Building Outline Extraction Using a Heuristic Approach Based on Generalization of Line Segments

Tahmineh Partovi, Reza Bahmanyar, Thomas Krauß, and Peter Reinartz, *Member, IEEE*

Abstract—Efficient and fully automatic building outline extraction and simplification methods are highly demanded for 3D model reconstruction tasks. In spite of the efforts put into developing such methods, the results of the recently proposed methods are still not satisfactory, especially for satellite images, due to object complexities and the presence of noise. Dealing with this problem, in this article, we propose a new approach which detects rough building boundaries (building mask) from *Digital Surface Model (DSM)* data, and then refines the resulting mask by classifying the geometrical features of the high spatial resolution panchromatic satellite image. The refined mask represents finer details of the building outlines which are close to the original building edges. These outlines are then simplified through a parameterization phase, where a tracing algorithm detects the building boundary points from the refined masks and a set of line segments is fitted to them. After that, for each building, the existing main orientations are determined based on the length and arc lengths of the building's line segments. Our method is able to determine the multiple main orientations of complex buildings. Through a regularization process, the line segments are then aligned and adjusted according to the building's main orientations. Finally, the adjusted line segments are intersected and connected to each other in order to form a polygon representing the building's outlines. Experimental results demonstrate that the computed building outlines are highly accurate and simple, even for large and complex buildings with inner yards.

Index Terms—Building detection, DSM, High resolution satellite image, Outline extraction, Outline simplification

I. INTRODUCTION

EXTRACTING buildings and reconstructing their full 3D models from remotely sensed images have been the focus of research by many scientists for over two decades [1], [2]. In many applications such as urban planning, network planning for mobile communication, tourism information systems, and disaster monitoring, 3D building models of urban areas play an essential role. For the latter they facilitate monitoring natural or man-made disaster, by providing a comprehensive view of structures and activities, which leads to better planning and providing quicker response [3].

Despite the existence of various methods for 3D building model reconstruction, developing efficient and fully auto-matic systems is a topic which is still approached by many researchers [2], [4]. Before reconstructing a building's 3D model, the building components such as outlines, should be extracted from the given images if the footprint is not available [5]. To this end, the building should be discriminated from other existing objects in the images such as vegetation, soil, and water surface. Following this, the parameters of the building components are computed.

A conventional approach for reconstructing a building's 3D model is to extract its geometrical components (e.g., lines, corners, planes) from *Digital Surface Model (DSM)* data. The extracted components are then grouped to form the 3D model of the building, considering their geometrical relationships [6]. Since buildings are elevated objects, the height information provided by DSM data helps distinguishing them from the other objects with a similar color or gray value (e.g. building with flat roofs from asphalt roads). DSM data is usually generated by applying stereo matching techniques to the acquired images from optical stereo satellite or aerial imaging equipments, or by using point clouds derived from LiDAR data. However, due to scene complexities (e.g., clutter, occlusion) and deficiencies in the techniques for generating DSM data such as the stereo matching methods, the resulting DSM data usually suffer from imperfections (e.g., gaps), especially at the building edges [6], [7]. For recovering gaps in DSM data, interpolation methods have been used by a number of previous works [8], [9]. Nevertheless, the obtained DSM data are still unsatisfactory due to the deficiencies of interpolation such as the decreasing sharpness of DSM data in building boundary areas [7]. To remedy this shortcoming, a number of previous works propose to integrate DSM data with additional sources of information, such as 2D cadaster maps and optical images [10], [11], [12].

In order to use the advantages of the DSM data while remedying its imperfections, in this article, we propose a new method which integrates DSM data with high spatial resolution panchromatic images to improve the building masks especially at the building boundaries. Using the resulting masks, we further develop a novel automatic method for optimized extraction and simplification of building outlines. Through a series of experiments, we show that our method can be generalized to various types of buildings, and that it is robust even for complex building structures.

T. Partovi[+], R. Bahmanyar[+,*], T. Krauß, and P. Reinartz are with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Wessling, Germany (e-mail: tahmineh.partovi@dlr.de; reza.bahmanyar@dlr.de; thomas.krauss@dlr.de; peter.reinartz@dlr.de).

[+]The corresponding authors.

[*]The author is also affiliated with the Munich Aerospace Faculty, Munich, Germany.

A. Related Works

Generally, the existing methods for building boundary extraction and 3D model reconstruction are categorized into model-driven and data-driven strategies [13].

1) Model-driven strategies: For reconstructing building roofs, model-driven strategies find optimum models of the roofs by fitting predefined parameterized roof models to given point clouds of buildings and minimizing a cost function [14], [15], [16].

Gerke *et al.* [5] introduced a hierarchical method for fitting rectilinear polygons to building boundaries. In this method, the building boundaries are detected and discriminated from surrounding objects such as trees, using DSM data of aerial imagery and *Normalized Difference Vegetation Index (NDVI)*. As a next step, the main orientation of each building is computed by using geometric moments. Then a rectangle is fitted to the building outline considering the building's main orientation. After that, the fitted rectangle is subtracted from the building mask, and a new rectangle is fitted to the remaining building area. The method iterates until all building parts become surrounded by rectangles. Although this method shows very good performance in the case of rectangular buildings, it fails for non-rectangular ones.

Dutter [17] proposed a similar model-driven method based on a *Minimum Bounding Rectangle (MBR)* approach. As opposed to the previous method [5], this method is able to fit polygons to the outlines of the buildings with T, L, and Z shape structures. Arefi *et al.* [18] proposed a method based on *Combined Minimum Bounding Rectangle (CMBR)* approach (which has been introduced mainly for regularization of non-rectilinear polygons) and *Random Sampling Consensus (RANSAC)* to extend the previously developed methods, in order to adapt them to more complex buildings such as the buildings with non-perpendicular edges.

Brédif *et al.* [19] developed a method for extracting building boundaries directly from DSM data without any building detection step. Rectangular building outlines are extracted from DSM data using a marked point process of rectangles. The optimum configuration of the rectangles is then found based on *Reversible Jump Markov Chain Monte Carlo (RJM-CMC)* coupled with a simulated annealing algorithm. This optimization process, minimizes an energy function which fits rectangle edges to height discontinuities of the DSM data and penalizes overlaps of different objects. As a next step, the rectangle edges are elongated to intersect each other and form line segments. Then a graph of the line segments is generated. Finally, using a graph cut technique, the line segments supporting the rectangle edges are arranged to create the final building outlines.

Using a novel active shape detection approach, Sirmacek *et al.* [7] proposed a method to extract building boundaries from DSM data of high and low resolution satellite images. Their proposed method detects building junctions and endpoints from building skeletons. The building is then divided into various pieces and a box-fitting algorithm is run on each piece, starting from the building piece's center point. The active rectangular shape grows until an energy function, considering the distance between the rectangle and a set of previously extracted canny edges, is minimized. This algorithm is able to approximate building shapes even if the building edges are not completed or closed; however, it is very slow and not very satisfactory for large and complex buildings.

2) Data-driven strategies: The data-driven strategies such as region growing [20], Hough transform [10], and RANSAC [21] extract roof planes and other geometrical information of buildings from their point clouds or dense meshes. The building roofs are reconstructed by assembling the extracted roof planes and determining the vertices, the ridge lines, and the eaves [4].

In order to delineate building boundaries, the data driven methods extract parameterized lines by fitting lines to building's boundary points using the Douglas-Peucker line simplification algorithm [22], graphical models, least squares, Hough transform, or RANSAC. All these methods require a regularization step followed by a least squares adjustment [23].

Shan and Sampath [24] proposed a method which detects buildings and extracts their boundaries from LiDAR point clouds. This method separates planar from nonplanar points by analyzing the eigenvalues of each roof point of a building within its Voronoi neighborhood. Then it clusters all the planar points using a fuzzy k-means method. After that, a convex hull algorithm is applied to the clustered points in order to trace the points lying on the building boundaries. Due to the irregular shapes of building boundaries and the presence of artifacts, the traced building boundaries need to be regularized before being used for determining the parameters of the building outlines. To this end, the authors consider the orientation of the longest boundary line as the main orientation of a building. Using a threshold, all the other boundary lines are then classified as being parallel or perpendicular to the building's main orientation. After that, the boundary lines are aligned according to their class using a least squares adjustment algorithm. The proposed method by Shan and Sampath [24] fails in the case of non-rectangular buildings (i.e., buildings with non-perpendicular edges) due to considering only rectilinear objects.

Sester and Neidhart [23] proposed a new method for building boundary extraction which can be generalized to non-rectangular buildings since it is operating independent of the buildings' main orientations. This method classifies the elevated points of a given point cloud and group them into a number of blobs corresponding to the existing buildings. The building outlines are then extracted from the obtained jagged blobs as a set of straight lines using RANSAC (which can robustly extract lines from noisy data [25]). As a next step, a least squares method is used to adjust and combine the straight lines to form meaningful building outlines, where the parallelism and rectangularity constraints of the consecutive lines are imposed to the least squares adjustment. Inspired by this work, Guercke and Sester [26] later proposed a new method which employs Hough analysis instead of RANSAC algorithm for extracting line segments. The generated line segments are then refined and adjusted to the building's original shape, using a least squares adjustment process.

Sohn *et al.* [4] developed a method based on BSP-tree optimization. The developed method detects building regions by computing NDVI from IKONOS images and segmenting
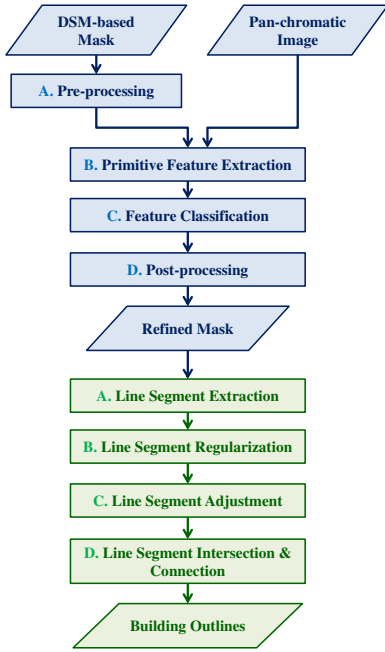
Fig. 1. Overview of our proposed building outline extraction method.

Image	Image size (pixels)	GSD	Roof Types
I_1	745×470	$50\ cm$	Flat
I_2	400×530	$50\ cm$	Flat
I_3	350×390	$50\ cm$	Flat
I_4	400×950	$50\ cm$	Pitched
I_5	800×450	$50\ cm$	Pitched and Flat
I_6	500×650	$50\ cm$	Pitched
I_7	400×590	$50\ cm$	Pitched
I_8	950×1300	$50\ cm$	Pitched
I_9	1350×750	$50\ cm$	Pitched and Flat
I_{10}	820×1070	$50\ cm$	Pitched and Flat

the corresponding LiDAR point clouds. For building boundary extraction, the authors introduced a method based on a combination of data-driven and model-driven strategies. This method extracts straight lines by the Burns algorithm [27] from IKONOS images. The lines shorter than a threshold are discarded and two rectangular boxes with predefined dimensions are considered around each line. The main orientations of the buildings are determined according to the obtained outlines. Finally, a BSP-tree algorithm is applied in order to receive the boundaries of the 3D models.

B. Methodology Overview

In this article, we introduce an improved framework, in comparison to the existing approaches, focusing on refinement of DSM-based building masks and simplification of the extracted building outlines. It concentrates on DSM derived from satellite data, where the DSM quality is low. Fig. 1 shows the main steps of our proposed method. In the mask refinement step (the blue blocks in Fig. 1), we enhance building boundaries by applying a classification method (e.g., *Support Vector Machine (SVM)* [28]) to the primitive geometrical features of their corresponding high resolution panchromatic images. In our experiments, we use *Scale Invariant Feature Transform (SIFT)* algorithm [29] to extract image primitive features. Experimental results demonstrate that this enhancement leverages the precision of the extracted building outlines.

Furthermore, we propose a new data-driven procedure for parameterized building outline extraction and simplification. Our method is able to delineate building outlines in complex scenarios, where the buildings have multiple non-rectilinear main orientations. As illustrated by the green blocks in Fig. 1, in order to extract a building's outline, we trace the building boundary points on its corresponding refined mask and a set

of line segments is fitted to them. The obtained line segments are then regularized by finding the building's main orientations, and assigning all the line segments to their appropriate main orientations, where they should be either parallel or perpendicular to their assigned main orientations. Since this classification is performed globally for the building, despite the close orientation distance of the line segments within a class, they might be spatially far from each other. Therefore, we impose a locality constraint to the class members and regroup the non-local members with the line segments in their neighborhood. As a next step, we propose a novel approach based on least squares adjustment to align the line segments. This approach considers the multiple orientations of each building which yields a more accurate delineation of building outlines. As final step, the aligned line segments are intersected and connected to each other resulting in the building's final outline. Experimental results demonstrate that the extracted building outlines are very close to the buildings' original edges even for complex buildings (e.g., buildings with inner yards and multiple non-rectilinear main orientations).

The article is organized as follows: Section II explains the process conducted for refining building masks. Section III proceeds by detailing the procedure followed for extracting building outlines including, line segment extraction, line segment regularization, and line segment intersection and connection. Section IV presents and discusses the resulting building outlines using our proposed method. Section V presents the conclusions.

II. BUILDING MASK REFINEMENT

In this section, we explain our proposed method for building mask refinement on orthorectified panchromatic images. To this end, we apply an SVM [28] classification method to the SIFT [29] description of the image features. In our experiments, we use the panchromatic images of DigitalGlobe's WorldView-2 for 10 sample regions of Munich. Fig. 2 shows these images, their specifications are mentioned in Table I. For evaluating different parts of our proposed method, they are compared to manually generated reference data which is exemplified in Fig. 3. (a).

In order to create DSM-based masks, the method introduced by Krauss *et al.* [30] is used. This method discriminates
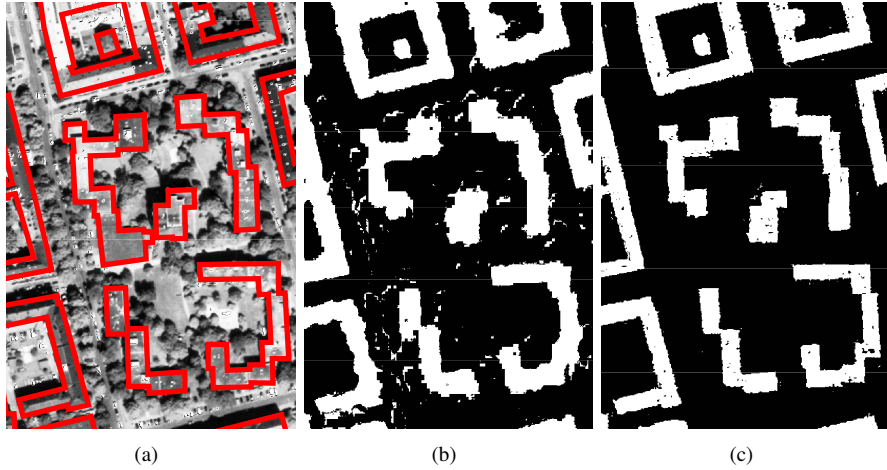
Fig. 3. (a) Manually generated reference data. (b) DSM-based building mask. (c) Building mask after refinement.
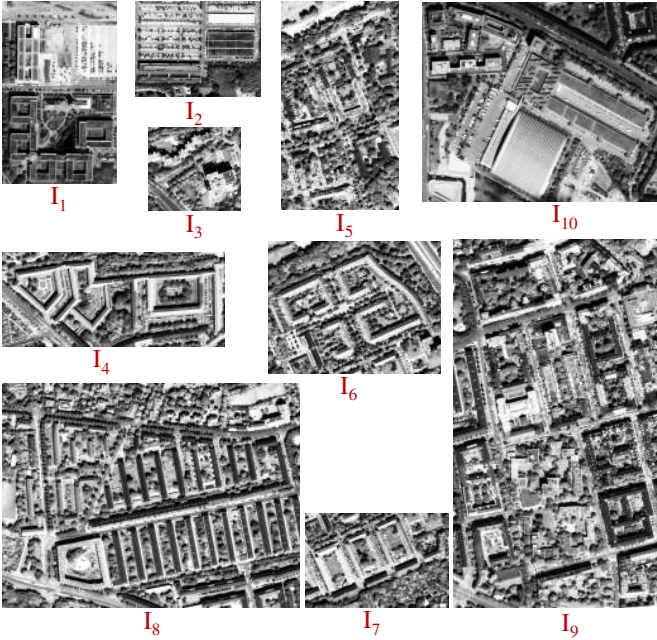

Fig. 2. Orthorectified panchromatic images used in our experiments.

the ground level objects from the higher level ones using *normalized Digital Surface Model (nDSM)* [31]. The buildings are then extracted from the surrounding patterns and clutter by using *Advanced Rule-based Fuzzy Spectral Classification* followed by height thresholding [30]. Fig. 3. (b) shows unrefined DSM-based building mask. As illustrated in the figure, a pure DSM-based mask usually represent rough building boundaries; however, some patterns and objects such as shadows and trees are incorrectly considered as building elements due to their similar characteristics or heights.

A. Mask Preprocessing

As the first step, considering the minimum possible building size in the given images, we remove the groups of points, indicated as buildings by the given DSM-based masks, which are smaller than a threshold. In our experiments, the threshold

is set to 300 pixels because at the 0.5 m resolution of our panchromatic images, we assume only buildings that cover areas larger than 75 m^2. Then we apply a morphological opening followed by a closing to the mask with the structuring element for the opening larger than that of the closing's. We use a disc with a diameter of 3 pixels for opening and a disc with a diameter of 2 pixels for closing. This makes the mask large enough to cover the probable missing building edges as well as some surrounding areas, which will be used later as negative samples for the classification phase.

B. Primitive Feature Extraction

In order to classify image regions into building and non-building, we describe each pixel within the masked area of panchromatic images by a vector of its important features using the SIFT method [29]. SIFT extracts local geometry-based features of an image such as edges and corners. In our experiments, each SIFT feature vector represents 128 dimensions, computed for a window of 16×16 pixels around each particular pixel.

C. Feature Classification

In each image, to discriminate the area covered by buildings from the rest of the image, we randomly sample 30% of the building and non-building candidate pixels, determined by the preprocessed mask and manually generated reference data, as positive and negative samples, respectively. Then we train the SVM on the sample points. In our experiments, the SVM parameters are selected empirically by varying the parameters and validating the resulting model on a small set of the sample points. Then the parameters leading to the best model are considered for further experiments.

Since we selected the images (i.e., I_1–I_{10}) to contain diverse building structures and environments (please refer to Table I), sampling the training data from different images allows us to study the effects of various roof types (e.g., flat, pitched) and their surroundings on the resulting SVM models. For example, in order to study model biases caused by the degree of feature variations in the training phase, one can train an SVM model
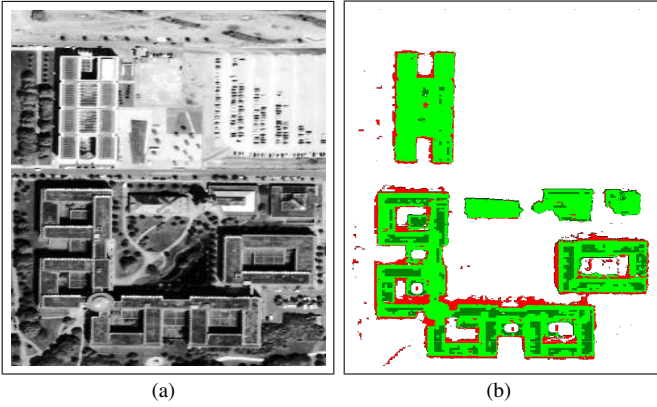
Fig. 4. (a) Original panchromatic image and (b) its refined mask, bright and dark green areas, together with the removed areas from the given mask, red areas.

Fig. 5. Line segment extraction process.

on I_1 and I_2 (which contain features only from flat roof buildings) and test the resulting model on I_4, I_6 and I_7 (which contain the features only from pitched roof buildings).

SIFT descriptors are highly sensitive to building edges compared to the other building features (e.g., patterns on rooftops). Since SIFT is computed for a window of size 16x16 pixels, the pixels within 8-pixel distance (i.e., half the SIFT window size) of building edges still contain edge information. Thus, usually these pixels are classified into the positive class forming building silhouettes with precisely determined contours. However, the pixels within building areas which are too far from building edges to contain building edge information, are usually classified into the negative class. This results in gaps within building boundaries which will be filled in the post-processing step. Fig. 4, shows a sample classification result in which the positive class is depicted in bright green and the negative class in dark green and red, where dark green refer to the gaps within building boundaries.

D. Mask Post-processing

After discriminating the regions covered by buildings, we perform a morphological closing (with a disc of 3 pixels diameter) followed by a gap filling method[1] (with a threshold of 400 pixels) in order to fill in the small imperfections and gaps within the building boundaries (dark green areas in Fig. 4). The resulting refined masks will be further used in the building outline extraction process. Fig. 3. (c) and Fig. 4 demonstrate building masks after refinement. Although there are still small gaps in the building areas, since usually they do not lie on building edges, they wont influence the results of the building outline extraction.

III. BUILDING OUTLINE EXTRACTION AND SIMPLIFICATION

In order to extract parameterized building outlines, in this section, we model the refined building masks as polygons. Experimental results show that the refined masks approximate the true position of the building boundaries (please refer to
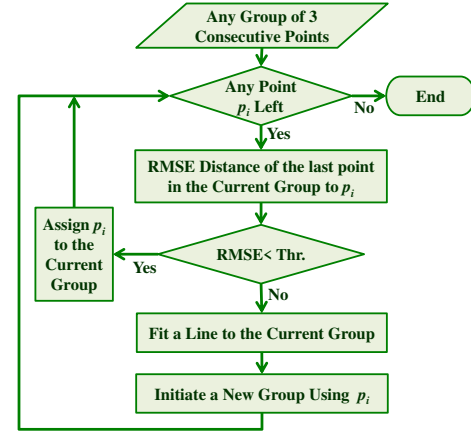
[1]We employed the gap-filling function of MATLAB.

Section IV-A). Therefore, tracing the boundary points of a building according to the refined mask, allows us to detect the building's outline. To this end, we use the *modified Moore-Neighbor* tracing algorithm [32] which detects boundary points in a consecutive manner. After that, a consecutive line fitting procedure based on the combination of RANSAC and least squares adjustment is applied to each group of the boundary points in order to extract line segments.

A. Line Segment Extraction

A complete overview of the process chain followed for this step is depicted in Fig. 5, which starts with an initial group of three consecutive boundary points. Then *Root Mean Square Error (RMSE)* distance of a next consecutive point is measured from the last point of the initial group. If the RMSE is less than a predefined threshold, the point will be grouped with the previous point. Otherwise, it will be used for initiating a new group, and to all the points within the previous group, a line is fitted locally. The line is centered at the mean of the points, and extends in the directions of the points' principal eigenvector, with the scale determined by the corresponding eigenvalue to the principal eigenvector. This process iterates until all the boundary points are represented by line segments. The line segments shorter than 5 pixels are then removed because at the 0.5 m resolution of our panchromatic images a building length cannot be smaller than 5 pixels. Due to the irregular shape of the structures formed by the remaining line segments caused by the existing noise and artifacts, the line segments need to be regularized. Fig. 6. (a) illustrates the extracted line segments for an example building.

B. Line Segment Regularization

To refine the extracted line segments, we determine the main orientation of each building. A conventional assumption is that buildings have mutually perpendicular orientations. However, in reality, complex buildings usually have more than one main orientation which might be non-perpendicular. In this article, we propose a method that considers multiple main orientations for each building, regardless of the perpendicularity of the orientations.
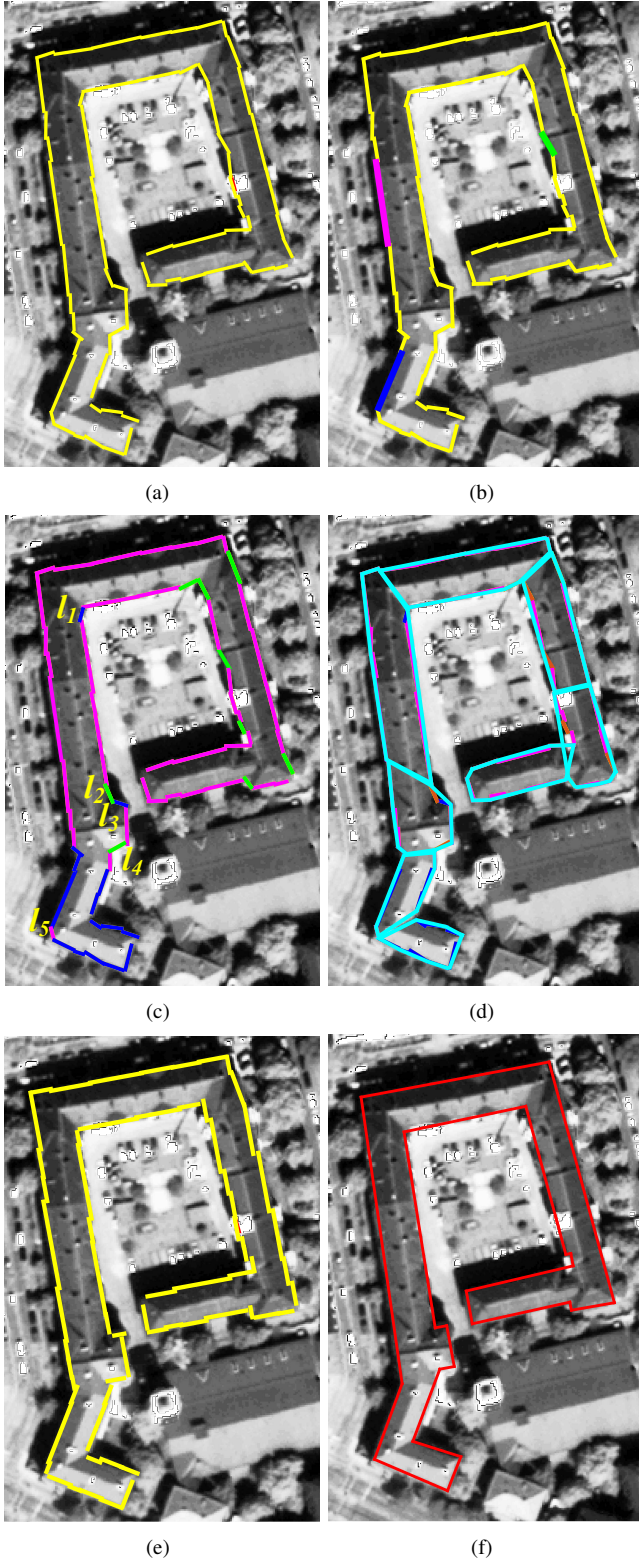
(a)   (b)

(c)   (d)

(e)   (f)

Fig. 6. (a) Extracted line segments for an example building. (b) Three main orientations of the building depicted by magenta, green, and blue colors. (c) Assignment of the line segments to the building's main orientations. (d) Splitting the building into several convex hulls. (e) Extracted line segments after regularization and adjustment. (f) Outline of the building after intersecting and connecting the line segments.
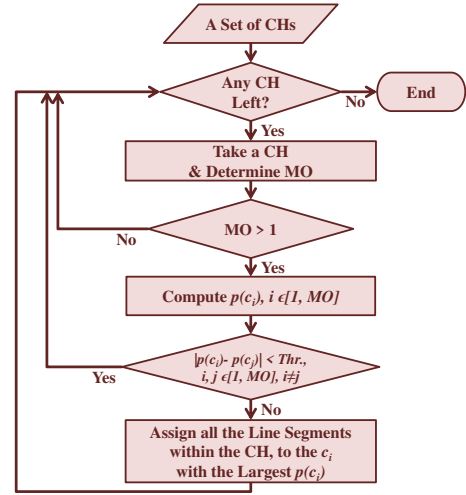

Fig. 7. Process chain of the line segment regularization, where CH denotes a convex hull and MO indicates the number of main orientation classes c_i in each convex hull.

We determine the buildings' main orientations based on the length of the line segments and their arc lengths. For each building, we select the orientation of its longest line segment as the building's main orientation. The other line segments which are either parallel or perpendicular to this main orientation will be grouped together. To check the parallelism and perpendicularity, we measure the arc lengths of the longest line segment and the other line segments, based on a predefined threshold (5 pixels in our experiments). With this threshold the angle distance would be smaller than $20°$ which is a good assumption for parallelism. In the next iteration, the longest of the remaining line segments is selected to determine another main orientation of the building. This process iterates until all the line segments are grouped with a main orientation of the building. Fig. 6. (b) shows three main orientations of the building depicted by magenta, green, and blue colors. The assignment of the other line segments to these three main orientations is demonstrated in Fig. 6. (c).

In addition to the orientations of the line segments within each orientation class, their locality is also important. In other words, the line segments within each class should also lay in the same neighborhood. However, as illustrated in Fig. 6. (c), there are a number of line segments misclassified due to the presence of noise and artifacts. For example, in Fig. 6. (c), the line segments l_1, l_2, l_3, and l_4 should be assigned to the magenta orientation class, whereas l_5 should be assigned to the blue orientation class. Dealing with this problem, we verify the assigned main orientation to the line segments in each neighborhood and assign the miss-classified line segments to a correct orientation class. Fig. 7 represents the main steps of this procedure.

We start by splitting each building into several segments (convex hulls) by extracting the points on the skeletons of the building masks. The points are then placed in 5 pixels distance from each other and for each point, 10 nearest neighboring line segments are selected and a convex hull is fitted (please refer to Fig. 6. (d)), where the number of nearest neighbors are

determined empirically. In each convex hull, the orientations of the line segments are compared and the probability of every existing main orientation class c_i is computed as:

$$p(c_i) = \frac{\sum_j l_{ij}}{\sum_i \sum_j l_{ij}}, \tag{1}$$

where l_{ij} indicates the length of the line segment j which belongs to the orientation class c_i. If the probabilities of the existing main orientation classes in a convex hull are close enough, no change will be imposed. Otherwise, all the line segments will be grouped into the main orientation class with the largest probability. As a next step, a least squares adjustment algorithm is performed to adjust the line segments according to their orientation classes.

C. Line Segment Adjustment

In this step, we adjust the line segments within each orientation class in order to align them as either parallel or perpendicular to the main orientation of the class, based on their arc lengths to the main orientation. Therefore, the line segment adjustment problem can be seen as fitting a rectangle to the points of the line segments. In our method, we solve the least squares adjustment based on the *Gauss-Helmert (GH)* model [33], [34]. In the GH model, the functional model for the estimated values is:

$$F(\hat{L}, \hat{X}) = 0, \tag{2}$$

where $\hat{L}$ and $\hat{X}$ are the estimated observed and unknown parameters, respectively. The functional model for the line fitting can be expressed as the following target functions,

$$F_1 = ax + by + c, \tag{3}$$

$$F_2 = a^2 + b^2 - 1, \tag{4}$$

where F_1 is a line equation with the coefficients a, b, and c. Since a and b are the normal vector components of each line segment, F_2 is a constraint equation to normalize the normal vectors of the line segments. F_2 is a non-linear function; therefore, in order to be used in the least squares adjustment procedure, we linearize it using Tailor's series expansion. The GH's functional model can therefore be written as:

$$F(\hat{L}, \hat{X}) = w + Bv + A\hat{x}, \tag{5}$$

where A and B are matrices containing the derivatives of the target functions (F_1 and F_2) with respect to the unknown parameters (a, b, and c) and the observed parameters (x and y), respectively. Furthermore, w is the vector of misclosure. Fig. 8 illustrates an example of fitting a model to a set of points using least squares adjustment. In this figure, n_1 and n_2 indicate the normal vectors of the parallel and perpendicular lines, respectively, while c_j are the line segment intercepts.
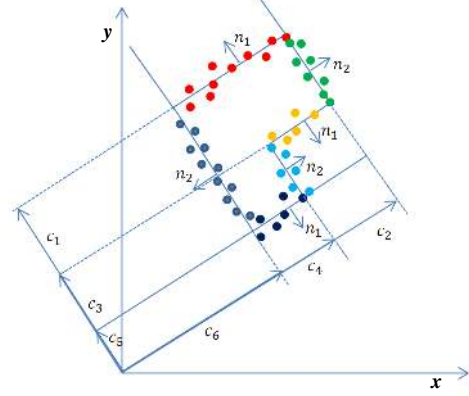

Fig. 8. Fitting a model to a set of points using least squares adjustment.

Assuming only one main orientation, the matrices A and B, and the vector w are as follows,

$$A = \frac{\partial F(L, X^0)}{\partial X^0} =$$

$$\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & x_{K_1(l_1)} & y_{K_1(l_1)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & x_{K_{N_1}(l_1)} & y_{K_{N_1}(l_1)} \\
0 & 1 & 0 & 0 & 0 & 0 & y_{K_1(l_2)} & -x_{K_1(l_2)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 1 & 0 & 0 & 0 & 0 & y_{K_{N_2}(l_2)} & -x_{K_{N_2}(l_2)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 1 & y_{K_1(l_6)} & -x_{K_1(l_6)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & 0 & 0 & 1 & y_{K_{N_6}(l_6)} & -x_{K_{N_6}(l_6)} \\
0 & 0 & 0 & 0 & 0 & 0 & (2a)^0 & (2b)^0
\end{pmatrix}, \tag{6}$$

$$B = \frac{\partial F(L, X^0)}{\partial L} = \begin{pmatrix}
a^0 & b^0 & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & a^0 & b^0 \\
0 & 0 & 0 & \cdots & 0 & 0 & 0
\end{pmatrix}, \tag{7}$$

$$w = F(L, X^0) = \begin{pmatrix}
a^0 * x_{K_1(l_1)} + b^0 * y_{K_1(l_1)} + c_{l_1}^0 \\
\vdots \\
-b^0 * x_{K_T(l_6)} + a^0 * y_{K_T(l_6)} + c_{l_6}^0 \\
(a^0)^2 + (b^0)^2 - 1
\end{pmatrix}, \tag{8}$$

where X^0 indicates the initial unknown parameters which are considered as a^0, b^0 and c^0, and computed based on *Singular Value Decomposition (SVD)*. Furthermore, L is the observation vector which represents the point coordinates of the line segments, in our experiments. Moreover, $x_{K_z(l_j)}$ and $y_{K_z(l_j)}$ are the x and y coordinates of the z-th point of the line segment l_j (i.e., $K_z(l_j)$, $z \in [1, N_j]$).

In Equation (5), v indicates the residuals of the observations and $\hat{x}$ is the difference between the estimated values of the unknown vector and its initial values. These two vectors are computed as:

$$v = \hat{L} - L, \tag{9}$$

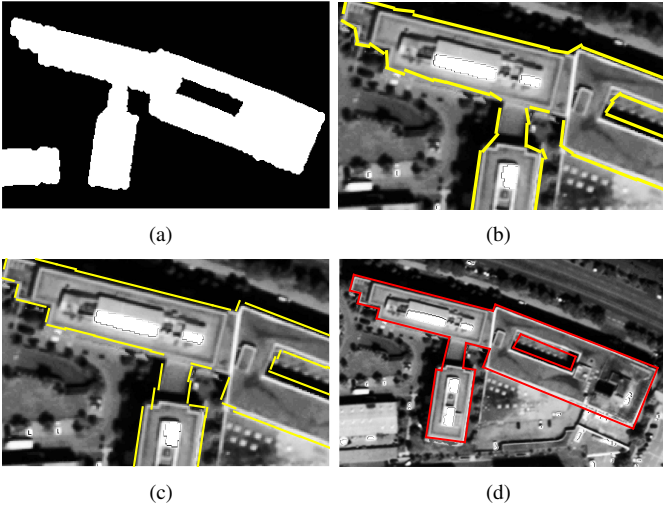$$\hat{x} = \hat{X} - X^0. \tag{10}$$

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

Fig. 9. (a) Refined mask of an example building. (b) Extracted line segments from the refined mask. (c) Extracted line segments after regularization and adjustment. (d) Outline of the building after intersecting and connecting the line segments.

In order to solve the least squares adjustment problem, we express the unknown parameters by:

$$\begin{pmatrix} k \\ \hat{x} \end{pmatrix} = - \begin{pmatrix} BQ^{-1}B^T & A \\ A^T & 0 \end{pmatrix}^{-1} \cdot \begin{pmatrix} w \\ 0 \end{pmatrix} \qquad (11)$$

and the residuals are calculated by:

$$v = Q^{-1}B^T k, \qquad (12)$$

where Q is a matrix determining the line segment's weight. Since we empirically found out that using various weightings do not influence the resulting building outlines, in our experiments, all the line segments are equally treated, and therefore, Q is an identity matrix.

In Equation 6, the last two columns correspond to one main orientation. In case of having more than one main orientation, two extra columns are added to the matrix A for each additional main orientation, which contains the coordinates of the additional main orientation. Moreover, one additional row is added to the matrix A as a normalization constraint. Furthermore, for each additional main orientation, one zero row is inserted to the end of the matrix B and a normalization constraint is inserted to the end of vector w.

Fig. 9. (a), (b), and (c) demonstrate the refined mask of a building, the extracted line segments from the refined mask, and the line segments after regularization and adjustment. As illustrated in the figure, the adjusted line segments are aligned to the building edges; however, they are not properly connected to each other. Fig. 6. (e) shows another example of regularized and adjusted line segments for a building.

D. Line Segment Intersection and Connection

In the last step, we intersect and connect each line segment to its closest neighbor according to the following rules:

1) Any two parallel line segments with perpendicular distance larger than 5 pixels are joined by inserting a connection line perpendicular to both line segments (please
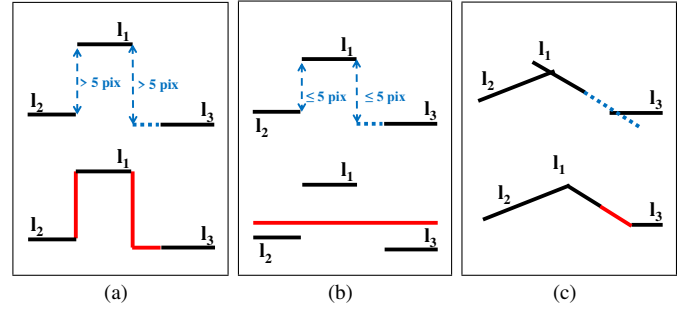

(a) (b) (c)

Fig. 10. Each sub-figure shows a condition and its corresponding constraint for connecting the line segments; as well as the result of line segment connection. (a) Join parallel lines with perpendicular distance larger than 5 pixels. (b) Join parallel lines with perpendicular distance smaller than or equal to 5 pixels with the assumption, $l_2 > l_1, l_3$. (c) Join intersecting lines.

refer to Fig. 10. (a)). If the line segments' endpoints are not aligned (e.g., l_1 and l_3 in Fig. 10. (a)), one of the line segments is selected randomly and its length is adjusted (e.g., either elongated or shortened) to make its endpoint aligned to the endpoint of the other line segment. The threshold of 5 pixels is selected according to the the resolution of the given panchromatic image and the DSM data (about 0.5 m). In this resolution, 5 pixels represent 2.5 m which for urban buildings is negligible within the generalization tolerance.

2) For the parallel line segments with perpendicular distance smaller than or equal to 5 pixels, we insert a line with the minimum distance from the other line segments and extend it to cover the extent of all the line segments (please refer to Fig. 10. (b)).

3) In the case of intersecting line segments, we adjust their length until their neighboring end points meet (please refer to Fig. 10. (c)).

IV. RESULTS AND DISCUSSION

In this section, we evaluate the precision of the extracted building outlines using our proposed method, both qualitatively and quantitatively. To this end, we analyze 10 regions of a panchromatic image of Munich, acquired by the WorldView-2 satellite (please refer to Fig. 2 and Table I). The regions are selected so that they contain a variety of building structures with different degrees of complexity (e.g., multiple main orientations, buildings with inner yards, different roof types). To extract a building's outline, we perform all the steps developed in Section II and III.

Our proposed method relies on some parameters in its various steps. Table II shows parameters present in each step and the values assigned to them in our experiments. As described in the corresponding sections, most of these parameters are dependent on the resolution of the panchromatic image being used. Therefore, for new images with the same resolution they can be used as they are and for new images with different resolution, they could be adapted accordingly.

A. Mask Refinement

In order to conduct a qualitative evaluation of the mask refinement process, Fig. 11 exemplifies the refinement applied

TABLE II
PARAMETERS USED IN OUR APPROACH

| Steps	Parameters	Values
Mask preprocessing (Section II-A)	Threshold for removing small masked areas	300 pixels
	Morphological opening disc size	3 pixels
	Morphological closing disc size	2 pixels
Feature Classification (Section II-C)	SVM parameters	Empirically selected
Mask Post-processing (Section II-D)	Morphological closing disc size	3 pixels
	Threshold for gap filling	400 pixels
Line Segment Extraction (Section III-A)	Short line extraction threshold (RMSE)	1 pixels
	Length threshold for deleting small line segments	5 pixels
Line Segment Regularization (Section III-B)	Arc length distance for main orientation computation	5 pixels
	Point distance of skeleton for building partitioning	5 points
	Number of nearest neighboring line segments to a selected point on skeleton	10 lines
Line Seg. Intersec. and Connec. (Section III-D)	Distance between line segments	5 pixels
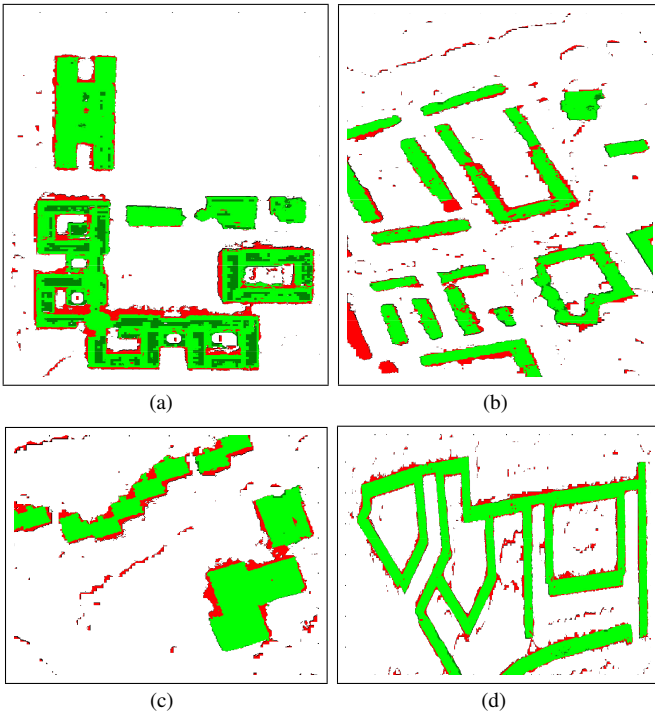
Fig. 11. Refined masks (bright and dark green areas) together with the removed areas from the given masks (red areas). (a) I_1, (b) I_5, (c) I_2, (d) I_4.

to the given masks for I_1, I_3, I_4, and I_5. In this figure, the resulting refined building area is depicted in green (both light and dark greens), where the dark green depicts the gaps in the given masks which have been filled during the refinement process. The regions depicted by red have been removed by the refinement process from the given masks. According to the results, building edges in the refined masks are finer and more precise than in the given masks.

In order to evaluate the refined building masks quantitatively, we compare them pixel by pixel to the reference data and compute the *Quality* measure,

$$Quality = \frac{TP}{TP + FP + FN}. \tag{13}$$

This measure provides an overall quality degree of the masks by considering building detection and boundary delineation. In Equation (13), TP ($True\ Positive$) and FP ($False\ Positive$) are the number of pixels which are correctly and incorrectly considered as building by the mask, respectively. In addition, FN ($False\ Negative$) indicates the number of building pixels which are missed by the mask.

Fig. 12 shows the Quality of the given and the refined masks. In this figure, each column shows an experiment in which the training data is sampled from particular images of our dataset. In order to cross-validate the classification, for each experiment, the SVM is trained and tested three times using three different pixel samplings (where the SVM parameters are set empirically). For each experiment, we test all the three SVM models on every images of our dataset, resulting in three sets of refined masks. All the three sets of masks are then evaluated and the average Quality value is considered for each image.

In Fig. 12. (a), the training data is sampled from images I_1 and I_2 which contain only flat roof buildings. As the results show, the refined masks outperform only for the images containing flat roof buildings (e.g., I_1, I_2, I_3), whereas in the case of pitched roofs the refined masks are even worse than the given masks. To demonstrate that adding more similar knowledge does not help in compensating the SVM model's bias, in Fig. 12 (b), we train the SVM model on samples from I_3 (which also contains only flat roof buildings) in addition to I_1 and I_2. The results indicate that there is practically no change in the classification performance (i.e., mask quality) comparing to the previous experiment. Fig. 12. (c) illustrates the model bias when it is trained on pitched roof buildings. As the figure illustrates, while the refined masks outperform the original masks for the pitched roofs (e.g., I_4, I_6, I_7, I_8, I_9), they are worse for the flat roofs (e.g., I_1, I_2, I_3).

Considering the results of these experiments, we conclude that SVM models should be trained by various building roof types and surroundings in order to avoid model biases. Fig. 12. (d) shows the quality of the refined masks using an SVM model trained on both flat and pitched roof buildings. The results indicate that for most of the images, the quality
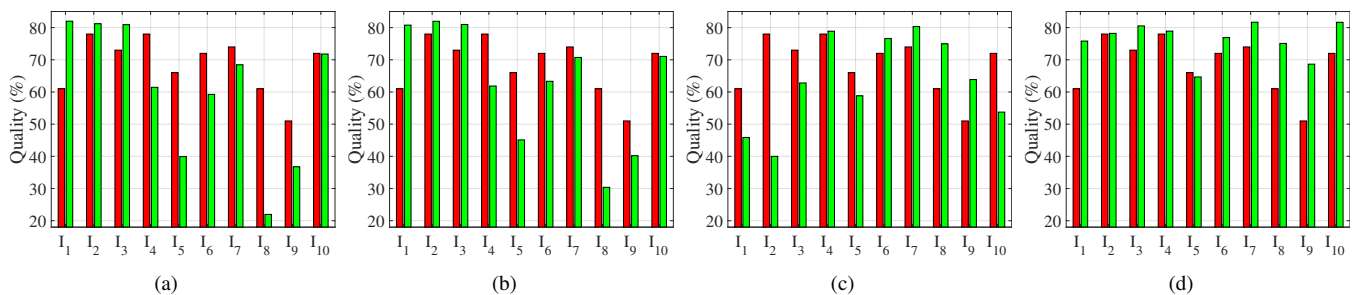
Fig. 12. The Quality measure of the given and refined masks of the images $I_1 - I_{10}$ using SVM classification. For training, 30% of the pixels within the building boundaries, indicated by the masks, is sampled from images: (a) I_1 and I_2, (b) I_1, I_2, and I_3, (c) I_4, I_6, and I_8, (d) I_1, I_3, I_4, and I_8.

of the refined mask is significantly higher than that of the given mask. Therefore, we will further use the refined masks resulting in this experiment for the building outline extraction task.

B. Building Outline Extraction

In this section, using the refined masks, we extract outlines of the buildings through the line segment extraction, regularization, adjustment, intersection, and connection steps.

1) Qualitative Evaluation: Fig. 13 and 14 show the final extracted building outlines depicted on our 10 sampled panchromatic images. As demonstrated by the results, despite the complexity of the buildings' structures such as the existence of inner yards, the extracted outlines are significantly close to the buildings' original edges. For delineating edges of building with inner yards, we extract the line segments of the buildings' outer boundaries and those of its inner yards separately. Then we consider all the extracted line segments together in order to determine the buildings' main orientations and regularizing the line segments. Since the edges of inner yards are usually aligned with the buildings' outer edges, considering them together increases the robustness of the resulting building outlines.

Besides its superior results, there are cases where our approach fails correctly extracting building outlines. Fig. 15 exemplifies these case. In Fig. 15. (a), the building edges e_1 and e_2 were not correctly delineated, probably due to the imperfection of the computed building skeleton caused by the selected threshold for the morphological process in the regularization step (see Section III-B). This affects the building decomposition (see Section III-B) in which those edges were grouped into the main orientation class of the e_3 and e_4. In Fig. 15. (b) and (c), the extracted building outlines were affected by the incompleteness of the building masks caused by occlusion (e.g., trees), shadow, height threshold used for mask generation, etc.

2) Comparison to an Existing Line Simplification Method: In order to extract building outlines, line simplification methods such as Douglas-Peucker [22] have been conventionally used. However, due to their shortcoming in delineating building fine details such as sequential corners, in this article, we propose a line segment extraction method as described in Section III-A. Fig. 16 compares the building outlines derived by our proposed method and by the Douglas-Peucker method,
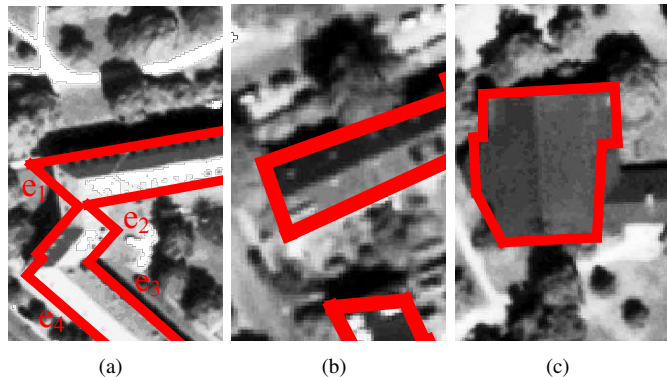

Fig. 15. Failure case examples of the proposed approach.

with the building footprints. The results for both methods are presented after regularization as explained in Section III-B. As the figure shows the building outlines obtained based on our proposed method are usually closer to the footprint as compared to the results based on the Douglas-Peucker method.

In Fig. 16. (a), due to the small size of the building, which limits the number of the available line segments, together with its multiple main orientations, the building outlines computed based on Douglas-Peucker cannot meet the buildings' original boundaries. However, since our proposed method relies on every single line segment, it results in a more accurate building outline. In Fig. 16. (b) the results of both methods are quite similar and pretty close to the building boundaries. However, a part of the building (the bottom part) remains undetected by both methods. This is due to the incompleteness (caused by occlusion, shadow, etc.) of the building mask. Fig. 16. (c) shows that Douglas-Peucker method performs poorly in delineating building corners composed by intersection of sequences of small line segments, whereas our proposed method is able to extract these details precisely.

3) Quantitative Evaluation: For a quantitative evaluation of the extracted building outlines, we compare them to the reference data using a newly introduced metric, namely *Polygons and Line Segment (PoLiS)* [35]. The PoLiS metric has been proposed for measuring the similarity of any two polygons [35]. It is a positive-definite and symmetric function which satisfies the triangle inequality. The PoLiS distance changes linearly with respect to small translation, rotation, and scale changes between the two polygons. In our experiments, in order to measure the PoLiS distance between a
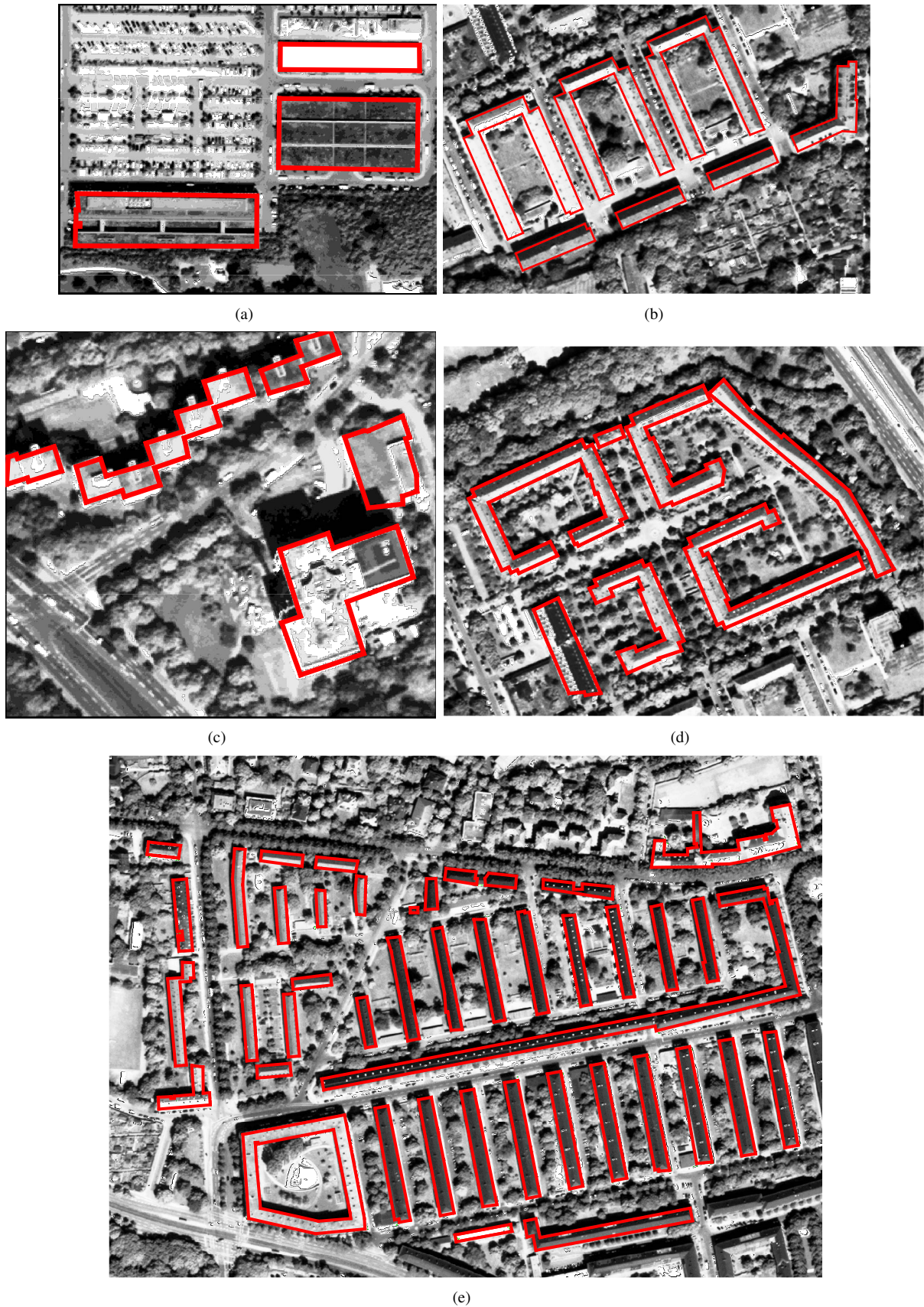
(a)

(b)

(c)

(d)

(e)

Fig. 13. Extracted outlines of the buildings within the images (a) I_2, (b) I_7, (c) I_3, (d) I_6, and (e) I_8, using our proposed approach.
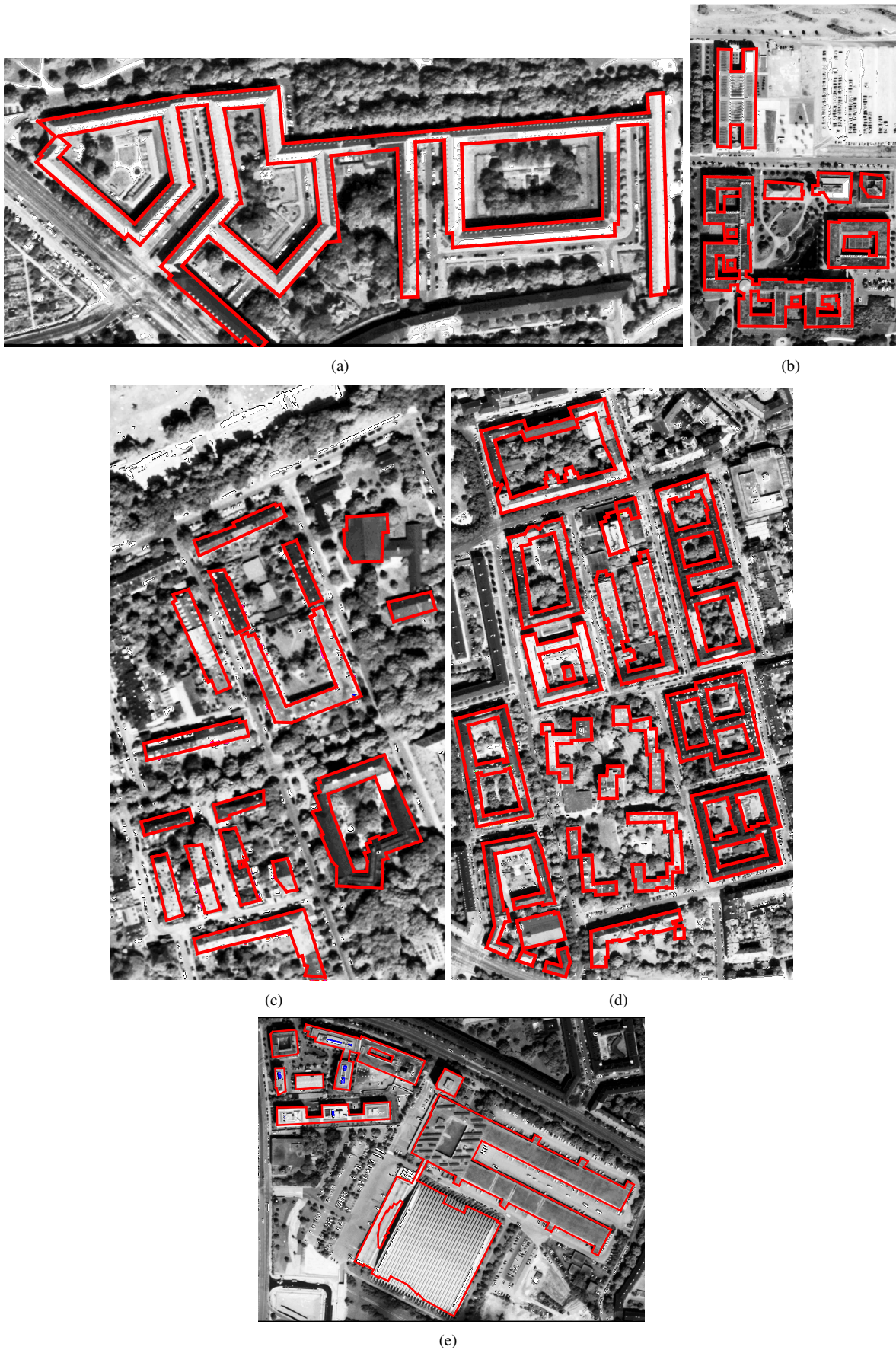
Fig. 14. Extracted outlines of the buildings within the images (a) I_4, (b) I_1, (c) I_5, (d) I_9, and (e) I_{10}, using our proposed method.
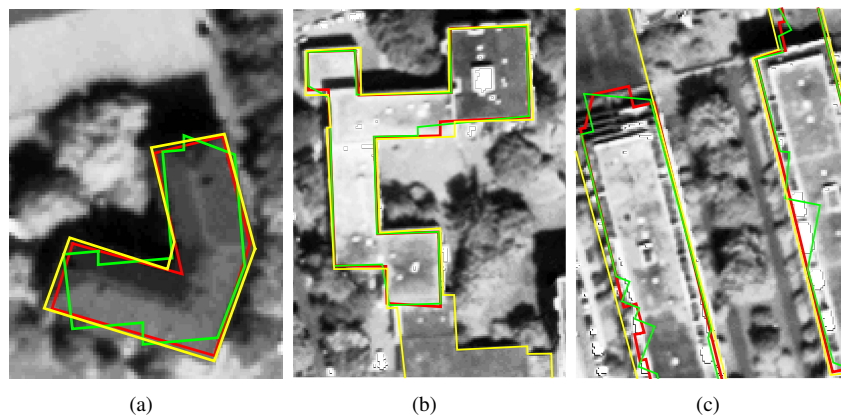
Fig. 16. Comparison of the building outlines based on our proposed line segment extraction method (red) and the Douglas Peucker algorithm (green), after regularization step. The yellow lines depict the building footprints.
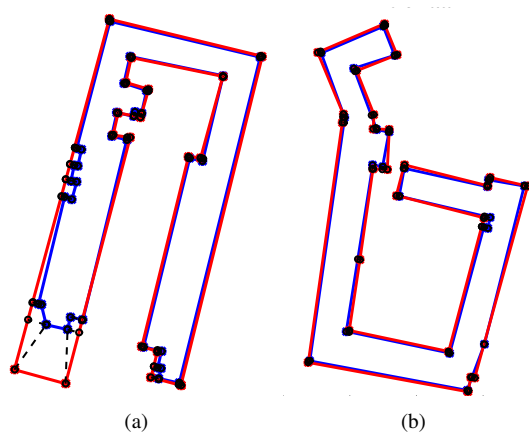

(a)   (b)

Fig. 17. Illustrations of PoLiS distance of the extracted (blue) building polygons to the reference (red) ones. (a) PoLis = 5.19, (b) PoLiS = 1.60.
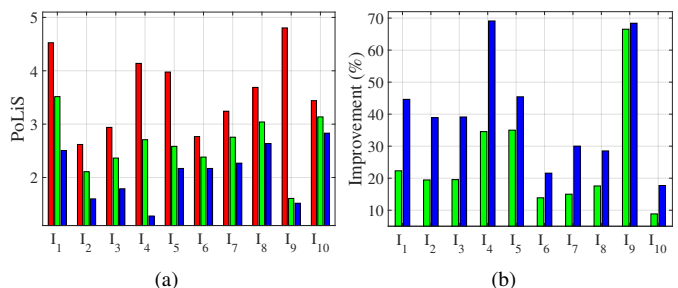

Fig. 18. (a) PoLiS distance between the extracted and the reference building outlines when, only the given DSM-based masks (the red bars), refined masks (the green bars), and regularization and line segment adjustment (the blue bars) are used. (b) Degree of improvement to the building outlines extracted using the given DSM-based masks when refinement step (the green bars), and regularization and line segment adjustment steps (the blue bars) are employed.

building's extracted outline (polygon) and its corresponding reference polygon, we compute the average distance between the extracted polygon vertices, and their closest vertices on the reference polygon. After that, the average distance between the reference polygon's vertices and their closest vertices on the extracted polygon is computed. The PoLiS distance between the two polygons is computed as the summation of the two average distances. Fig. 17 exemplifies PoLiS distance measurement.

The average PoLiS distances of the extracted building outlines in our 10 images are displayed in Fig. 18. (a). In this figure, the red bars indicate the PoLiS distance when only the given masks are used, the green bars represent the PoLiS distance after mask refinement, and the blue bars demonstrate the PoLiS distance when the refined masks are used coupled with line segment regularization, adjustment, intersection, and connection steps. As demonstrated by the results, using the mask refinement step reduces the PoLiS distance between the extracted outlines and the reference data to a large degree. Moreover, applying the regularization, adjustment, intersection, and connection steps further reduces the PoLiS distances significantly.

Fig. 18. (b) indicates up to which degree the mask refine-

ment (the green bars), and the line segment regularization, adjustment, intersection, and connection steps (the blue bars) improve the building outlines obtained by using only the given masks. According to the results, our proposed additional modules can increase the precision of the extracted building outlines up to 70% for some images (e.g., I_4 and I_9). For I_4, refining the building mask improved the building outlines about 35%, while after line segment regularization, adjustment, intersection, and connection steps, the results' precision increased up to about 70%. This shows that the imperfections of the building outlines are caused by both the building masks' faults and the complexities of the building structures. However, for I_9 the improvement is already almost achieved by refining the building mask, which indicates that in I_9, the building complexities are less problematic than in I_4.

Table III represents the PoLiS distance of the building outlines obtained based on our line segment extraction method and the Douglas-Peucker method, for the buildings shown in Fig. 16. Results indicate that the higher accuracy of the outlines extracted are based on the proposed method.

V. CONCLUSION AND FUTURE WORK

This article proposes a new approach for extracting and simplifying building outlines. It detects buildings and generates their mask using DSM data. The resulting masks are

TABLE III
COMPARING THE PoLiS DISTANCE OF THE EXTRACTED BUILDING
OUTLINES USING OUR METHOD AND DOUGLAS-PEUCKER

Image	Our method	Douglas-Peucker
Fig. 16. (a)	1.34	3.17
Fig. 16. (b)	9.11	9.24
Fig. 16. (c)	5.19	5.54

further refined based on high spatial resolution panchromatic images in order to represent the building edges closer to the original ones. After that, the refined masks are used in a new method composed of line segment extraction, regularization, adjustment, intersection, and connection steps, to extract building outlines as polygons. Experimental results show that the extracted building outlines are close to the buildings' original edges to a high degree. Moreover, our method can be generalized to various types of buildings and it is robust against complexities of building structures such as the existence of inner yards and multiple main orientations. In our further work, we will fit curves to the building round corners and the extracted building outlines will be used for full 3D model reconstruction.

ACKNOWLEDGMENT

REFERENCES

[1] F. Rottensteiner, G. Sohn, J. Jung, M. Gerke, C. Baillard, S. Benitez, and U. Breitkopf, "ISPRS benchmark on urban object classification and 3d building reconstruction," in *ISPRS Annals of Photogrammetry and Remote Sensing*, vol. I-3, 2012, pp. 293–298.

[2] N. Haala and M. Kada, "An update on automatic 3D building reconstruction," *ISPRS Journal of Photogrammetry, Remote Sensing*, vol. 56, pp. 570–580, 2010.

[3] C. Poullis and S. You, "Photo-realistic large-scale urban city model reconstruction," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 4, pp. 654–669, 2009.

[4] G. Sohn, X. Huang, and V. Tao, "A data-driven method for modelling 3D building objects using a binary space partitioning tree," in *Topographic Laser Ranging and Scanning: Principles and Processing*, J. Shan and C. K. . Toth, Eds. CRC Press, 2008, pp. 479–50.

[5] M. Gerke, C. Heipke, and B. M. Straub, "Building extraction from aerial imagery using a generic scene model and invariant geometric moments," in *Proc. IEEE/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, 2001, pp. 85–89.

[6] T. Partovi, R. Bahmanyar, T. Krauß, and P. Reinartz, "Building roof component extraction from panchromatic satellite images using a clustering-based method," *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-3, pp. 247–252, 2014.

[7] B. Sirmacek, H. Taubenbock, P. Reinartz, and M. Ehlers, "Performance evaluation for 3-D city model generation of six different DSMs from air- and spaceborne sensors," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 1, pp. 59–70, 2012.

[8] P. d'Angelo, M. Lehner, T. Krauss, D. Hoja, and P. Reinartz, "Towards automated DEM generation from high resolution stereo satellite images," *International Society for Photogrammetry and Remote Sensing*, vol. 37, pp. 1137–1342, 2008.

[9] D. Hoja, P. Reinartz, and M. Lehner, "DSM generation from high resolution satellite imagery using additional information contained in existing DSM," *ISPRS International Archives of the Photogrammetry and Remote Sensing and Spatial Information Sciences*, vol. 36, 2005.

[10] G. Vosselman and S. Dijkman, "3D building model reconstruction from point clouds and ground plans," *International Archive of Photogrammetry and Remote Sensing*, vol. 34, pp. 37–43, 2001.

[11] N. Haala, S. Becker, and M. Kada, "Cell decomposition for building model generation at different scales," in *Proc. Urban Remote Sensing Joint Event*, 2007, pp. 1–6.

[12] T. Partovi, H. Arefi, T. Krauß, and P. Reinartz, "Automatic model selection for 3D reconstruction of buildings from satellite imagary," *ISPRS Archives*, vol. XL-1/W, pp. 315–320, 2013.

[13] H.-G. Maas and G. Vosselman, "Two algorithms for extracting building models from raw laser altimetry data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 54, no. 23, pp. 153 – 163, 1999.

[14] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot-Deseilligny, "Structural approach for building reconstruction from a single dsm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 135–147, 2010.

[15] H. Huang, C. Brenner, and M. Sester, "A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 79, pp. 29 – 43, 2013.

[16] T. Partovi, H. Huang, T. Krauss, H. Mayer, and P. Reinartz, "Statistical building roof reconstruction from WorldView-2 stereo imagery," *ISPRS International Archives of the Photogrammetry and Remote Sensing and Spatial Information Sciences*, vol. XL-3/W2, pp. 161–167, 2015.

[17] M. Dutter, "Generalization of building footprints derived from high resolution remote sensing data," Diplomarbeit, Vienna University of Technology, 2007.

[18] H. Arefi, J. Engels, M. Hahn, and H. Mayer, "Approximation of building boundaries," in *Proc. Workshop on Urban Data Management Systems (UDMS)*, 2007, pp. 25–33.

[19] M. Brédif, O. Tournaire, B. Vallet, and N. Champion, "Extracting polygonal building footprints from digital surface models: A fully-automatic global optimization framework," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 77, pp. 57 – 65, 2013.

[20] F. Rottensteiner, J. Trinder, S. Clode, and K. Kubik, "Automated delineation of roof planes in LiDAR data," *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, pp. 221–226, 2008.

[21] L. T. Tarsha-Kurdi F. and P. Grussenmeyer, "Extended RANSAC algorithm for automatic detection of building roof planes from LiDAR data," *The Photogrammetric Journal of Finland*, vol. 21, no. 1, pp. 97–109, 2008.

[22] D. H. Douglas and T. K. Peucker, *Algorithms for the reduction of the number of points required to represent a digitized line or its caricature*. John Wiley & Sons, Ltd, 2011, pp. 15–28.

[23] M. Sester and H. Neidhart, "Reconstruction of building ground plans from laser scanner data," in *Proc. International Conference on Geographic Information Science (AGILE)*, 2008, pp. 1–11.

[24] J. Shan and A. Sampath, "Urban terrain and building extraction from airborne LiDAR data," in *Urban Remote Sensing*, Q. Weng and D. A. . Quattrochi, Eds. CRC Press, 2006, pp. 21 – 47.

[25] R. Raguram and J.-M. Frahm, "RECON: Scale-adaptive robust estimation via residual consensus," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 1299–1306.

[26] R. Guercke and M. Sester, "Building footprint simplification based on hough transform and least squares adjustment," in *Proc. Workshop of the ICA Commission on Generalization and Multiple Representation*, 2011, pp. 1–9.

[27] J. B. Burns, A. R. Hanson, and E. M. Riseman, "Extracting straight lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 4, pp. 425–455, 1986.

[28] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[29] D. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE International Conference on Computer Vision*, vol. 2, 1999, pp. 1150–1157.

[30] T. Krauss, B. Sirmacek, H. Arefi, and P. Reinartz, "Fusing stereo and multispectral data from WorldView-2 for urban modeling," in *Proc. SPIE Defense, Security and Sensing*, vol. 8390, 2012, pp. 1–15.

[31] U. Weidner and W. Foerstner, "Towards automatic building extraction from high -resolution digital elevation models." *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 50, no. 4, pp. 38–49, 1995.

[32] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB*, 2nd ed. Gatesmark Publishing, 2009.

[33] W. Niemeier, *Ausgleichungsrechnung: statistische Auswertemethoden*, ser. De Gruyter Lehrbuch, 2008.

[34] J. Avbelj, "Fusion of hyperspectral images and digital surface models for urban object extraction," Ph.D. dissertation, Technical University of Munich, 2015.

[35] J. Avbelj, R. Muller, and R. Bamler, "A metric for polygon comparison and building extraction evaluation," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 1, pp. 170–174, 2015.

Tahmineh Partovi received her B.S. degree in Geomatics Engineering at the college of national geographic organization, Tehran, Iran in 2007 and M.Sc. degree in Photogrammetry at University of Tehran, Tehran, Iran in 2011. Currently, she is Ph.D. student at the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, Germany. In autumn 2015, she was a guest scientist at the International Institute for Geo-Information Science and Earth Observation (ITC) in Enschede, Netherlands. Her research interests include 3-D building reconstruction for urban area, building boundaries extraction and classification using very high resolution satellite imagery and Digital Surface Model (DSM).

Reza Bahmanyar completed his B.Sc. degree in Electrical Engineering at University of Mazandaran (Babol, Iran) in 2009. He received his M.Sc. degree in Computer Science at Saarland University (Saarbruecken, Germany) in 2012. He wrote his master thesis in Computer Vision and Multimodal Computing department of Max-Plank Institute for Computer Science (Saarbruecken, Germany). Since 2012, he has enrolled at the Human-Machine Communication department of Technical University of Munich (Munich, Germany) as a doctoral student. He also holds a doctoral student position at the Remote Sensing Institute (IMF) of the German Aerospace Center (DLR; in Wessling, Germany). His main research interests include Computer Vision, Image Processing, Data Mining, and Artificial Intelligence.

Thomas Krauss received his Diploma (Dipl.-Phys.) in applied physics in 1993 from the Technical University of Munich. During the last 23 years he gained professional experience in the private and public sector. Currently he is team leader of the team 3D and Matching of the department Photogrammetry and Image Analysis at the German Aerospace Centre (DLR), Remote Sensing Technology Institute (IMF). He has more than 12 years of experience in image processing and remote sensing and participated in many national, international, EU and ESA projects in these fields. His main interests are in stereo-photogrammetry, generation of digital elevation models, interpretation and analysis of VHR data from sensors like WorldView, GeoEye, and Pleiades and data fusion of airborne, spaceborne and SAR imagery for higher georeferencing accuracy. Beneath these he also developed the cloud based processing chain CATENA for fully automatic processing of earth observation data at the DLR.

Peter Reinartz (M'09) received his Diploma (Dipl.-Phys.) in theoretical physics in 1983 from the University of Munich and his PhD (Dr.-Ing) in civil engineering from the University of Hannover, in 1989. His dissertation is on optimization of classification methods for multispectral image data. Currently he is department head of the department Photogrammetry and Image Analysis at the German Aerospace Centre (DLR), Remote Sensing Technology Institute (IMF) and holds a professorship for geomatics at the University of Osnabrueck. He has more than 25 years of experience in image processing and remote sensing and over 300 publications in these fields. His main interests are in direct georeferencing, stereo-photogrammetry and data fusion of space borne and airborne data, generation of digital elevation models and interpretation of VHR data from sensors like WorldView, GeoEye, and Pleiades. He is also engaged in using remote sensing data for disaster management and using high frequency time series of airborne image data for real time image processing and their operational use in case of disasters as well as for traffic monitoring.