methods: First, Nevatia's rooftop hypotheses are limited to simple shapes that are rectangular or compositions thereof. Our system can detect all buildings with polygonal shapes with maximum 8 sides (the method is general and could be extended into  $N$ -sided polygons). Second, Nevatia's system uses parallel line segments to generate hypotheses and therefore creates a large number of hypotheses that must be refined using local and global characteristics. We are employing corners with directions of the lines that created such corners and evidence of line segment between the corners to generate rooftop hypotheses using an efficient graph-based approach. Third, for the height estimation, Nevatia matches edges and corners on the walls and shadow boundaries. Relying on edges for such purpose is tricky as generally nearby edges from roads, trees and adjacent buildings could be wrongly interpreted as the true edges corresponding to the shadow or walls. Moreover, if a shadow region includes bright objects with distinctive edges, the estimated height might be completely wrong. Also, relying on wall or shadow lines potentially makes the system less inaccurate under occlusions, as occlusion will also results in edges that lead to wrong estimation. The proposed method in this paper is a region based method that utilizes rooftop

shape as the matching constraint. It copes with occlusion more gracefully, since it penalizes for the occluded section while counting in both the non-occluded section and the remaining part of the occluded section. Moreover it can estimate accurate height values when the shadow region includes edges within, or when the shadow or wall boundaries are not strong.

### B. Contribution

The main contributions of this paper are as followings:

1. We propose a simple yet intuitive method for automatically detecting polygonal shape rooftops in panchromatic gray scale satellite images. The proposed method relies on lines and their intersections to explore those potential relationships that form rooftop hypotheses. We make no assumption about the rooftop color or shape except for the local texture smoothness. The system in general is capable of finding rooftop hypotheses with any number of sides. To reduce the time complexity, we have limited the number of polygonal sides to a maximum of 8.
2. We introduce a reliable approach for estimating accurate heights of complicated rooftops using single gray scale satellite images. This method includes a fuzzy logic-based approach that accounts for uncertainties associated with the shadow identification and complex shape matching processes. The proposed method accounts for inter-relationships between various parts of the rooftops and it is capable of accurate estimation under partial occlusion of the shadows.

## II. METHODOLOGY

The proposed system includes two main parts: *2D Rooftop Detection* and *3D Building Estimation*. In the first part, rooftops are detected. The output of this part is rooftop definitions in image coordinate system that includes an array of consecutive vertices. The second part uses the rooftop definitions and the acquisition geometry to estimate building heights. Figure 1 shows the flowchart of the proposed system. Section II-A represents details of the 2D rooftop detections. Section II-B describes the height estimation process. Experimental results corresponding to each section are described in Sections III.A and III.B.

### A. 2D Rooftop Detection

1) *Line intersection detection*: The most distinctive lines and their potential intersections as corner features in the image are found first. This process includes following sub-processes:

- Pre-processing: The input image is low-pass filtered to smooth out the noise. Here a Gaussian smoothing with a standard deviation of  $\sigma = 0.8$  and a kernel size of  $7 \times 7$  pixels is used.
- Straight line extraction: The objective of this step is to extract a set of straight-line segments from the image. The algorithm implemented to achieve this goal is the Burns line detector [21], which utilizes both the gradient

magnitude and gradient orientation to form line support regions and eventually straight line segments. The following steps describe this procedure.

1. Partition the pixels into bins based on the gradient orientation values. A bin size of 45 degrees was selected. This results in eight bins being used, and pixels are assigned to bins according to the rules set out in Table I.

TABLE I: Partitioning Pixels into Gradient Orientation Bins.

Bin Number	Gradient Orientation (GO)
1	$0^\circ \leq GO < 45^\circ$
2	$0^\circ \leq GO < 90^\circ$
3	$90^\circ \leq GO < 135^\circ$
4	$135^\circ \leq GO < 180^\circ$
5	$180^\circ \leq GO < 225^\circ$
6	$225^\circ \leq GO < 270^\circ$
7	$270^\circ \leq GO < 315^\circ$
8	$315^\circ \leq GO < 360^\circ$

2. Run a connected-components algorithm to form line support regions from groups of 4-connected pixels that share the same shifted gradient orientation bin (as shown in Figure 2).
3. Eliminate line support regions that have an area smaller than a specified threshold. Given the line support regions shown in Figure 2 and an area threshold of three pixels, regions 3, 5, and 6 would thus be removed.
4. Repeat steps 1, 2, and 3 by shifting the gradient bins to produce a second set of line support regions. This accounts for the possibility that some true lines may have component pixels that lie on either side of an arbitrary gradient orientation boundary (e.g.  $45^\circ$  in Table I). Shifted partition bins are shown in Table II. The resulting gradient orientation partitioning and line support regions are shown in Figure 3.

TABLE II: Partitioning Pixels into Gradient Orientation Bins.

Bin Number	Gradient Orientation (GO)
1	$-22.5^\circ \leq GO < 22.5^\circ$
2	$22.5^\circ \leq GO < 67.5^\circ$
3	$67.5^\circ \leq GO < 112.5^\circ$
4	$112.5^\circ \leq GO < 157.5^\circ$
5	$157.5^\circ \leq GO < 202.5^\circ$
6	$202.5^\circ \leq GO < 247.5^\circ$
7	$247.5^\circ \leq GO < 292.5^\circ$
8	$292.5^\circ \leq GO < 337.5^\circ$

5. Use the following voting scheme [21] to select preferred lines from the two sets (i.e. original set and shifted set) of candidate lines:
  - a) Line lengths are determined for each region.
  - b) Because each pixel is a member of two regions (one in the line support region image and the shifted version), every pixel votes for and is associated with that region of the two that provides the longest interpretation.

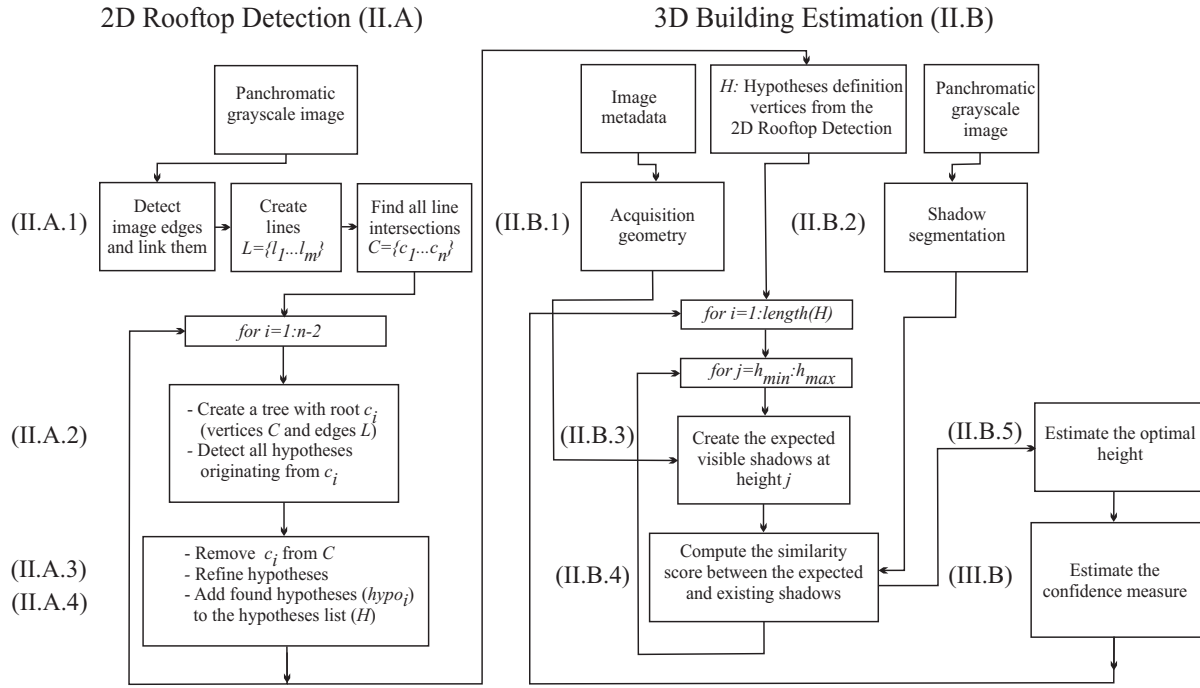


Fig. 1: Flowchart of the proposed system including two consecutive sub-systems.

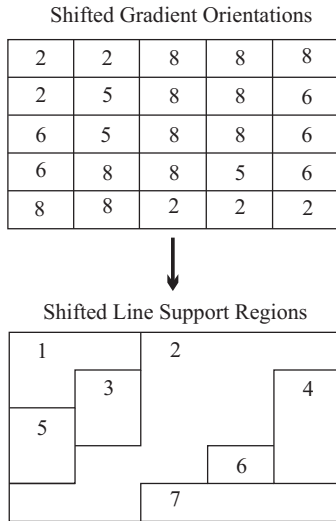


Fig. 2: Example of converting shifted gradient orientation bins to shifted line support regions.

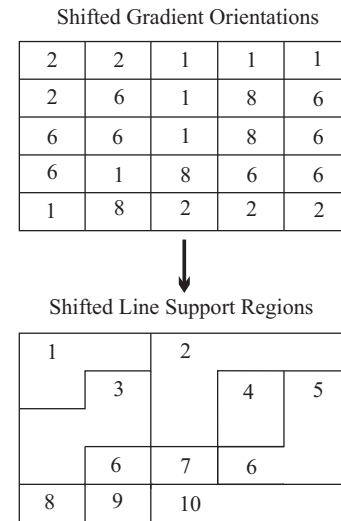


Fig. 3: Example of converting shifted gradient orientation bins to line support regions.

- c) Each region is associated with a count of the number of its pixel.
- d) Each region is given a support which is equal to the percentage of the total number of pixels voting for it. The regions selected are those that have a majority support (in this work the support that is greater than 50 percent).
6. For each line support region, compute the line represented by that region by performing a least squares fit. The least-squares fit estimates planar model of each line using the gradient magnitude

values. Figure 4 represents the fitted lines for the previous example case.

- **Line linking:** The objective of this step is to link collinear line segments that are separated by very small gaps. Following algorithm describes linking process:
  1. Sort the lines in the order they would be encountered if a horizontal sweep was performed across the image.
  2. Use a divide-and-conquer method to efficiently determine nearby pairs of lines.
  3. Test each pair of nearby lines to determine whether

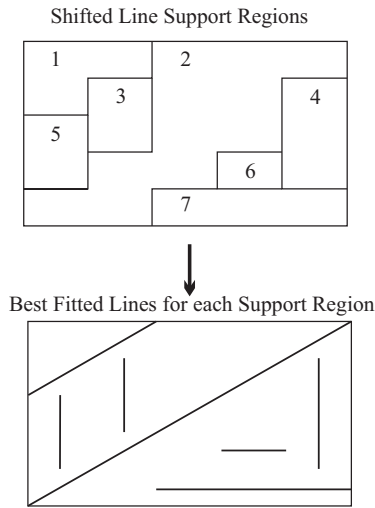


Fig. 4: Example of converting line support regions into lines.

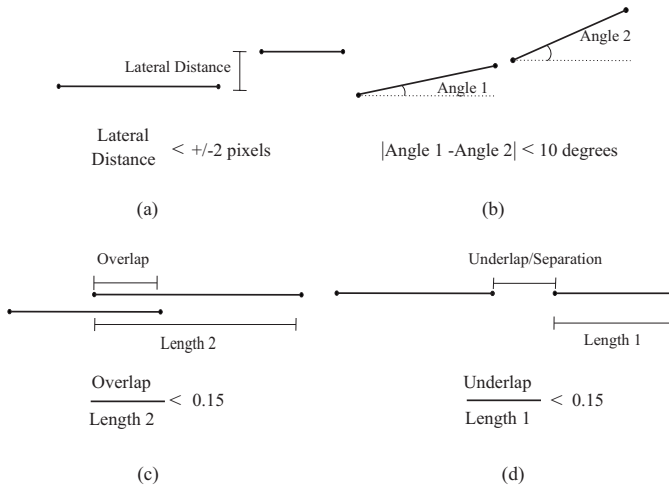


Fig. 5: Four criteria for line linking process.

they should be linked. Conditions (a) and (b) and one of the conditions of (c) and (d), which are illustrated in Figure 5, must be satisfied for a pair of lines to be linked.

Relaxing the threshold values in the line linking process can increase the number of detected lines by grouping lines that indeed are not connected. Such relaxation can increase the overall number of potential candidate rooftops. For instance, if threshold value of 0.15 in condition (d) is increased, hypotheses with two buildings in them might be generated. Increasing the threshold 10 degrees in condition (b) could cause hypotheses to have imprecise boundaries, due to the grouping of lines that are not co-linear. Through the filtering steps that are implemented in Section II-A3, many of such faulty hypotheses can be caught and removed. However, if the above conditions are too relaxed, chances of announcing a faulty hypothesis as a real building rooftop will increase.

- Line intersection detection: All *potential line intersections* are extracted from the image next. The term *potential* infers both intersections that occur within the image

and lines that they do not intersect in the image but their continuations do.

These intersections are used in creating an intersection graph in which each intersection represents a vertex with two edges. Detecting polygonal rooftop hypotheses corresponds to the problem of detecting loops in this graph. Figure 6 represents detected intersections and their corresponding edges in an image.

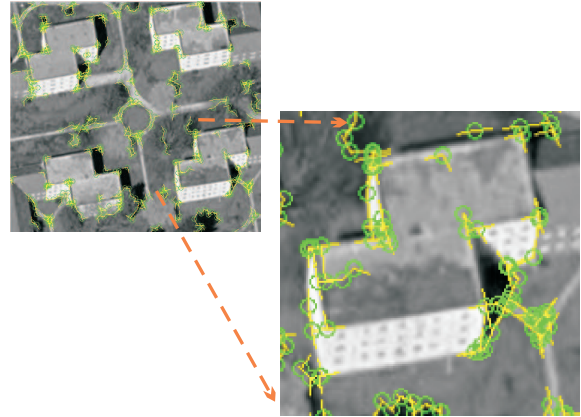


Fig. 6: Line intersections with their directions for a typical image. Green circles represent the intersections and yellow line segments represent the lines that have created these intersections.

2) *Hypothesis creation using graph:* Using graph-based search mechanism for the rooftop detection application has been presented previously. [22] reported a system in which corners on the right angle line intersections (junctions of form L, T, ...) were grouped together and used as nodes of the graphs. [23] also presented a graph-based search approach in their system for building detection. They used the graph concept to establish relationship between the lower level information like line segments and the higher level information like junctions and closed contours. In this work, we use a graph-based approach to exploit the relationship between line intersections in establishing polygonal shapes (rooftop definitions).

The above detected intersections are used in creating an intersection graph in which each intersection represents a vertex with two edges. Detecting polygonal rooftop hypotheses corresponds to the problem of detecting loops in this graph. In order to detect loops in the intersection graph, a dynamic programming approach is employed. Each loop is defined with a set of vertices and edges. Starting from a vertex, a path along one of the corner edges (called starting edge) is selected. A tube shape window with a width of  $w$  pixels (in this work  $w$  is set to 21 pixels), centered at the starting vertex and along the starting edge, is used to find all vertices with only one edge in the similar direction as the starting edge. Therefore for the first vertex ( $C_1$ ) and along one of its edges (starting edge), the algorithm finds all vertices with only one edge in the similar direction as the starting edge (in Figure 7,  $C_{11}$  and  $C_{12}$ ). Corners  $C_{11}$ ,  $C_{12}$ , ..., and  $C_{1m}$  represent the 1st level (or in general  $n + 1$ ) of the tree while  $C_1$  represents level 0

(or  $n$  in general). After finding all candidate vertices in the current tree level, a filtering process is performed to remove outlier vertices. Following conditions are employed as outlier rejection criteria,

1. Minimum distance between a vertex at level  $n$  with vertices at level  $n + 1$  must be greater than  $d_{min}$ .  $d_{min}$  is the minimum length of a rooftop side and is set to 20 pixels in this work.
2. Maximum distance between a vertex at level  $n$  with vertices at level  $n + 1$  must be smaller than  $d_{max}$ .  $d_{max}$  is a configurable parameter that represents the maximum length of building hypotheses. In this work (for all the results presented in Section III),  $d_{max}$  was set to 300 pixels.
3. There must exist a physical edge between the vertices at levels  $n$  and  $n + 1$ . The length of this edge must be smaller than the Euclidean distance between the two level vertices.

Once all outliers are removed, all edges of the corners at level  $n + 1$  are labeled as *in* or *out*. The labeling of the edges of vertices at level  $n + 1$  is always performed with respect to the direction of *out* edge of the vertex at level  $n$ . As shown in Figure 7, the edge with direction similar to the starting edge (maximum difference of  $45^\circ$  is allowed) is labeled as *in* and the other as *out*. In order to find a loop, for each vertex at level  $n$ , all candidate vertices at level  $n + 1$  are found and filtered. A value of 8 ( $N_{max}$ ) is utilized for the level parameter in this work, implying that the search for polygons with maximum 8 sides will be carried out. The algorithm continues with each

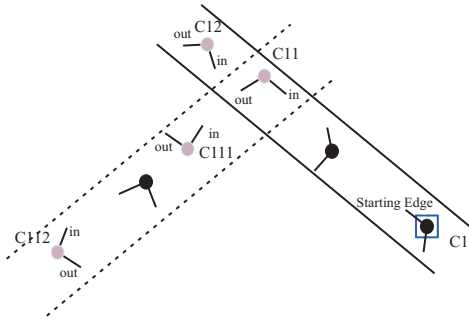


Fig. 7: Corner edges are indexed as *in* or *out*.

one of the candidate vertices at level 1. For each vertex (e.g. in Figure 7,  $C_{11}$ ), all candidate vertices with one edge in the similar direction as the *out* edge of the  $C_{11}$  are found and filtered (e.g. in Figure 7,  $C_{111}$  and  $C_{112}$ ). These vertices create level 2 (or in general  $n+2$ ) of the tree. The process of establishing a new level ( $n + 1$ ) of tree and associating corners to it according to the corner direction of the vertex at the current level ( $n$ ) is called the forward tracking process. The forward tracking process stops once  $C_1$  is met again. Meeting  $C_1$  for the second time (it was initially met at level 0) represents a loop that includes all possible vertices between the first and the second viewings of  $C_1$ . These vertices together define one potential rooftop candidate. If the  $N_{max}$  is reached before reaching to  $C_1$ , the process returns back to the previous

level and chooses another vertex from the candidate vertices of that level. This process is the backward tracking process.

Once the new vertex is chosen, the forward process will take over and one of the candidate vertices of this vertex in the next level is selected. In each backward tracking attempt, only one level lower is inspected. If however all candidate vertices are tried before, the backward tracking process will move to the one level before the previous level. Reaching to the lowest level (0) indicates that all potential rooftop hypotheses with  $C_1$  in them has been examined and the process should continue with the next vertex at the level 0 (e.g.  $C_1, C_2$ , and  $C_k$ ). At the end of this process, when all vertices of level 0 are examined, all potential rooftop candidates are identified. Figure 8 represents graphical representation of the intersection tree or graph.

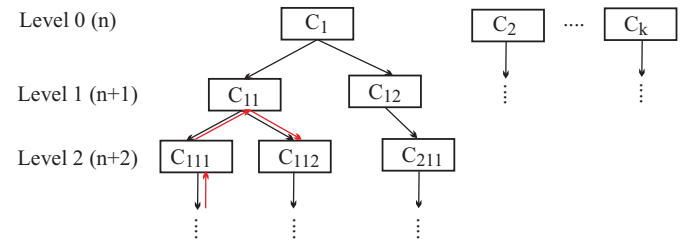


Fig. 8: Search tree levels are generated according to relationships between edges of corners. The black arrows represent the forward tracking path and the red arrows show a typical backtracking path.

The parameters in this section are set in accordance to the resolution of the input images. Initially, a number of input images were inspected to find the maximum and minimum length of typical buildings. The distribution of the building length for the inspected images showed that 95% of the times the sizes of the buildings of interest are within the range of 20 ( $d_{min}$ ) to 300 ( $d_{max}$ ) pixels.

The width of  $w$  ( $10 \times 2 + 1$  pixels) in the tube shaped search window was found by inspecting building boundaries. Sometimes, the rooftop of a building includes thick wedges (as can be noticed in the enlarged part of Figure 6) on its boundary. For such cases, chances are that two corners exist, one corresponding to the inner edge of the wedge and one to the outer edge of the wedge. The width of  $w$  allows choosing both such corners and creating all potential hypotheses combinations. Also, the location of the line intersection includes a hereditary uncertainty that originates from the precision of detected lines. Through a  $w$  of 21 pixels, a location deviation within a circle with radius of 10 pixels around each corner is permitted.

3) *Hypothesis refinement*: Naturally for each building instance, several candidate hypotheses are detected. Figure 9 shows the entire set of extracted candidate rooftop hypotheses for a sample scene. A two-step filtering scheme is implemented to remove weak and redundant hypotheses as followings:

1. Under the assumption of smoothness of the rooftop surface, the standard deviation of pixels intensities inside each hypothesis ( $\sigma_h$ ) is calculated. Only hypotheses with standard deviation smaller than 50 are kept. The value



Fig. 9: Detected candidate rooftop hypotheses for a sample scene.

of 50 was found by analyzing 10 input images and inspecting rooftops and regions around them. Figure 10(a) displays the results after this step.

2. Image intensities inside and outside hypotheses are compared against each other. The mean values between the interior and exterior pixels of a rooftop are computed. The difference percentage is computed by  $|mean_{in} - mean_{out}|/mean_{in}$  and only rooftops with a ratio larger than 20% are kept. The exterior pixels are chosen by extending the rooftop boundary towards outside by 40 pixels. Figure 10(b) depicts the results after this process.



Fig. 10: (a) Rooftop hypotheses after first (a) and second (b) elimination steps for a sample scene.

4) *Hypothesis retrieval*: Occasionally, true hypothesis corresponding to an actual building would be removed due to the sensitivity to parameters setting (as shown in Figure 10(b) for the bottom-left building). To recover such candidates, hypotheses that have overlaps with each other are grouped into one group. The number of created groups defines the potential number of rooftop hypotheses in the image. If after the previous 2-step refinement, the number of detected hypotheses is smaller than the number of identified groups, a recursive local retrieving process will be initiated. To ensure that the true hypotheses are not removed, due to the value setting of the global parameters, the sensitivity of threshold parameters in the filtering process will be automatically adjusted. These parameters include:

- The threshold related to the standard deviation of pixels inside the rooftop boundaries ( $\sigma_h$ ) from Section II-A.3.1,

and

- The percentage of the mean intensity variation between the regions inside and outside the rooftop boundaries from Section II-A.3.2.

In each successive iteration,  $\sigma_h$  is increased by 10% (e.g. in the first iteration a value of 55 pixels and in the second iteration a value of 60 pixels) and the percentage of the mean intensity variation is decreased by 10% (e.g. in the first iteration a value of 18% and in the second iteration a value of 16%). If after 5 successive iterations, no hypothesis is obtained, the missing group will be eliminated and otherwise a new hypothesis will be added to the detected list. Figure 11(a) displays a group of hypotheses that were dismissed in the first run of the algorithm and Figure 11(b) shows the retrieved hypothesis.



Fig. 11: Missing hypothesis is retrieved by fine tuning of the system global parameters.

While this process is designed to retrieve missing hypotheses, it could potentially add to the number of false positives (cases where wrong regions are identified as rooftops). In the data set used in this work we did not observe such a case; however, if images with poorer qualities are used, the above suggested retrieval may cause detection of false positive rooftops.

Figure 12 highlights extracted hypotheses for the sample satellite image. More results, including both quantitative and qualitative evaluations are presented in Section *Experimental Results*.



Fig. 12: Final detected hypotheses for a typical scene image.

5) *Discussion*: The presented rooftop detection algorithm was originally designed for QuickBird satellite imagery (0.6 meter/pixel). Later the algorithm was tested on gray scale

aerial imageries (Pictometry, 0.15 meter/pixel). After adjusting some of the input parameters ( $d_{min}$ ,  $d_{max}$ ,  $w$ , Line linking conditions  $a$  to  $d$ ) for the higher image resolution, good results were obtained. Unfortunately same cannot be said if the quality of the image is worse or the image resolution is less than 0.6 meter/pixel.

### B. 3D Building Estimation

This section describes steps involved in creation of 3D models of buildings corresponding to rooftops detected in Section II-A. The main idea is to identify casted shadows and use them to estimate the height of buildings. Utilizing shadows in estimating building heights requires two sets of information: 1) the acquisition and sun geometries, and 2) the length of the shadows of a building on the ground. The sun geometry provides us with the location of the sun and therefore the direction of the expected shadows on the ground. The acquisition geometry provides us with information such as 3D location of the camera system and the heading, scale, and calibration parameters. Acquisition geometry information is necessary in predicting the building walls and base area especially when the viewing is not perpendicular to the surface of the earth. To measure the length of the shadows, first the shadow areas in the image must be identified. Clearly taller buildings project longer shadows. Also as the sun moves towards the horizon, the area (and the length) of the shadow will increase. Accurate identification of shadows in each image, therefore is a requirement for correct estimation of building heights. In this section we describe the acquisition geometry and the shadow segmentation method first. We will then explain the proposed approach for estimating the height using the acquisition geometry, shadow regions, and the rooftop definition.

1) *Acquisition geometry*: The acquisition geometry describes the geometrical relationships between the sun, the sensor and the horizontal plane at the target location. The image metadata contain a large amount of information among which only the sensor and the sun azimuth and elevation angles are used for this work.

We adopted the acquisition geometry proposed by Huang and Kwo [24] which handles the normal viewing of the input imageries. Figure 13 shows the geometry of the system of rays, when a 3D building is imaged. From this figure, the top point of the building in the 3D world is projected on the  $p(l, s)$  of the 2D satellite image, while its base is located at  $b(l_b, s_b)$ . The shadow of this point is casted at point  $s(l_s, s_s)$ . By measuring the length of  $L_{pb}$  and the length of  $L_{sb}$  in the 2D image, the height of the building ( $h$ ) can be estimated by:

$$h = \tan\lambda \cdot L_{pb} \quad (1)$$

$$h = \tan\lambda' \cdot L_{sb} \quad (2)$$

$$L_{ps}^2 = L_{pb}^2 + L_{sb}^2 - 2L_{pb} \cdot L_{sb} \cdot \cos(\alpha - \alpha') \quad (3)$$

$$h = \frac{L_{ps}}{\sqrt{\frac{1}{\tan^2\lambda} + \frac{1}{\tan^2\lambda'} - \frac{2\cos(\alpha - \alpha')}{\tan\lambda \cdot \tan\lambda'}}} \quad (4)$$

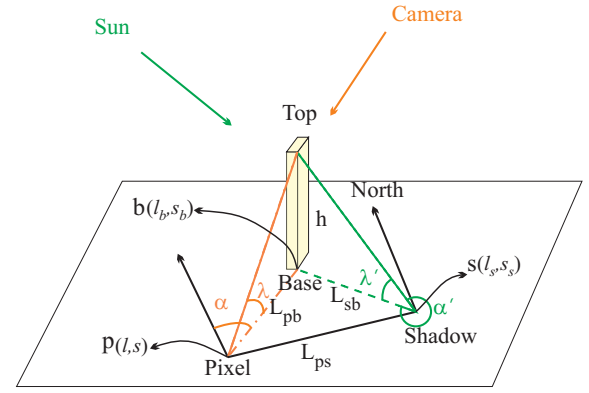


Fig. 13: The geometry of the system of rays from the sun and the camera position when a 3D building is imaged.

Here the azimuth and elevation angles of the camera and the sun are  $\alpha$ ,  $\lambda$ ,  $\alpha'$ , and  $\lambda'$  respectively. The described acquisition geometry is implemented to project the shadows of the rooftop vertices (at any given height) onto their corresponding locations on the ground.

2) *Shadow segmentation*: Many shadow segmentation approaches rely on threshold values to separate shadow from non-shadow regions [24]–[27]. These methods could suffer from inaccuracies when encountering variant shadow intensities that may exist under natural varying/non-uniform illumination conditions. Tsai [28] utilized a segmentation method in an automatic de-shadowing approach for shadow detection compensation in color aerial images. He employed spectral ratio values with an automatic thresholding technique to detect shadows. He showed that shadow detection using *HSI* color space has very high accuracy. Therefore we utilize the spectral ratio of  $(H + 1)/(I + 1)$  to construct a ratio image. In the ratio image, pixels corresponding to shadow areas have higher values than those of others. The ratio image is then segmented using the Mean Shift Segmentation algorithm [29]. The output of this stage includes a set of potential shadow regions  $R_i$ .

3) *Expected shadow prediction*: It is important to establish the difference between existing shadows on the ground (detected in Section II-B2) and expected shadows due to a specific building height. This section provides details on estimating the expected shadows corresponding to a building at a specific height. The geometry of a rooftop, in general, could create partial occlusions on some of the projected shadows of the building on the ground. The procedure for estimating the visible expected shadows of a building can be explained by the following steps:

- In the first step, visible wall regions are computed. To create the visible wall regions corresponding to a rooftop at a given height, first the vanishing point (using the intersection of all vertical lines) is computed. Next, all vertices of the rooftop are projected onto the ground in the direction of the vanishing point using equation (1). The projected vertices are then connected together (in the same order as the rooftop) to create the base of the building ( $R_{base}$ ). By changing the height of a building from zero to a given height ( $h$ ) (with steps of  $\Delta h$ ) and



projecting the base region in each case, regions associated with the walls are identified as shown in Figure 14(b). These regions include invisible parts of the walls that are occluded by the rooftop. Therefore, by removing the rooftop, the visible wall regions are calculated.

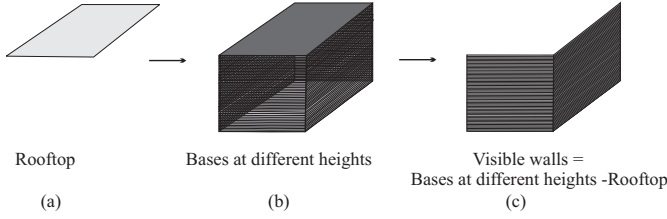


Fig. 14: Visible walls are identified using base projection at different height values.

The above procedure can be summarized by:

$$R_{wall} = \text{OR}[f_{base}(R_{rooftop}, \forall h \in \{0, \Delta h, \dots, h\}, P)] - R_{rooftop} \quad (5)$$

In this equation "OR" presents the union operation and "-" represents the logical subtraction. Function  $f_{base}$  computes the base region associated with a building at height  $z$  using the rooftop definition ( $R_{rooftop}$ ) and acquisition geometry ( $P$ ).

$$f_{base}(R_{rooftop}, z, P) \longrightarrow R_{base} \quad (6)$$

- To predict the visible part of the expected shadow at any height, first the unit vector of the sun direction is computed. This vector represents the direction along which shadow points on the ground are projected. The shadow regions are generated by the rooftop definition and wall areas that naturally block the sun rays. The location of the projected shadow associated with a building point depends on the location of that point, its height and the acquisition geometry of  $P$  (computed using equation (2)). To estimate the projected shadow of a building, we propose an incremental approach that is fast and easy to implement. Given a candidate height of  $h$ , the rooftop definition of  $R_{rooftop}$ , and the acquisition geometry of  $P$ , the wall regions are estimated using equation 5 first. At height  $h$ , the projected shadow of the rooftop boundaries on the ground are identified by projecting all vertices of that rooftop along the sun direction onto the ground. This creates a polygonal region on the ground that, partially, represents the expected shadow of the building at that height. Generally, this region has some overlap with the rooftop area. The non-overlapping part of this region, however, corresponds to the visible projected shadow at height  $h$ . The remaining parts of the shadow originate from the walls. To add the contribution of the walls to the expected shadow, the base region profile (polygonal presentation) at each height (incremental values of  $\Delta h$  from 0 to  $h$ ) is computed and projected along the sun direction onto the ground. The region corresponding to

the logical sum of the projected shadows by the rooftop and all the base regions is then estimated by:

$$R_{bases \text{ at all heights}} = \text{OR}[f_{shadow}(f_{base}(R_{rooftop}, \forall z \in \{0, \Delta h, \dots, h\}, P), h - z, P)] \quad (7)$$

To compute the bases at all heights from zero to  $h$ , parameter of  $z$  is used in the above equation that holds an incremental value of the height. At the beginning this parameter is zero and therefore the base at height  $h$  is computed. As  $z$  increases towards the value of  $h$ , projection of the base on the ground at different heights is estimated.  $R_{bases \text{ at all heights}}$  includes overlaps with the rooftop and the wall regions; therefore rooftop and wall regions must be removed (logical subtraction) to estimate the complete visible shadow of the building at height  $h$ . Equation 8 highlights this procedure.

$$R_{shadow} = \text{OR}[f_{shadow}(f_{base}(R_{rooftop}, \forall z \in \{0, \Delta h, \dots, h\}, P), h - z, P)] - R_{rooftop} - R_{wall} \quad (8)$$

In our implementation  $\Delta h$  was set to half image pixel size or 0.3 meter. Here  $f_{shadow}$  computes the shadow of the building at a specific height of  $z$  using the acquisition geometry of  $P$  in the sun direction.

$$f_{shadow}(R_{rooftop}, z, P) \longrightarrow R_{shadow} \quad (9)$$

Figure 15 represents the shadow estimation results at different stages. In this example, the estimation is performed at the exact height of the building. Figure 15(a) shows a rooftop that is highlighted with red contour. The green dots represent the vertices of this rooftop. Figure 15(b) depicts the region generated by the first term in equation 5. Figure 15(c) displays the results of equation 5 or  $R_{wall}$ . Figure 15(d) depicts results generated by the OR term in equations 8 and Figure 15(e) highlights the estimated shadow regions found by equation 8.

Ideally, in the absence of occlusion by other surrounding buildings or objects,  $R_{shadow}$  will be exactly the same as the existing shadow of that building in the image at its true height. Since  $R_{shadow}$  changes at various heights, it must be recalculated at each candidate height. Next section describes details of the fitness function that is used to assess the similarity of predicted shadows with the existing ones.

4) *Fuzzy rule based fitness*: In this section an evaluation function using fuzzy rules is introduced. The purpose of this function is to measure the similarity between existing shadows in the image and all predicted shadows of a building at different heights. Naturally, the height associated with the most similar region is chosen as the estimated height of the building.

Justification for using a fuzzy-based approach originates from two facts. Firstly, the intensity of the shadow regions varies according to the amount of indirect sky light and the reflective properties of the regions or objects within them. Therefore, in large satellite/aerial images shadows have different characteristics across each image. Assuming a fuzzy nature for shadows is not an unrealistic assumption. Secondly, casting

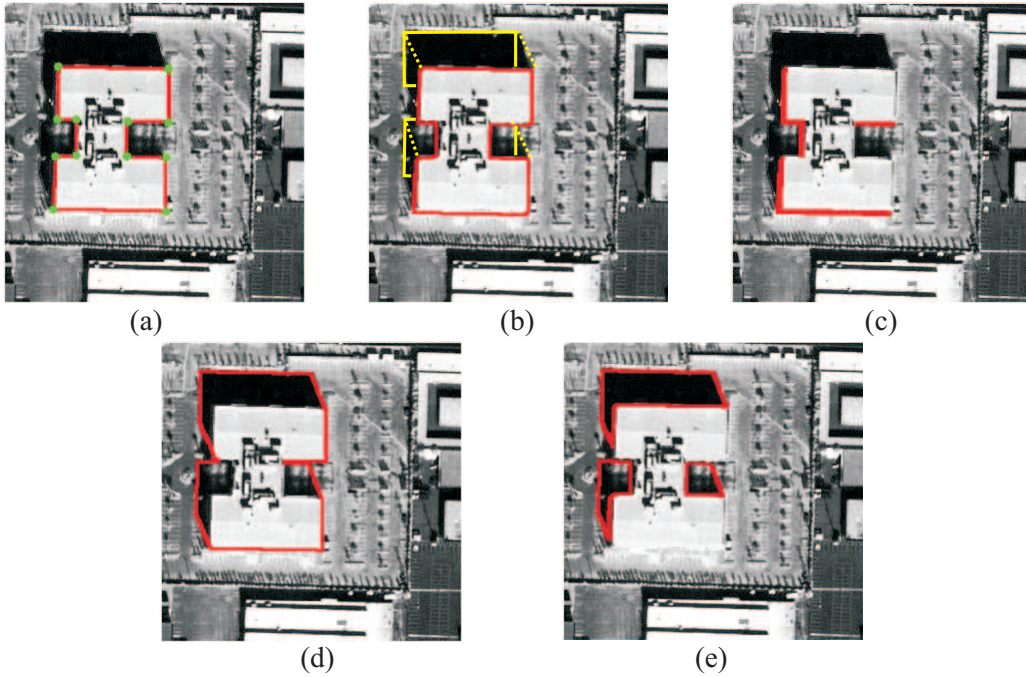


Fig. 15: (a) A polygonal rooftop defined by a set of vertices. (b) Base regions at height zero (red) and  $h$  (yellow) are shown and the dotted lined show the locations where vertices at heights between zero and  $h$  are projected onto. (c) Wall regions. (d) Projection of all expected shadow regions for height  $h$ . (e) Visible expected shadow regions of the building.

shadows of buildings are related to the geometrical shape characteristics of these buildings. Depending on the building shapes and neighboring buildings or objects, the projected shadows of a building could partially be obstructed (by objects or their shadows). Therefore, even for a precisely known height value, the expected shadows could be only partially similar to the existing shadows in the image.

The proposed fuzzy assessment function is based on shadow properties of the spectral ratio segments ( $R_{is}$ ) and shape characteristics of the expected shadow regions ( $R_{shadow}$ ) at a given height. The result of assessment function is a *Height Score*. Given a set of candidate heights for a building, the height score is estimated for each height and the height associated with the highest *Height Score* is chosen as the estimated building height.

Since the proposed fuzzy function is based on the building shape and shadow properties, two fuzzy sets input variables (Spectral Ratio or SR and Shape Fitness or SF) are defined for the function. Table III lists labels and variables for the fuzzy function.

TABLE III: Linguistic variables and labels for the fuzzy rule-based fitness function

	Linguistic Variable	Linguistic Label
Input	Spectral Ratio	Small, Large
	Shape Fitness	Small, Medium, Large
Output	Score	Negative Large, Negative Small, Moderate, Positive Small, Positive Large

To evaluate the similarity between building expected shadow

regions and the existing shadow in an image, we need to validate the shadow property of pixels inside the expected shadow regions. As described in Section II-B2, in the ratio image, a pixel with higher spectral ratio value has a higher probability of belonging to a shadow region. Therefore, an expected shadow region with a higher mean spectral ratio value has a higher probability to belong to a real shadow region than another region with a lower mean. In our fuzzy assessment function, we have defined a variable, Spectral Ratio (SR), to present pixels shadow property. We classify image pixels in two classes of shadow and non-shadow pixels. Shadow pixels have large spectral ratio values and non-shadow pixels have small spectral ratio values. We define two membership functions  $\mu_1(x)$  and  $\mu_2(x)$  for the spectral ratio variable. We also assume normal distribution for both of these membership functions, as we are dealing with a natural phenomenon.

Hence, spectral ratio image pixels are clustered into two classes by fuzzy c-means clustering method [30] and the mean and standard deviation of the two clusters are computed. The smaller mean value is called  $c_1$  and its corresponding standard deviation multiplied by 2 is named  $\sigma_1$ . Also  $c_2$  and  $\sigma_2$  are assigned to the larger mean value and its corresponding standard deviation multiplied by 2, respectively. The choice of twice standard deviations of the two clusters for  $\sigma_1$  and  $\sigma_2$  was to guarantee that there is no gap between the two Gaussian functions. This prevents ignoring those pixels on the two ends of the shadow and non shadow boundaries. The following Gaussian models are used to define  $\mu_1(x)$  and  $\mu_2(x)$ :

$$\mu_1(x) = \begin{cases} e^{\frac{-(x-c_1)^2}{2\sigma_1^2}} & \text{if } x \geq c_1 \\ 1 & \text{if } x < c_1 \end{cases} \quad (10)$$

and

$$\mu_2(x) = \begin{cases} e^{\frac{-(x-c_2)^2}{2\sigma_2^2}} & \text{if } x \leq c_2 \\ 1 & \text{if } x > c_2 \end{cases} \quad (11)$$

Here  $x$  represents the mean value of a region in the ratio image. A large value of  $\mu_1(x)$  implies that the spectral ratio of the region is small, indicating lower probability of that region to be part of the shadow region. Large values of  $\mu_2(x)$  indicate that the spectral ratio of the region is large and the region most likely belongs to the shadow segment. Figure 16 represents a graphical representation of the two Gaussian models.

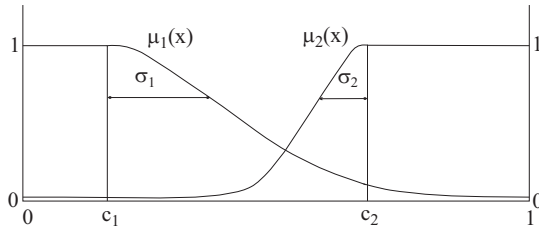


Fig. 16: Graphical representation of the membership functions  $\mu_1(x)$  and  $\mu_2(x)$ .

The second set of the fuzzy input variable is the Shape Fitness (SF) variable. This variable is designed to present the shape similarity between the expected shadow regions and existing image shadows. We expect that the expected and the existing shadow regions are fully or partially (in the case of occlusion by other objects or shadows) matched. We have considered three values for the Shape Fitness variable: Small (when the expected and existing shadow regions have no or very little overlap), Medium (when regions partially overlap) and Large (when regions fully overlap). Therefore, three triangular membership functions  $f_1(x)$ ,  $f_2(x)$  and  $f_3(x)$  are defined for the shape fitness set (Figure 17(a)) such that they do overlap with each other. We have used a triangular shape for the shape fitness function since it is easy to implement and it provides good accuracy as later shown in Section III. The implemented membership function  $f_1(x)$ ,  $f_2(x)$  and  $f_3(x)$  are detailed in equation 12.

$$f_1(x) = \begin{cases} 1 & x \leq 0 \\ 1 - \frac{3}{2}x & 0 < x \leq \frac{3}{2} \\ 0 & \frac{3}{2} < x \end{cases}, \quad (12)$$

$$f_2(x) = \begin{cases} 1 - 2|x - \frac{1}{2}| & 2|x - \frac{1}{2}| \leq 1 \\ 0 & 2|x - \frac{1}{2}| > 1 \end{cases},$$

$$f_3(x) = \begin{cases} 1 & 1 \leq x \\ \frac{3}{2}x - \frac{1}{2} & \frac{1}{3} \leq x < 1 \\ 0 & x < \frac{1}{3} \end{cases}.$$

Table IV represents the fuzzy rules for generating the output variable  $Score$  ( $Score_{R_i}$ ). As shown in this table, there are six different combinations for the two input fuzzy sets. Among these combinations, two (no. 1 and no. 4) are assessed equally (where the Shape Fitness is small). This reduces the number of score variables into five. Each combination represents a unique case as described below:

1. Negative Large: Non-shadow pixels exist in the expected shadow region and the expected and existing shadow regions fully overlap.
2. Negative Small: Non-shadow pixels exist in the expected shadow region and the expected and existing shadow regions only partially overlap.
3. Moderate: The expected and existing shadow regions have very small overlaps.
4. Positive Small: Shadow pixels exist the expected shadow region and both regions partially overlap.
5. Positive Large: Shadow pixels exist in the expected shadow region and both regions fully overlap.

The five membership functions  $g_1(x)$  to  $g_5(x)$  for the output Score variable ( $Score_{R_i}$ ) are presented in equation 13 and Figure 17(b)).

$$g_1(x) = \begin{cases} 1 & x \leq -1 \\ -1 - 2x & -1 < x \leq -\frac{1}{2} \\ 0 & -\frac{1}{2} < x \end{cases},$$

$$g_2(x) = \begin{cases} 1 - 2|x + \frac{1}{2}| & 2|x + \frac{1}{2}| \leq 1 \\ 0 & 2|x + \frac{1}{2}| > 1 \end{cases},$$

$$g_3(x) = \begin{cases} 1 - 2|x| & 2|x| \leq 1 \\ 0 & 2|x| > 1 \end{cases}, \quad (13)$$

$$g_4(x) = \begin{cases} 1 - 2|x - \frac{1}{2}| & 2|x - \frac{1}{2}| \leq 1 \\ 0 & 2|x - \frac{1}{2}| > 1 \end{cases},$$

$$g_5(x) = \begin{cases} 1 & 1 \leq x \\ 2x - 1 & \frac{1}{2} \leq x < 1 \\ 0 & x < \frac{1}{2} \end{cases}.$$

For a predicted shadow of a building ( $R_{shadow}$ ) at a candidate height  $h$ , all regions  $R_i$  (extracted in Section II-B2) that partially or fully overlap with  $R_{shadow}$  are extracted from the segmented ratio image and placed into the set  $S$ . For each region  $R_i$  in  $S$ , two parameters  $m_{R_i}$  and  $v_{R_i}$  are then estimated.

$m_{R_i}$  represents the mean value of  $R_i$  in the segmented ratio image.  $v_{R_i}$  denotes the overlap between  $R_i$  and  $R_{shadow}$  and is computed from:

$$v_{R_i} = \frac{Area(R_{shadow} \cap R_i)}{Area(R_i)} \quad (14)$$

To compute a score for  $R_i$ , a fuzzy inference method is required to process and deduce the fuzzy rules. Two of the common methods for the fuzzy inference are proposed by Mamdani [31] and Takagi-Sugeno [32]. In this work we have adopted Mamdani's method since it works well when the rule

TABLE IV: Fuzzy rules for generating output variable Score.

- |  |
|--|
| <ol style="list-style-type: none"> <li>1. IF Spectral Ratio is Small AND Shape Fitness is Small THEN Score is Moderate.</li> <li>2. IF Spectral Ratio is Small AND Shape Fitness is Medium THEN Score is Negative Small.</li> <li>3. IF Spectral Ratio is Small AND Shape Fitness is Large THEN Score is Negative Large.</li> <li>4. IF Spectral Ratio is Large AND Shape Fitness is Small THEN Score is Moderate.</li> <li>5. IF Spectral Ratio is Large AND Shape Fitness is Medium THEN Score is Positive Small.</li> <li>6. IF Spectral Ratio is Large AND Shape Fitness is Large THEN Score is Positive Large.</li> </ol> |
|--|

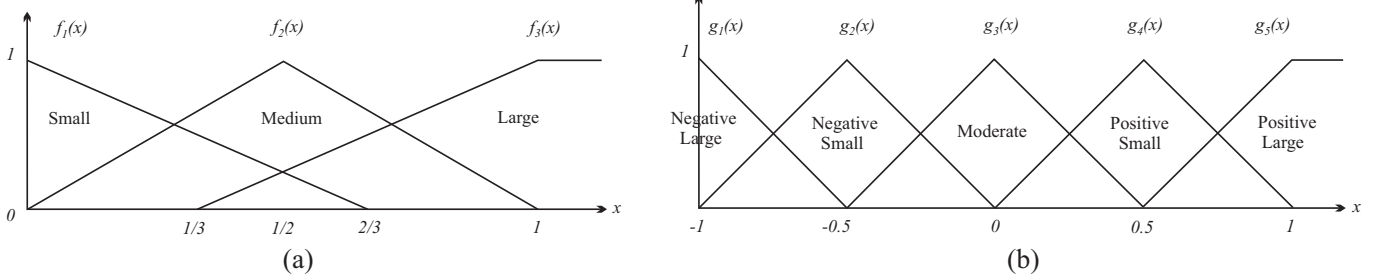


Fig. 17: Membership functions of Shape Fitness (a) and Score variables (b).

base presents a static mapping between the input and the output (such as the one presented in this work). The static mapping here refers to the fact that the rules are fixed during the process and are not added/modified/removed. Mamdani's fuzzy inference method involves five processing steps including (in order): input fuzzification, fuzzy operator application, implication, finding maximum output and defuzzification.

In the first step, each input is fuzzified over all the qualifying membership functions as required by the rules shown in Table IV. For instance, for the rule no. 1, the two inputs  $m_{R_i}$  and  $v_{R_i}$  are fuzzified using  $\mu_1(m_{R_i})$  and  $f_1(v_{R_i})$ . Next a fuzzy operator is applied on the fuzzified inputs for each rule. Here, minimum and maximum functions are used in the place of logical AND and OR, respectively. Therefore, the strength of each rule is computed by:

$$\begin{aligned}
 h_1 &= \min(\mu_1(m_{R_i}), f_1(v_{R_i})), & h_2 &= \min(\mu_1(m_{R_i}), f_2(v_{R_i})) \\
 h_3 &= \min(\mu_1(m_{R_i}), f_3(v_{R_i})), & h_4 &= \min(\mu_2(m_{R_i}), f_1(v_{R_i})) \\
 h_5 &= \min(\mu_2(m_{R_i}), f_2(v_{R_i})), & h_6 &= \min(\mu_2(m_{R_i}), f_3(v_{R_i}))
 \end{aligned} \tag{15}$$

In the third step, the implication is applied on each rule. The strength of each rule represents the amount by which a rule is satisfied. It is a number between 0 and 1. A strength value closer to 1 indicates that the rule is more satisfied or closer to the truth. By definition, the implication of each rule is the conclusion (or the logical judgment) made based on the strength of that rule. If the strength is closer to 1 (true) the implication of that strength is closer to the predefined output of the rule (*Score* variable in Table IV). Here, the computed strength of a rule ( $h$ ) is compared against the output function of each rule ( $g$ ). The minimum function is adopted for this purpose (as suggested by [31]). Therefore, the output of the implication for each rule is the minimum value between the rule strength ( $h$ ) and the output function of each rule ( $g$ ). As shown in Figure 17(b), the range of  $g$  is  $[-1, 1]$ . Therefore the range of values for  $D_1$  to  $D_6$  is within  $[-1, 1]$ .

$$\begin{aligned}
 D_1(x) &= \min(h_1, g_3(x)), & D_2(x) &= \min(h_2, g_2(x)) \\
 D_3(x) &= \min(h_3, g_1(x)), & D_4(x) &= \min(h_4, g_3(x)) \\
 D_5(x) &= \min(h_5, g_4(x)), & D_6(x) &= \min(h_6, g_5(x))
 \end{aligned} \tag{16}$$

In the fourth step, the total contribution from all rules is calculated for each candidate height  $h$ :

$$C(x) = \max(D_1(x), \dots, D_6(x)) \tag{17}$$

In the last step, the output fuzzy set is defuzzified by computing the centroid of the output fuzzy set. This centroid value represents the score of region  $R_i$ :

$$S(R_i) = \frac{\int_{-1}^1 xC(x)dx}{\int_{-1}^1 C(x)dx} \tag{18}$$

In this work, the integral terms in equation 18 have incremental steps of 0.01. Finally, the *Height Score* is computed for the shadow region  $R_{shadow}$  by calculating the mean of computed scores,  $S(R_i)$ , for all  $R_i$  in the set  $S$ . Each computed score of  $S(R_i)$  is weighted by the percentage of overlap between  $R_i$  and  $R_{shadow}$ :

$$Height\ Score = \frac{\sum_{\forall R_i \in S} Area(R_{shadow} \cap R_i) \times S(R_i)}{\sum Area(R_{shadow})} \tag{19}$$

Note that in the above equation the *Height Score* represents the similarity between the existing shadows in the image and the expected shadow at a specific height for a building. Also note that the contribution  $C(x)$ , the intermediate scores  $S(R_i)$  and the final *Height Score* are computed for every candidate height  $h$ .

5) *Optimal height estimation*: Traditionally when estimating a building height, an estimate goodness (*HeightScore*) is computed for an incremental range of height values from which the best score is chosen [10], [33]. For our system the *HeightScore* for values between 2 meters to 60 meters are computed. An incremental step of 0.3 meter is utilized in accordance with the QuickBird satellite image resolution (0.6 meter/pixel). Therefore estimating the height of a building requires computing the *HeightScores* 193 times.

In an attempt to improve the efficiency of the height estimation process, the Genetic Algorithm [34], [35] (GA) was utilized. GA is based on a population of candidate solutions that evolves toward the best height estimate. To set the GA's population setting  $N$ , the formula proposed in [36] is incorporated:

$$N \approx \lceil 1 + \log_2\left(\frac{-l}{\ln(\alpha)}\right) \rceil \quad (20)$$

In this equation,  $N$  represents the number of GA population,  $l$  is the variable length in bits and  $\alpha$  is the reliability factor (the GA convergence probability). In this work, building height is represented by an 8 bit variable ( $l = 8$ ) with a reliability of 90% ( $\alpha = 90\%$ ). Placing these two values in equation 20 results in the value of 12 for  $N$ . The maximum number of generations in GA is set to 10. Therefore, when estimating a building height at most  $12 \times 10$  individuals (heights) are created and evaluated. We found that the accuracy of the two methods are very similar for 61 out of 62 cases (mean difference of 0.1 meter). There was one case in which the estimated height through GA was wrong by 13 meters. Although the processing time was improved using the GA by 37%, we ruled out the use of GA in this case. Therefore all estimated heights (presented in the next section) are estimated using uniform increments from minimum to maximum height values.

### III. EXPERIMENTAL RESULTS

The performance of the system is assessed using 20 QuickBird satellite images. To prevent the propagation of the errors from the 2D rooftop detection process and to verify the accuracy of the height estimation independently, false rooftop hypotheses are removed manually before they are processed by the 3D estimation procedure. The ground truth in each case is prepared manually.

#### A. 2D Rooftop Detection

Figure 18 represents several examples in which rooftop boundaries overlaid over input images. To evaluate the proposed method quantitatively, three metrics are used: Detection

Rate (DR), False Negative Rate (FNR), and the McKeown shape accuracy factor [37].

$$DR = \frac{N_{TP}}{N_{TP} + N_{FP}} \quad , \quad FNR = \frac{N_{FN}}{N_{FN} + N_{TP}} \quad (21)$$

Here,  $TP$ ,  $FP$  and  $FN$  represent True Positives, False Positives and False Negatives in each scene. To calculate the McKeown's factor, the areas of buildings in the ground truth ( $A_{GT}$ ) is compared against the areas of the automatically detected buildings by our algorithm ( $A_{DB}$ ).

$$Shape \ accuracy = 1 - \frac{|A_{GT} - A_{DB}|}{A_{GT}} \times 100 \quad (22)$$

Table V summarizes results for all test images. The mean shape accuracy of the proposed method is 94.1%.

TABLE V: Summary of the detection results.

No. of Scenes	Total No. of Buildings	No. of TP	No. of FP	No. of FN	Mean Shape Accuracy %	Mean FNR %
20	70	62	6	8	94.1	11.08

In computing these results, partial buildings at image boundaries are not processed. The ground truth data were prepared by manual identification of the rooftop boundaries. Also buildings with at least one dimension smaller than  $d_{min}$  were excluded from the detection process. Table VI compares the performances of some of the previous works (as reported in the literature) with that of the proposed method.

TABLE VI: Comparison of the average DR.

Method	Wei [38]	Lin [10]	Jin [39]	Wei [40]	Nevatia [11]	Our method
Average DR	68.9%	71.9%	72.7%	84.3%	95.45%	95.2%

From this table, it can be observed that the proposed method delivers an average detection rate that is much better than those presented by [10], [38]–[40]. The presented result is close to the one presented by [11]. The proposed method however has the advantage of detecting general polynomial shapes which are not limited to simple rectilinear profiles. The proposed method also assumes that the rooftops are flat or flat looking. Under even lighting illumination condition with sun at nadir, often gabled rooftop look flat and the algorithm can detect them correctly. If however the sun is closer to the horizon, the intensity variation on different components of a gabled rooftop could cause the algorithm to detect these parts as separate rooftops, or miss one or more such components. This issue

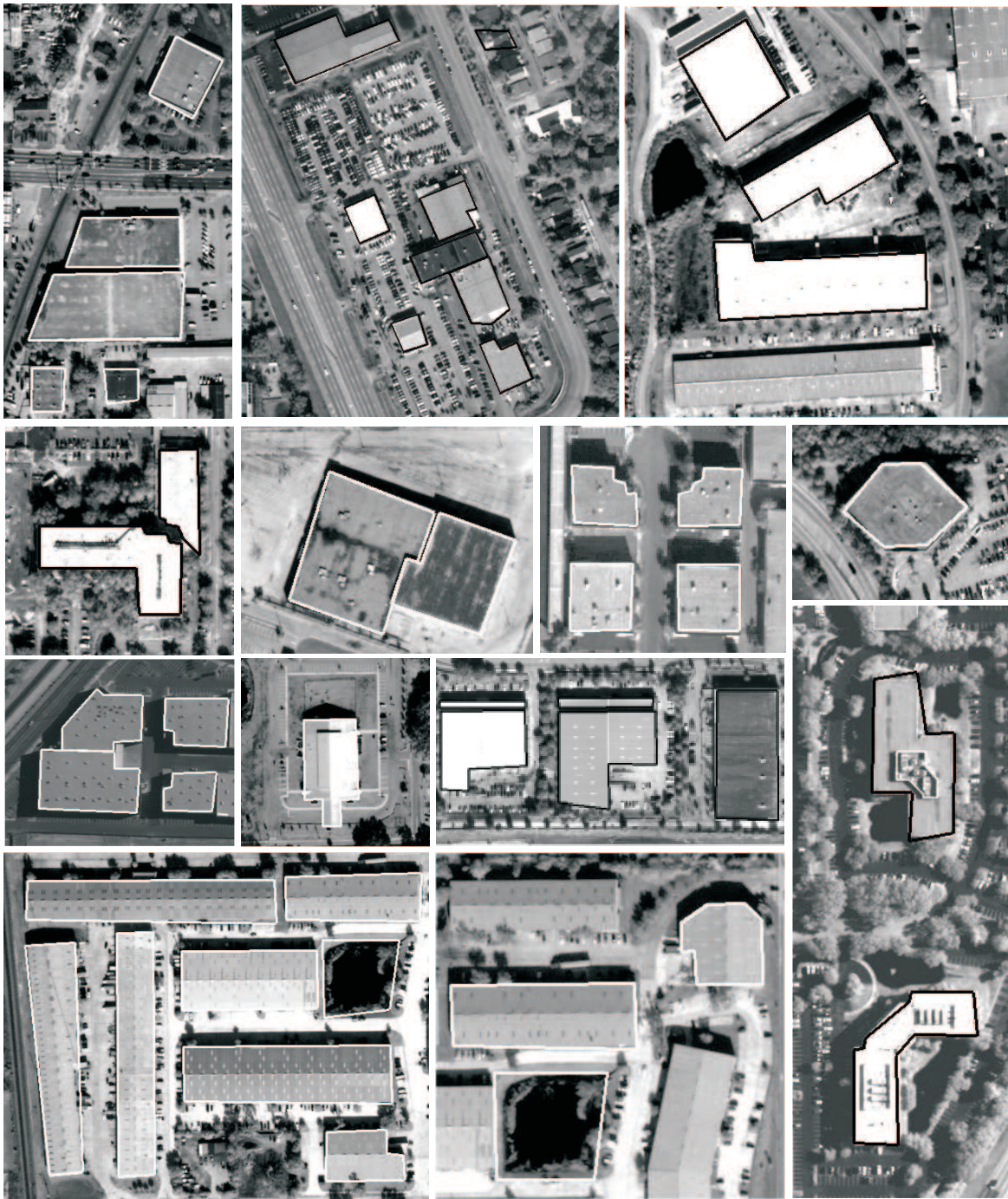


Fig. 18: Examples of the output results for the proposed 2D rooftop detection algorithm.

however is a general problem in most of the previous works especially if single input imagery is utilized.

It must also be noted that the presented average DR results, in Table VI, are adopted directly from the results presented in

each of these works. Unfortunately, each researcher has used his/her own set of images. Moreover, all of these algorithms are very complicated and do not have open sources which made it impossible for us to assess the DR rate for these

methods based on the input images that were used in this system.

### B. 3D Estimation

Each rooftop definition is presented by a vector of vertices in a connected loop format. Using the methodology described in Section II-B, the best estimated height value for each building is computed. Clearly, the quality of shadows and specific characteristics of buildings in the input image will affect the performance and reliability of the estimated heights by this algorithm. Relying on shadows in estimating the height of a building is advantageous by requiring minimum amount of information for such estimation. However, in dense areas, shadows of a building can be obstructed by neighboring buildings from one or more sides. To present our algorithm confidence in estimated heights, we present a belief measure ( $\beta$ ) that simply is a confidence score indicating how reliable the estimated measure of the height is. For every building,  $\beta$  is computed using the following algorithm:

1. Estimate the height of  $h$  for the building A using the proposed methodologies in Section II-B.
2. Extract the expected shadow regions of  $R_{shadow}$  for building A at the estimated height of  $h$  as described in Section II-B3 (regions surrounded by the red-solid lines in Figure 19).
3. Find all the immediate neighboring buildings of the building A.
4. Estimate the overlap between the expected shadow of building A and rooftop areas of all neighboring buildings ( $R_n$ ).
5. Compute the *belief measure* from:

$$\beta = 1 - \frac{\sum_{\forall R_n} Area(R_{shadow} \cap R_n)}{Area(R_{shadow})} \quad (23)$$



Fig. 19: Estimating the *belief measure*.

If a building is in complete isolation or far from neighboring buildings or obstacles, its shadow will not be disturbed and

therefore the estimated height can be potentially fully trusted. Since in this work the main focus is on building detection and height estimation, the belief measure is defined based on those detected buildings in close vicinity of the building of interest. This however does not mean that the presented results are always fully trustworthy. For instance, Figure 20 represents a case (building 5 of image 1 of Table VII) with a *belief measure* of one. As shown in this image, the estimated height is wrong due to two large trees with shadows that interfere with the building shadow. The undisturbed shadow of the building on the vertical side is very small and therefore is not sufficient to accurately estimate the correct height. In an interfered case, if the sun is not at nadir and the shadows of the building are not disturbed from at least one side, the estimated height will be correct (building 2 and 4 of image 17 of Table VII). Therefore *belief measure* is a good measure to be more aware of potential wrong estimates. A sub-system that can identify trees and account for their disturbance would be a necessary component for further improvement of this work.

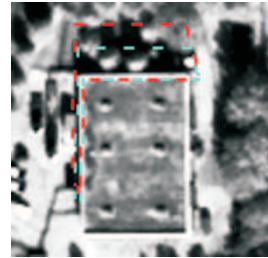


Fig. 20: Shadow disturbance by trees around a building. The *belief measure* is equal to one. The red dotted line represents the shadow due to the height estimated by the system. The blue dotted line represents the ground truth shadow.

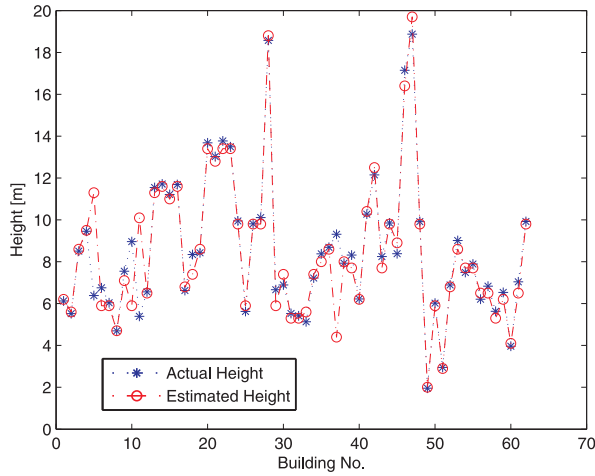
Table VII represents the estimated height values for 62 buildings in 20 satellite images. It also compares the height values with the manually found ground truth in each case.

From this table the mean error of all cases (without considering the *belief measure*) is 0.53 meter and the RMS error is 1.18 meters. Figure 21(a) displays the estimated heights versus the ground truth. Figure 21(b) shows the absolute error in each case.

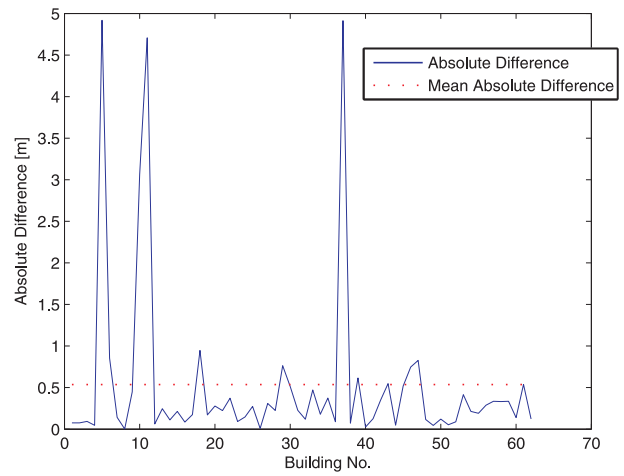
In these results, there are 9 (out of 62) cases in which the error between the estimated height and the ground truth is more than 0.6 meter. In all these cases the belief measure is equal to 1. Further investigations for these cases revealed that the error in each case is due to either overlap of the actual shadows with the non-building objects (such as trees) or their shadows in the vicinity, or poor quality of the existing shadows. Among 62 cases there are also 4 cases for which estimated heights are off by 3 meters or more.

TABLE VII: Comparison of the estimated heights with the manually prepared ground truth.

Img. No.	Bldg. No.	Estimated Height[m]	Actual Height[m]	Abs. Diff.[m]	Belief Measure	Img. No.	Bldg. No.	Estimated Height[m]	Actual Height[m]	Abs. Diff.[m]	Belief Measure
1	1	6.2	6.124	0.076	0.55	1	2	5.6	5.524	0.076	1
1	3	8.6	8.507	0.093	1	1	4	9.5	9.453	0.047	1
1	5	11.3	6.382	4.918	1	2	1	5.9	6.756	0.855	1
2	2	5.9	6.042	0.142	1	2	3	4.7	4.697	0.003	1
2	4	7.1	7.549	0.448	0.42	2	5	5.9	8.964	3.063	1
2	6	10.1	5.393	4.707	1	2	7	6.5	6.562	0.062	0.31
3	1	11.3	11.711	0.246	1	3	2	11.6	11.214	0.111	1
3	3	11.0	11.685	0.213	1	4	1	6.8	6.625	0.085	1
4	2	7.4	8.348	0.175	1	4	3	8.6	8.430	0.947	1
5	1	13.4	13.680	0.171	1	5	2	12.8	13.023	0.279	0.81
5	3	13.4	13.773	0.223	0.45	5	4	13.4	13.492	0.373	0.73
6	1	9.8	9.947	0.092	1	7	1	5.9	5.627	0.147	1
7	2	9.8	9.810	0.274	1	7	3	9.8	10.109	0.009	1
8	1	8.8	8.852	0.309	1	9	1	18.8	18.576	0.224	1
9	2	5.9	6.665	0.764	1	10	1	7.4	6.889	0.512	1
10	2	5.3	5.528	0.227	1	10	3	5.3	5.419	0.119	1
10	4	5.6	5.129	0.471	1	11	1	7.4	7.221	0.179	1
12	1	8.0	8.375	0.374	1	12	2	8.6	8.690	0.089	1
12	3	4.4	9.314	4.913	1	12	4	8.0	7.928	0.072	1
12	5	7.7	8.316	0.616	1	12	6	6.2	6.225	0.025	1
12	7	10.4	10.275	0.125	1	13	1	12.5	12.151	0.349	1
14	1	7.7	8.248	0.548	1	14	2	9.8	9.848	0.048	1
15	1	8.9	8.383	0.518	1	16	1	16.4	17.149	0.748	1
16	2	19.7	18.873	0.827	1	17	1	9.8	9.917	0.116	1
17	2	2.0	1.956	0.044	0.36	17	3	5.9	6.022	0.121	1
17	4	2.9	2.952	0.052	0.21	17	5	6.8	6.889	0.088	1
18	1	8.6	9.016	0.415	1	19	1	7.7	7.486	0.214	1
19	2	7.7	7.891	0.190	1	19	3	6.5	6.213	0.288	1
19	4	6.5	6.836	0.335	0.73	19	5	5.3	5.633	0.335	1
19	6	6.2	5.535	0.139	1	19	7	4.1	3.961	0.538	0.13
20	1	6.5	7.038	0.123	1	20	2	9.8	9.923	0.123	1



(a)



(b)

Fig. 21: Comparison of actual heights (ground truth) with the values estimated by the proposed system.



Figure 22 represents several examples of building profiles with their casted shadow footprints at the estimated height values. In these images, for each rooftop hypothesis the expected shadow footprint corresponding to the estimated height is overlaid on the ground by the red lines. The manually prepared ground truth shadows (corresponding to the true heights) are displayed with the blue lines.

Figure 23-right represents 3D models of buildings for a sample aerial scene image (shown in left) reconstructed by the proposed method.

In the literature, there are a few methods estimating buildings heights using shadow [8]–[10], [24]–[26], [41]. Some of these are semi-automatic and require user’s interaction [24], [41]. Others could automatically estimate height of buildings with simple rooftop shape such as rectilinear with the condition that their shadows are completely visible and identifiable [8]–[10], [25], [26]. To our best knowledge, there is no comparable shadow-based method similar to ours that is capable of estimating heights of buildings with complex rooftops. Therefore we only evaluated our estimating heights using manually prepared ground truth. The method by Nevatia [11], which has some similar aspects to our work, does not present any quantitative results regarding the accuracy of the estimated heights.

### C. Performance Issues

All presented algorithms are implemented in Matlab 7.4 on a PC with CPU Intel Core2 2.4GHz with 2GB RAM. For the rooftop detection, the system processing time is highly dependent on the dimensions of the input image, the amount of details, and maximum and minimum lengths of potential buildings in the image. The processing time ranges from several seconds up to several minutes (for larger images). Table VIII represents the typical time for images of specific sizes. In generating these results we have used images with similar building density.

TABLE VIII: Processing time versus image dimensions.

Image Size (pixels)	Time (minutes)
100 × 100	0.57
200 × 200	3.54
300 × 300	8.18
400 × 400	23.12
500 × 500	50.52
600 × 600	62.44
700 × 700	71.05
800 × 800	120.24

One of the main factors (beside the image size) affecting the processing time is the number of detected lines in each image. The higher this number is the higher the processing

time will be. This is due to the fact that higher number of lines contributes to a larger number of line intersections and potentially more hypotheses that need to be verified. A large image from an urban area includes many line segments and therefore requires a high processing time. However, same size image, if taken from a forest region, will be processed much faster since it does not include manmade edges and straight lines.

Generally smaller image cuts lead to much faster processing and more accurate results (due to less building candidates). Processing smaller images and stitching the results together could be a way for improving the systems processing time.

### D. System Portability Issues

The proposed system is adjusted for grayscale QuickBird satellite images. Like any complex system, there are a few parameters that are set according to the characteristics of the input imagery. In order to customize the system for other types of images (captured by different photometric sensors), some of the parameters in different sections of the proposed work must be adjusted accordingly. In Section II-A.2, parameters  $d_{max}$ ,  $d_{min}$  and  $w$  should be set according to the pixel size of the input images. Moreover, the rooftop candidate filtering related parameters in Section II-A.3 smoothness of intensity values inside each rooftop ( $\sigma_h$ ), and the ratio of intensity difference between inside and outside regions of each rooftop candidate ( $|mean_{in} - mean_{out}|/mean_{in}$ ) should be adjusted according to noise level of in the input imagery. These two parameters are adjusted later in Section II-A.4 (Hypothesis retrieval) by 10%. This percentage may also require changing for different type of input images. In the 3D Building Estimation, the optimal height estimation part of the system (Section II-B.5), the incremental height step should also be adjusted according to the resolution of the input image. For example, in this work (QuickBird images with resolution of 0.6 meter/pixel) the incremental height step is set to 0.3, i.e., half the pixel size. In Pictometry images where the pixel size is 15 cm/pixel this parameter should be set to 7.5 cm.

## IV. CONCLUSIONS

This paper presented a new system for detecting flat or flat-looking rooftops with polygonal profiles and estimating their heights using singular satellite/aerial images. In the first phase of this work, lines and their intersections (real or potential) are detected. The orientation of the lines at their intersections is used for creating a graph-based search algorithm in which subsequent polygonal vertices are tracked and identified. There is no angular limitation in detection of such rooftops. The general algorithm has no limit in the number of sides of the detected polygonal rooftops. However as this number



- - - - - : Shadow estimated by the proposed algorithm  
 - - - - - : Manually prepared ground truth

Fig. 22: Examples of the building 3D profiles of the input buildings (casted shadow footprints at the estimated height values). In these images, for each rooftop hypothesis the expected shadow footprint corresponding to the estimated height is overlaid on the ground by the red lines and the manually prepared ground truth projected shadows (corresponding to the true heights) are displayed with the blue lines.



Fig. 23: Visual representation of 3D models of buildings for a sample aerial scene image reconstructed by the proposed method.

increases the computational complexity of the algorithm also will increase. For that reason, in this work a limit of 8 was imposed on the number of sides of the detected polygonal rooftops.

In the second phase of this work, to estimate the height of each building, we proposed a new method which incorporates rooftop definitions, shadows, and satellite metadata. The profiles of buildings expected shadows, at various heights, are estimated and projected into the image using acquisition geometry provided by the satellite metadata. The best height is found by measuring the similarity between the expected shadows at various heights and the existing shadows on the ground using a fuzzy-based approach. Special attention is paid in creating accurate geometrical representation of the projected shadows on the ground so that the visible parts can be identified and separated from the hidden parts. System results are compared against the manually found ground truth and against results reported by previous work in the literature.

#### V. ACKNOWLEDGMENT

Authors would like to acknowledge with gratitude the NSERC Canada and MacDonald Dettwiler and Associates Ltd. for support through the NSERC Strategic Grant Program.

#### REFERENCES

- [1] H. Mayer, "Automatic object extraction from aerial imagery—a survey focusing on buildings," *Comput. Vis. Image Underst.*, vol. 74, no. 2, pp. 138–149, 1999.
- [2] J. Peng and Y. C. Liu, "Model and context-driven building extraction in dense urban aerial images," *International Journal of Remote Sensing*, vol. 26, pp. 1289–1307(19), April 2005.
- [3] S. Mayunga, Y. Zhang, and D. Coleman, "Semi-Automatic Building Extraction utilizing Quickbird Imagery," 2005, pp. 131–136.
- [4] H. Ruther, H. M. Martine, and E. G. Mtalo, "Application of snakes and dynamic programming optimisation technique in modeling of buildings in informal settlement areas," in *ISPRS J. Photogrammet. Remote Sensing*, 2002, pp. 269–282.
- [5] Z. Liu, J. Wang, and W. Liu, "Building extraction from high resolution imagery based on multi-scale object oriented classification and probabilistic Hough transform," vol. 56, no. 4, 2002, pp. 2250–2253.
- [6] T. Baillioleul, V. Prinet, B. Serra, and P. Marthon, "Spatio-temporal Prior Shape Constraint for Level Set Segmentation," in *Proc. of Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2005, pp. 503–519.
- [7] K. Karantzaos and N. Paragios, "Recognition-Driven 2D Competing Priors Towards Automatic And Accurate Building Detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 1, pp. 133–144, 2009.
- [8] R. Irvin and D. McKeown, "Methods for Exploiting the Relationship between Buildings and Their Shadows in Aerial Imagery," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1564–1575, November 1989.
- [9] C. Lin, A. Huertas, and R. Nevatia, "Detection of buildings using perceptual grouping and shadows," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 62–69, 1994.
- [10] C. Lin and R. Nevatia, "Building detection and description from a single intensity image," *Comput. Vis. Image Underst.*, vol. 72, no. 2, pp. 101–121, 1998.
- [11] S. Noronha and R. Nevatia, "Detection and Modeling of Buildings from Multiple Aerial Images," *IEEE Trans. PAMI*, vol. 23, no. 5, pp. 501–518, 2001.
- [12] M. Cord and D. Declercq, "Three-dimensional building detection and

- modeling using a statistical approach," *IEEE Trans. IP*, vol. 10, no. 5, pp. 715–723, May 2001.
- [13] Z. Kim and R. Nevatia, "Automatic description of complex buildings from multiple images," *Computer Vision and Image Understanding*, vol. 96, no. 1, pp. 60–95, Oct. 2004.
- [14] C. Baillard, C. Schmid, A. Zisserman, and A. Fitzgibbon, "Automatic Line Matching And 3D Reconstruction Of Buildings From Multiple Views," in *In ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, IAPRS Vol.32, Part 3-2W5*, 1999, pp. 69–80.
- [15] C. Baillard and A. Zisserman, "Automatic Reconstruction of Piecewise Planar Models from Multiple Views," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, p. 2559, 1999.
- [16] K. Fujii and T. Arikawa, "Urban object reconstruction using airborne laser elevation image and aerial image," *IEEE Trans. GRSS*, vol. 40, no. 10, pp. 2234–2240, Oct. 2002.
- [17] C. Jaynes, E. Riseman, and A. Hanson, "Recognition and Reconstruction of Buildings from Multiple Aerial Images," *Computer Vision and Image Understanding*, vol. 90, no. 1, pp. 68–98, Apr. 2003.
- [18] M. Xie, K. Fu, and Y. Wu, "Building Recognition and Reconstruction from Aerial Imagery and LiDAR Data," *International Conference on Radar CIE06*, pp. 1–4, 2006.
- [19] G. Sohn and I. Dowman, "Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction," *International Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 1, pp. 43–63, May 2007.
- [20] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot Deseilligny, "Automatic building extraction from DEMs using an object approach and application to the 3D-city modeling," *ISPRS J. Photogrammetry and Remote Sensing*, vol. 63, no. 3, pp. 365–381, May 2008.
- [21] J. Burns, A. Hanson, and E. Riseman, "Extracting straight lines," *Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 4, pp. 425–455, July 1986.
- [22] A. Katartzis, H. Sahli, E. Nyssen, and J. Cornelis, "Detection of buildings from a single airborne image using a markov random field model," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, ser. IGARSS 2001, vol. 6, 2001, pp. 442–446.
- [23] D.-M. Woo and D.-C. Park, "Rooftop detection based on 3d line data using fast graph search," in *Proceedings of the 2009 Ninth International Conference on Hybrid Intelligent Systems - Volume 03*, ser. HIS '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 442–446. [Online]. Available: <http://dx.doi.org/10.1109/HIS.2009.303>
- [24] X. Huang and L. K. Kwok, "3D building reconstruction and visualization for single high resolution satellite image," *IEEE International In Geoscience and Remote Sensing Symposium*, pp. 5009–5012, 2007.
- [25] T. Kim, T. Javzandulam, and T. Y. Lee, "Semiautomatic reconstruction of building height and footprints from single satellite images," *IEEE International In Geoscience and Remote Sensing Symposium*, pp. 4737–4740, 2007.
- [26] A. Laviigne, P. Saeedi, A. Dlugan, N. Goldstein, and H. Zwick, "Automatic building detection and 3D shape recovery from single monocular electro-optic imagery," *SPIE Defense and Security Symposium*, 2007.
- [27] C. Tison, F. Tupin, and H. Maitre, "Retrieval of building shapes from shadows in high resolution SAR interferometric images," *IEEE International In Geoscience and Remote Sensing Symposium*, vol. 3, pp. 1788–1791, 2004.
- [28] V. Tsai, "A Comparative Study on Shadow Compensation of Color Aerial Images in Invariant Color Models," *GeoRS*, vol. 44, no. 6, pp. 1661–1671, June 2006.
- [29] D. Comanicu and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Trans. PAMI*, vol. 24, no. 5, pp. 603–619, May 2002.
- [30] R. Cannon, J. Dave, and J. Bezdek, "Efficient Implementation of the Fuzzy C-Means Clustering Algorithm," *IEEE Trans. PAMI*, vol. 8, no. 2, pp. 248–255, March 1986.
- [31] E. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *Journal of Man-Machine Studies*, vol. 7, no. 2, pp. 1–13, 1975.
- [32] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions On Systems Man And Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.
- [33] P. Saeedi and H. Zwick, "Automatic Building Detection in Aerial and Satellite Images," *International Conference on Control, Automation, Robotics and Vision (ICARCV 2008)*, pp. 623–629, 2008.
- [34] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison: Wesley Publishing, 1989.
- [35] K. Deb, *Optimization for Engineering Design*. Prentice-Hall India, 1998.
- [36] C. R. Reeves and J. E. Rowe, *Genetic algorithms: principles and perspectives: a guide to GA theory*. Springer, 2002.
- [37] D. McKeown, T. Bulwinkle, S. Cochran, W. Harvey, C. McGlone, and J. Shufelt, "Performance evaluation for automatic feature extraction," *International Archives of Photogrammetry and Remote Sensing 33 (Part B2)*, p. 379394, 2000.
- [38] Y. Wei, Z. Zhao, and J. Song, "Urban building extraction from high-resolution satellite panchromatic image using clustering and edge detection," *IEEE International In Geoscience and Remote Sensing Symposium*, pp. 2008–2010, 2004.
- [39] X. Jin and C. Davis, "Automated Building Extraction from High-Resolution Satellite Imagery in Urban Areas Using Structural, Contextual, and Spectral Information," *EURASIP Journal on Applied Signal Processing*, vol. 2006, no. 1, pp. 2196–2206, 2005.
- [40] L. Wei and V. Prinet, "Building Detection from High-resolution Satellite Image using Probability Model," *IEEE International In Geoscience and Remote Sensing Symposium*, vol. 6, pp. 3888–3891, July 2005.
- [41] T. Lee and T. Kim, "Reconstruction of 3D building structures from IKONOS images through monoscopic line and shadow analysis," *Processing of Asian Conference on Remote Sensing*, pp. 7–11, 2005.



**Mohammad Izadi** (S'06) received the B.S. degree in Computer Engineering from the Isfahan University, Isfahan, Iran and the M.S. degree in Computer Engineering (Artificial Intelligence) from Amirkabir University of Technology, Tehran, Iran, in 2003 and 2006, respectively. He is currently pursuing the Ph.D. degree in Engineering Science (Intelligent Systems and Control) at Simon Fraser University, Canada. His research interests include computer vision, pattern recognition, neural networks, automatic 3D map generation, and object tracking.



**Parvaneh Saeedi** (M'04) received the B.A.Sc. degree in electrical engineering from the Iran University of Science and Technology, Tehran, Iran, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada, in 1998 and 2004, respectively. From 2004 to 2006, she was a Research Associate with MacDonald Detwiller and Associates Ltd. Since 2007, she has been an Assistant Professor with the School of Engineering Science, Simon Fraser University, BC, Canada. Her research interests include pattern recognition, machine vision, image understanding, and artificial intelligence.