Building Segmentation and Classification from Aerial LiDAR via Local Planar Features

Jun Chu[1,2(✉)], Wei Han[1,2,3], Wei Sui[3], Lingfeng Wang[3], Qi Wen[4], and Chunhong Pan[3]

[1] Institute of Computer Vision, Nanchang Hangkong University, Nanchang, China
chuj@nchu.edu.cn
[2] Key Laborator of Jiangxi Province for Image Processing and Pattern Recongnition, Nanchang, China
[3] NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[4] National Disaster Reduction Center of China, Ministry of Civil Affairs, Beijing, China

Abstract. In this paper, we propose a framework on building segmentation and classification from Aerial Lidar data via planar features. In this framework, the planar points corresponding to planar objects are obtained first by an unsupervised Markov random field clustering model. The ground normal is detected from planar points via the proposed constrained K-means algorithm. Within constrained K-means algorithm, the building points are generated by removing ground points from planar points. Furthermore, the candidate buildings are obtained by using region growing algorithm. Finally, these candidate buildings are classified into two types, that is, abnormal building and normal building based on the proposed vertical feature. Experimental results on a real world dataset demonstrate the effectiveness of our framework.

Keywords: Building segmentation · Planar objects · Aerial lidar data · Ground detection

1 Introduction

Earthquake and flood have taken place frequently over the world and brought disasters to natives. Many buildings are collapsed and damaged in the affected areas. In practice, it is difficult to measure and evaluate the damaged condition of buildings by manpower. Many algorithms based on computer vision have been proposed, among which image-based approaches are widely used. However, image-based approaches may not be applied to real-world problems effectively because the image acquisition is susceptible to lighting conditions. By contrast, 3D point cloud is robust to light. Hence, we propose a method to detect and classify buildings from 3D point cloud.

J. Chu—This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61263046, 61403376, 61175025 and 41301485).

H. Zha et al. (Eds.): CCCV 2015, Part II, CCIS 547, pp. 313–322, 2015.
DOI: 10.1007/978-3-662-48570-5_31

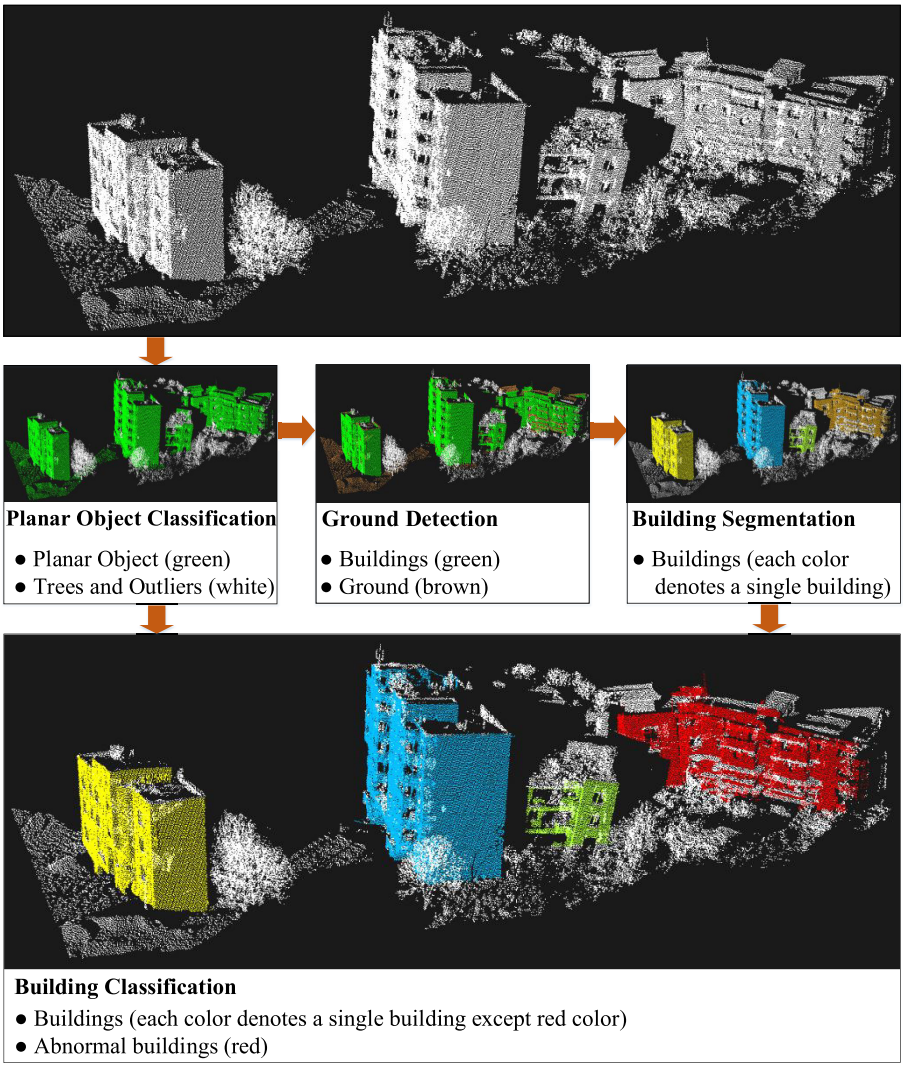

Fig. 1. A framework of building detection and classification.

1.1 Related Work

Building detection and classification approaches can be mainly divided into two categories, i.e., supervised learning based methods [1–4], and unsupervised learning based methods [5–9].

For the supervised learning based approaches, the features for points classification are learned by fitting a mixture of Gaussian model by Charaniya et al. [1] and Lalonde et al. [2] [3]. Secord et al. [4] proposed a method based on support vector machines for object detection using aerial lidar and image data.

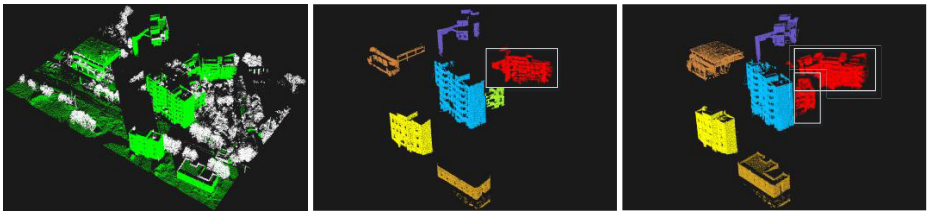

Fig. 2. One segmentation and classification result on **data1**. In sub-figures (b) and (c), each color represents a normal building except the red color. The red color represents abnormal buildings. Specifically, for red color buildings, each white box is a single abnormal building.

This method can provide satisfactory result. Unfortunately, the training sets are hard to be labeled due to the unstructured distribution of 3D point cloud.

The unsupervised learning approaches directly use the scatter or elevation for object including building detection and classification from 3D point cloud. Generally, for feature extraction, they focused on the neighborhood size of each point. For example, Weinmann et al. [7] minimized a energy function to obtain the optimal neighborhood size with the unknown density of 3D point cloud. The main limitation of this method is its computational complexity and fixing-density assumption. For the application of building detection, Carlberg et al. [5] and Lafarge et al. [6] used 3D point cloud to detect buildings in urban scenes. Specifically, Carlberg et al. [5] used a multi-category classifier to classify water, ground, roof, and trees. The height information has been used to remove ground and water under the assumption that the ground and water are in low height. Buildings and trees are detected through 3D shape analysis and region growing. Lafarge et al. [6] combined the features of local non-planarity, elevation, scatter and regular grouping to classify 3D point cloud data into buildings, vegetation, ground and clutters. The constructive solid geometry is then used to reconstruct buildings. The region growing segmentation and gradient orientation segmentation algorithms are used for classification of building, ground and vegetation in [8]. Matei et al. [9] proposed a building segmentation method by using error back propagation algorithm. These methods perform well on urban scenes, however, they may not be robust for rural scenes, where houses are built with different elevations.

1.2 Our Method

The framework of our method is illustrated in Fig. 1. The planar objects are first detected from 3D point cloud by a Markov random field (MRF) based model, which is motivated by [6] . Then, the ground normal is extracted by constrained K-means to further remove the ground points. After that, region growing method is used to obtain a single building and remove some of the clusters. Finally, the vertical feature is utilized for building classification.

The main contributions are highlighted as follows:

1. We propose an unsupervised building detection and classification framework based on planar features. Experimental results show that our method can provide satisfactory results.

2. A new dataset, which includes major difficulties in building detection and classification, is created to evaluated performance of our method. For quantitative evaluation, we have labeled the ground truth including all normal and abnormal buildings.

2 Local Planer Feature

In this section, we introduce the features used in our method. Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ be the set of input noise-free 3D points[1] in the scene, where N is the number of points. For each point $\mathbf{x}_i$, we first consider a subset $\mathcal{X}_i = \{\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \cdots, \mathbf{x}_{i,K}\} \subset \mathcal{X}$ as its K-nearest neighbors (K-NN). Then, the covariance matrix $\mathbf{C}_i \in \mathbb{R}^{3\times3}$ of these K-NN points $\mathcal{X}_i$ is calculated by

$$\mathbf{C}_i = \sum_{k=1}^{K} (\mathbf{x}_{i,k} - \mu_i)(\mathbf{x}_{i,k} - \mu_i)^{\mathsf{T}} , \tag{1}$$

where $\mu_i = \frac{1}{K}\sum_{k=1}^{K} \mathbf{x}_{i,k}$ is the mean of the K-NN points. After that, we can obtain the eigenvalues $\{\lambda_{i,j}\}_{j=1}^{3}$ of the covariance matrix $\mathbf{C}_i$ and the corresponding eigenvectors $\{\mathbf{u}_{i,j}\}_{j=1}^{3}$ by performing singular value decomposition, namely,

$$\mathbf{C}_i = \sum_{j=1}^{3} \lambda_{i,j} \mathbf{u}_{i,j} \mathbf{u}_{i,j}^{\mathsf{T}} . \tag{2}$$

Without loss of generality, we assume that $\lambda_{i,1} \geq \lambda_{i,2} \geq \lambda_{i,3}$. The local planar features of the point $\mathbf{x}_i$ are composed of two parts, that is,

$$\{f_i, \mathbf{n}_i\} = \left\{ \frac{\lambda_{i,2}}{\lambda_{i,3} + \epsilon}, \mathbf{u}_{i,3} \right\},$$

where $\epsilon = 10^{-5}$ is a small positive value to prevent dividing by zero. On the one hand, f_i represents the degree of planarization of the K-NN points $\mathcal{X}_i$ and the larger f_i is, the closer the plane distribution is. On the other hand, $\mathbf{n}_i$ is the normal of this potential plane.

3 The Proposed Framework

3.1 MRF-Based Planar Object Classification

Let $\mathcal{L} = \{0, 1\}$ be two class labels to represent whether a 3D point locates on the planar object. Denote

$$\mathcal{Z} = \{z_1, z_2, \cdots, z_N\}$$

[1] The noise is removed by applying the method proposed in [10].

as a potential classification result of all points $\mathcal{X}$, where $z_i \in \mathcal{L}$ is the class label of the i-th point. Note that, $z_i = 1$ represents the i-th point that locates on a planar object. The MRF model [11] used in this work is defined as

$$E(\mathcal{Z}) = \sum_{i=1}^{N} D(z_i) + \gamma \sum_{i \sim j} S(z_i, z_j) \,, \tag{3}$$

where $D(\cdot)$ is the data term, $S(\cdot, \cdot)$ is the pairwise smooth term, $i \sim j$ represents the pairs of neighboring points, and γ is a weighting constant. In this paper, the data term is defined as

$$D(z_i) = \begin{cases} 1, & f_i \geq \theta \\ 0, & \text{otherwise} \end{cases} \,, \tag{4}$$

where θ is a positive threshold. The smooth term $S(z_i, z_j)$ is considered as the Potts model, given by

$$S(z_i, z_j) = \begin{cases} 1, & z_i = z_j \\ 0, & \text{otherwise} \end{cases} \,. \tag{5}$$

During implementation, the Graph-cut (GC) algorithm [12] is utilized to solve Eqn. (3).

After performing GC algorithm, all points $\mathcal{X}$ can be divided into two subsets, i.e., $\mathcal{X}^0$ and $\mathcal{X}^1$, satisfying that

$$\mathcal{X}^0 \cup \mathcal{X}^1 = \mathcal{X}, \quad \mathcal{X}^0 \cap \mathcal{X}^1 = \emptyset \,,$$

where $\mathcal{X}^0 = \{\mathbf{x}_i | z_i = 0\}_{i=1}^N$ and $\mathcal{X}^1 = \{\mathbf{x}_i | z_i = 1\}_{i=1}^N$. The points in subset $\mathcal{X}^1$ represent planar objects, including buildings and ground (see Fig. 2). To segment and classify buildings, the ground should be removed beforehand. The details of removal of the ground is presented in the following subsection.

3.2 Ground Detection via Constrained K-Means

In real application, the normals of buildings and ground are mutually perpendicular. Based on this observation, the ground is detected by classifying all normals into two types, i.e., ground normal and building normal. To obtain these two types of normals, we propose the constrained K-means clustering model, which is defined as

$$\begin{aligned} &\min_{\{\mathbf{m}_j\}_{j=1}^M} \sum_{i=1}^{N} \sum_{j=1}^{M} \|\mathbf{n}_i - \mathbf{m}_j\|_2^2 \delta_{i,j} \,, \\ &\text{s.t.} \quad \|\mathbf{m}_j\|_2^2 = 1, \quad \forall j \in \{1, 2, \cdots, M\} \,, \end{aligned} \tag{6}$$

where M is the number of clusters, $\{\mathbf{m}_j\}_{j=1}^M$ are M cluster centers, and $\delta_{i,j}$ is a Dirichlet function, satisfying that $\delta_{i,j} = 1$ if the i-th point belongs to the j-th

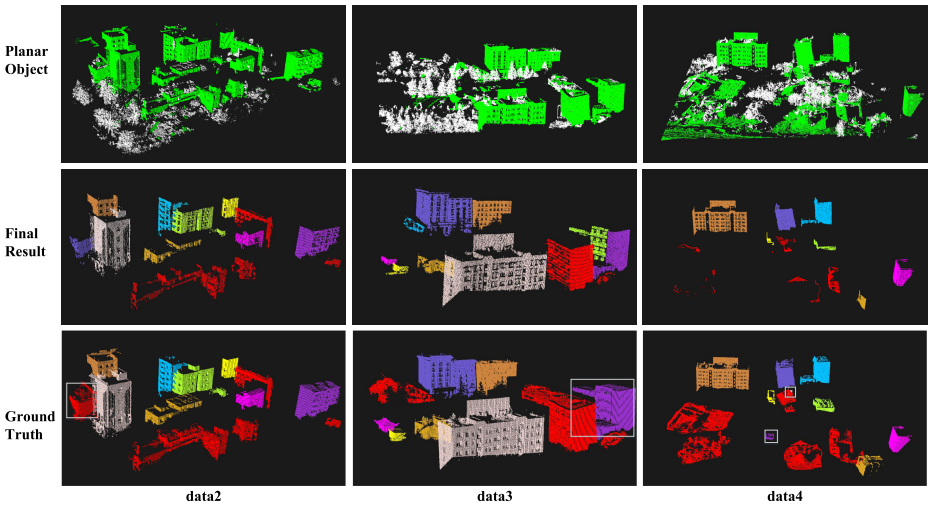

Fig. 3. Building segmentation and classification results on the other three data (each column). The first row represents the planar object by MRF. The second row shows our segmentation and classification results. The last row illustrates the ground truths labeled manually. The color indexes are the same with those of Fig. 2.

cluster, otherwise $\delta_{i,j} = 0$. In Eqn. (6), the constraint is to restrict the cluster centers to be normals. During implementation, the number of clusters M is set to be 3.

Optimization: The proposed constrained K-means in Eqn. (6) is optimized by iterating the following two steps (the max number of iterations is set to be 10):

Step_1: Computing normal clusters $\{\mathbf{m}_j\}_{j=1}^{M}$ by K-means.

Step_2: Restricting each normal cluster $\mathbf{m}_j$ with the normalizing operation, that is, $\mathbf{m}_j = \frac{\mathbf{m}_j}{\|\mathbf{m}_j\|_2}$.

Ground Detection: Let $\mathcal{I} = \{1, 2, \cdots, M\}$ be the indexes of all the normal clusters obtained by the proposed constrained K-means. In order to detect ground normal, we first define a perpendicular value for each normal cluster. For example, the perpendicular value p_i for the i-th normal cluster is calculated by

$$p_i = \sum_{j \in \{\mathcal{I}/i\}} \mathbf{m}_i^{\top} \mathbf{m}_j \ , \tag{7}$$

where $\mathcal{I}/i$ means all indexes except the index i. The index of the ground normal $i^{\star}$ is computed as the index with the minimal perpendicular value, given by

$$i^{\star} = \min_i \{p_1, p_2, \cdots, p_M\} \ . \tag{8}$$

Hence, the $i^{\star}$-th normal cluster $\mathbf{m}_{i^{\star}}$ is the ground normal.

For the i-th point, we first calculate the inner product q_i between its normal $\mathbf{n}_i$ and the ground normal $\mathbf{m}_{i^\star}$, that is, $q_i = \mathbf{n}_i^\mathsf{T}\mathbf{m}_{i^\star}$. The i-th point is detected as ground point if $|q_i| > \tau$, where τ is a positive threshold. During implementation, the threshold τ is set to 0.3 experimentally. After removing the ground points, the rest are building points, which are denoted as $\mathcal{X}_B$ (see Fig. 2).

3.3 Region Growing for Building Segmentation

The region growing method is used to segment candidate buildings. Specifically, after performing region growing algorithm, the building points $\mathcal{X}_B$ are clustered into C non-overlapped clusters, that is,

$$\mathcal{X}_B = \bigcup_{c=1}^{C} \mathcal{X}_B^c, \quad \mathcal{X}_B^i \cap \mathcal{X}_B^j = \emptyset, 1 \leq \ i \neq j \leq C \ , \tag{9}$$

where $\mathcal{X}_B^c$ is the c-th candidate building. We remove the small clusters with the number of points less than a threshold ξ. During implementation, the threshold ξ is set to 1000.

3.4 Building Classification

In practice, the tilt angles between abnormal and normal buildings are different. Based on this fact, we first compute a vertical feature for each candidate building. For example, the vertical feature v_c for the c-th candidate building $\mathcal{X}_B^c$ is calculated by

$$v_c = \frac{1}{N_c} \sum\nolimits_{j=1}^{N_c} \mathbf{m}_{i^\star}^\mathsf{T}\mathbf{n}_j \ , \tag{10}$$

where $N_c = |\mathcal{X}_B^c|$ is the number of points on c-th building and $\mathbf{m}_{i^\star}$ is the ground normal obtained by the constrained K-means (see Subsection 3.2). Then, a thresholding operation is used to classify all candidate buildings into two classes, that is, normal building and abnormal building, given by

$$l_c = \begin{cases} \text{normal building}, & v_c \geq \beta \\ \text{abnormal building}, & \text{otherwise} \end{cases} \ , \tag{11}$$

where l_c is the label of the c-th candidate building and β is a positive threshold (refer to Fig. 2 as example).

4 Experiments

In this section, we evaluate our building segmentation and classification method on the real dataset, which is obtained from Aerial LiDAR. Our algorithm is implemented in C++ on platform Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz with 4GB RAM.

Table 1. The description on the dataset.

	data1	data2	data3	data4
Num. of 3D Points	335172	209353	441796	534735
Num. of Buildings	7	14	10	15
Num. of Abnormal Buildings	2	6	3	6

4.1 Parameter Setting

Three major parameters, i.e., γ, θ and β are used for building segmentation and classification. During implementation, they are experimentally set as follows: 1) parameters γ and θ are set to $\gamma = 7500$ and $\theta = 25$ for MRF-based planar object classification; 2) in building classification, the threshold of β is set to $\beta = 0.08$.

4.2 Dataset Description

The dataset obtained from Aerial LiDAR is created to evaluate the effectiveness of our method (refer to Fig. 2 and Fig. 3 for visual perception). The 3D point numbers of them are 335172, 209353, 441796 and 534735, respectively. They contain 5, 8, 7 and 9 normal buildings, as well as 2, 6, 3 and 6 abnormal buildings (refer to Table 1). We manually labeled the ground truth for quantitative evaluation. There are several difficulties, which make building segmentation and classification problems challenging.

- Buildings are built on mountains. On the one hand, the buildings to be segmented are often shaded by the other objects, which results in severe data missing. On the other hand, the elevations of them are different with each other. Hence, they do not locate on the same ground plane.
- Some abnormal buildings are seriously destroyed. In this case, it is very hard to distinguish them from clutter objects, such as trees. For some abnormal buildings, the inclination angle is small, which causes that it is very hard to distinguish them from the normal buildings.

4.3 Visual Results

The visual comparisons of our method with the ground truth are shown in Fig. 2 and Fig. 3. With the local planer feature and ground normal estimation, the buildings are segmented and classified accurately. It is very close to the ground truth. Although our method can provide satisfactory segmentation and classification results, it still has some small errors. As shown in Fig. 3, in **data3**, one building is segmented into two buildings (see the white rectangle in the third row). In **data4**, two buildings are not segmented out. The main reason is that

Table 2. The building classification results on the dataset.

	data1	data2	data3	data4	Total
Num. of Buildings	7/7	14/14	9/10	13/15	**93.48%**
Num. of Abnormal Buildings	1/2	5/6	2/3	6/6	**82.35%**

the numbers of building points are very small. However, in practice, our method works well in building segmentation and classification, and it can be used in real world application.

4.4 Quantitative Results

To further evaluation the effectiveness of our method, the TPR criterion is adopted to evaluate the segmentation and classification results quantitatively. The criterion TPR is defined as follows:

$$TPR = \frac{TP}{TP + FN} , \tag{12}$$

where TP is the number of correct segmented (or classified) buildings, while FN is the number of missed segmented (or classified) buildings. Table 2 demonstrates the results of our method on the dataset. As illustrated in this table, our method achieves high TPR in building segmentation and classification.

5 Conclusion

In this paper, we propose a novel method for building segmentation and classification via local planar feature. The core idea is to detect planar objects from clutter 3D points. To evaluate the effectiveness of our method, the dataset is created, which contains major difficulties in building detection and classification. Experimental results on this dataset demonstrate the effectiveness of our method.

However, our method still has some limitations. For example, the very small building may be miss-detected, and two near buildings would be detected as one building. In addition, as some abnormal buildings are destroyed very seriously, we can only detect a part of them. In the future, we will solve the above problems to make our method more general.

References

1. Charaniya, A.P., Manduchi, R., Roberto M., Lodha, S.K.: Supervised parametric classification of aerial lidar data. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop, pp. 25–32 (2004)

2. Lalonde, J.-F., Unnikrishnan, R., Vandapel, N., Hebert, M.: Scale selection for classification of point-sampled 3-d surfaces. In: Fifth International Conference on 3-D Digital Imaging and Modeling, pp. 285–292 (2005)
3. Lalonde, J.-F., Vandapel, N., Huber, D., Hebert, M.: Natural terrain classification using three-dimensional ladar data for ground robot mobility. Journal of Field Robotics **23**(10), 839–861 (2006)
4. Secord, J., Zakhor, A.: Tree detection in aerial lidar and image data. Tech. Rep. (2006)
5. Carlberg, M., Gao, P., Chen, G., Zakhor, A.: Classifying urban landscape in aerial lidar using 3d shape analysis. In: Proceedings of the International Conference on Image Processing, pp. 1701–1704 (2009)
6. Lafarge, F., Mallet, C.: Creating large-scale city models from 3d-point clouds: A robust approach with hybrid representation. International Journal of Computer Vision **99**(1), 69–85 (2012)
7. Weinmann, M., Jutzi, B., Mallet, C.: Semantic 3d scene interpretation: a framework combining optimal neighborhood size selection with relevant features. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences **II–3**, 181–188 (2014)
8. Forlani, G., Nardinocchi, C., Scaioni, M., Zingaretti, P.: Complete classification of raw lidar data and 3d reconstruction of buildings. Pattern Anal. Appl. **8**(4), 357–374 (2006)
9. Matei, B.C., Sawhney, H.S., Samarasekera, S., Kim, J., Kumar, R.: Building segmentation for densely built urban regions using aerial lidar data. In: Computer Vision and Pattern Recognition, pp. 1–8 (2008)
10. Wang, J., Kai, X., Liu, L., Cao, J., Liu, S., Zeyun, Y., Xianfeng David, G.: Consolidation of low-quality point clouds from outdoor scenes. Comput. Graph. Forum **32**(5), 207–216 (2013)
11. Li, S.Z.: Markov Random Field Modeling in Image Analysis, 3rd edn. Springer Publishing Company, Incorporated (2009)
12. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence **23**(11), 1222–1239 (2001)