Engineering and Laboratory Note Vol. 58, No. 14 / 10 May 2019 / Applied Optics 3807

applied optics

Automated tree detection from 3D lidar images using image processing and machine learning

Kenta Itakura and Fumiki Hosoi*

Graduate School of Agricultural and Life Sciences, The University of Tokyo, Yayoi 1-1-1, Bunkyo-ku, Tokyo, 113-8657, Japan
**Corresponding author: ahosoi@mail.ecc.u-tokyo.ac.jp*

Received 6 March 2019; revised 5 April 2019; accepted 11 April 2019; posted 12 April 2019 (Doc. ID 360695); published 7 May 2019

Trees in 3D images obtained from lidar were automatically extracted in the presence of other objects that were not trees. We proposed a method combining 3D image processing and machine learning techniques for this automatic detection. Consequently, tree detection could be done with 95% accuracy. First, the objects in the 3D images were segmented one by one; then, each of the segmented objects was projected onto 2D images. Finally, the 2D image was classified into "tree" and "not tree" using a one-class support vector machine, and trees in the 3D image were successfully extracted.

https://doi.org/10.1364/AO.58.003807

1. INTRODUCTION

Accurate determination of a tree structure is necessary for a variety of applications within the fields of urban forestry, ecology, environmental protection, and forestry [1]. Trees have spatially complicated structures, which significantly influence urban environments through solar shading, transpiration, wind breaking, air purification, and soundproofing [2]. Therefore, it is important to obtain 3D information about tree structures, such as their number, location, trunk diameter, height, and shape.

To collect the data, lidar (light detection and ranging) can be used instead of manual measurement, which is laborious and time-consuming. It has been reported that structural parameters, such as tree trunk diameter, height, leaf inclination angle, and leaf area index, can be estimated accurately [3–5]. As a next step, it is desirable to automatically detect each tree and analyze its structural parameters [6]. Lovell *et al.* (2011) [7] proposed a method to detect the trees in a 3D point cloud model using the reflection intensity of a lidar laser beam. However, the type of lidar that can record reflection intensity is limited due to cost and technical reasons, and it is difficult to record the intensity with a lightweight and portable lidar [8] that is suitable for a large-area measurement with a mobile platform. A method that performs Hough transform or circle (cylinder) fitting to a cross section at a certain height of a 3D point cloud tree images and detects tree trunks has also been reported [6,9–12]. Nevertheless, if a tree trunk cross section forms an incomplete circle and/or includes tree branches and other noises, the cross section cannot be identified as a tree trunk [13]. Furthermore, these previous methods cannot detect trees accurately when the target site includes various objects other than trees because the other objects are mis-classified as trees. In particular, it becomes more difficult when the target site has an object such as a pole (called pole-like objects). In this study, our target is only the tree, and we do not have to classify the other objects. Generally, it becomes difficult, and a large amount of training data is needed to classify all the objects included in the 3D images. However, if we confine the target (i.e., trees) and construct a classifier that discriminates between a tree and "non-tree objects," then the trees can be detected efficiently and accurately. Therefore, we proposed an automatic tree detection method that combines 3D image processing and machine learning techniques in the presence of many objects other than trees.

2. MATERIALS AND METHOD

A. Study Site and Lidar

The campus of the University of Tokyo, Japan, was selected for the study site. It has many kinds of trees, including the Himalayan cedar (*Cedrus deodara*), Japanese zelkova (*Zelkova serrata*), maidenhair tree (*Ginkgo biloba*), and camellia. The 58.3 ha Shinjuku Gyoen National Garden in Tokyo, Japan, was also selected as a study site. There are more than 10,000 trees in the garden, such as the cherry tree (*Cerasus Mill.*), tulip tree (*Liriodendron tulipifera*), plane tree (*Platanus*), Himalayan cedar (*Cedrus deodara*), Formosan sweetgum (*Liquidambar formosana*), and bald cypress (*Taxodium distichum*). Three-dimensional images of the trees were obtained using lidar (221 trees). The lidar used in this study was a VLP-16 (Velodyne Lidar Inc., USA) with 16 laser sources rotating at a frequency of 10 Hz. The lidar measurement was conducted while walking, and the lidar was attached to a tripod and handheld, whose height and angle was about 2 m and 15°, respectively. The angular resolution (vertical) is 2°, and the resolution

(horizontal/azimuth) is 0.1 to 0.4°. The field of view is 360°. The wavelength of the laser beam is 905 nm.

B. Automatic Tree Detection Method

For the automatic tree detection, first, each object in the 3D images was segmented. Second, it was predicted whether the segmented object was a tree by projecting it onto 2D images. Finally, the detected trees were reconstructed in the 3D images. The overview for automatic tree segmentation from 3D images is illustrated in Fig. 1. Panel [i] is a 3D image obtained from lidar. First, the ground was automatically extracted; then, the ground and the other parts containing trees and other objects were separated, as shown in panels [ii'] and [ii]. For the ground extraction, the MLESAC algorithm [14], which is an improved version of RANSAC [15], was utilized. After excluding the ground, the points whose heights are from 0.5 to 1.0 m in the panel [ii] were focused on. Then, the points with a distance less than 0.3 m were classified into the same cluster; further, the neighboring points in the 3D images were segmented into a single cluster. Herein, a different object number was assigned to each cluster for their discrimination. Initially, only the points with heights of 0.5 to 1.0 m have the object number, while the other points do not. In panel [iii], each cluster has a different object number; consequently, each of the clusters is illustrated by a different color. Next, each cluster was expanded upward and downward, as shown in panel [iii]. In this process, object numbers were assigned gradually from the lower to the upper parts. By repeating this expansion process, each object was segmented, as shown in panel [iv]. For this segmentation, we referred to the method of Itakura and Hosoi [16].

It was then necessary to predict whether each object is a tree or another non-tree object. For this classification, a method combining deep learning and machine learning techniques was adopted. This is illustrated in Fig. 2. First, each of the segmented trees was picked, as shown in panel [i] in Fig. 2; then, the segmented 3D image was projected onto a 2D image (panel [ii]). Next, a pre-trained network, Alexnet [17], was imported. Alexnet is a deep neural network that delivered a remarkable result in image classification with ImageNet [18]. It consists of eight layers, and the layers from the initial layer to the seventh layer were used for feature extraction in this study. The projected 2D image was regarded as an input image, and the convolutional operation was performed on the image (panel [iii]). The input size of the images was 227 × 227 pixels; thus, from panel [i] to [ii] in Fig. 2, the image size was resized to 227 × 227 pixels. The weight and bias in the Alexnet were not changed because the network is only used for the feature extraction. After this feature extraction, 4096 features of the image were obtained.

Subsequently, whether the segmented 3D image was a tree was assessed based on the extracted features with a one-class support vector machine (SVM) [19]. To construct the one-class SVM, its training is necessary and is described in the next paragraph. The training data are the extracted features from the projected tree images after the convolution with Alexnet. One-class SVM fits a tight hyper-sphere in the nonlinearly transformed feature space to include most of the target images based on the positive examples (i.e., trees) [20]. The panel [iv] exemplifies the one-class SVM method. In the example, we assume that the sample has two features (features A and B). In the panel, the x and y axes represent features A and B, respectively, and each plot has a score value. In this study, a total of 4096 features were extracted by Alexnet. Then, the same calculation was conducted in 4096-dimensional space. The Gaussian kernel was used for the SVM in this study with a parameter (ν) that controls the number of support vectors used in the one-class SVM algorithm [21,22]. If the value calculated based on decision function $f(x)$ is positive, the image is classified as a

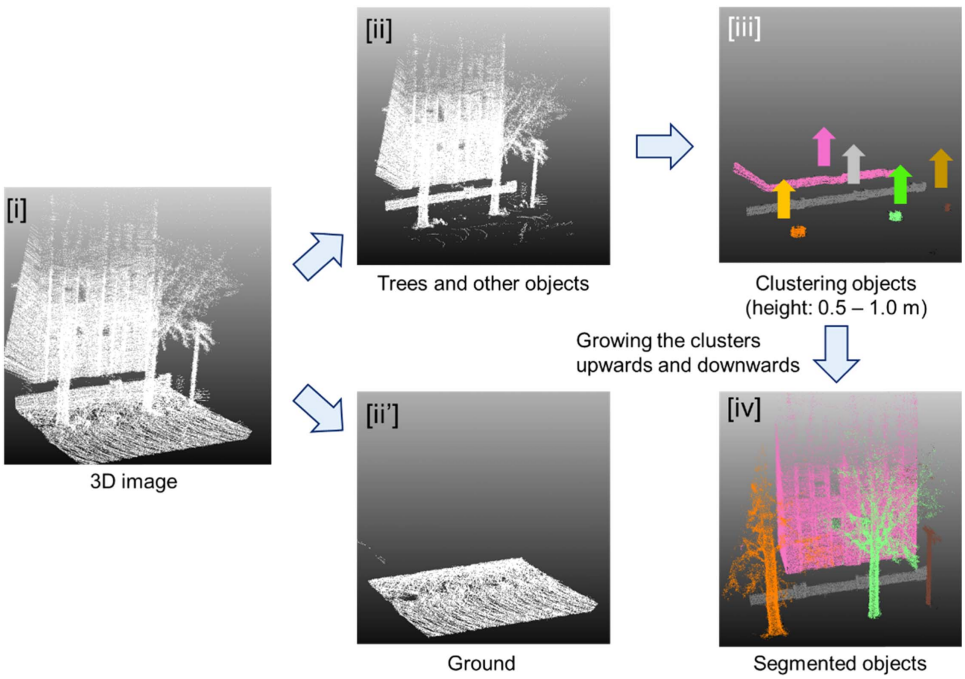

Fig. 1. Overview of automatic object segmentation from 3D images.

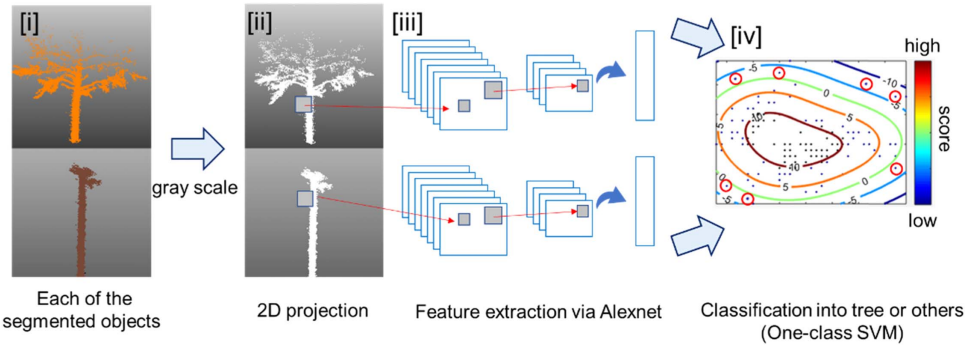

Fig. 2. Classification of trees and other non-tree objects from 3D images using deep learning and machine learning techniques.

tree; otherwise, it is classified as a non-tree. If the input image is not a tree, the score is supposed to be less than zero. With this threshold of the score (zero), we can predict whether the object is a tree. For the optimization of the decision function, the sequential minimal optimization (SMO) algorithm [23] was used. The PC used in this study was an Intel Core i7-6700. Processor-based frequency and random access memory are 3.40 GHz and 8.0 GB, respectively. A graphical processing unit is not used in this study, including the training. Therefore, this work can be implemented with a consumer-based PC.

For the training and test data, 221 images of trees obtained from the lidar mentioned in Section 2.1 were used. In the collection of the training and test data, we tried to obtain the trees whose shape is varied among them. When picking up each segmented object (panel [i] in Fig. 2), the object was rotated by $\pm 30°$ around the z axis for data augmentation [24] and then projected onto the 2D image (panel [ii] in Fig. 2). Because of the augmentation, the number of data was tripled. The tree images were split into 578 and 85 for training and test data, respectively. An additional 95 3D images, such as poles, traffic signals, humans, and buildings, were added after segmentation to the test images (panel [iv] in Fig. 1). The accuracy of the classification of the one-class SVM (panel [iv] in Fig. 2) was calculated with the test data.

After this prediction, each object predicted as a tree was reconstructed in the 3D images with a different tree number, and other non-tree objects were reconstructed with no tree numbers.

3. RESULTS AND DISCUSSION

In the test data, 3D images of trees and other objects, including a pole-like object, were classified with a high accuracy of 95.0%. Of the 85 test trees and 95 test non-trees, the system found 83 trees to be trees and seven non-trees to be trees. This suggests that the system is likely to find that a non-tree is found as a tree. The segmentation of each object is the first step for tree detection (Fig. 1, panels [iii] and [iv]). The accurate segmentation leads to tree detection using a one-class SVM. Generally, a region growing method [25] is useful and popular for image segmentation, which segments the objects by expanding an original (seed) point to the surrounding region and can be performed not only in 2D images but also in 3D images. However, in this case, this method cannot be helpful because the branches and leaves of neighboring trees are in contact in many cases. The branches (leaves) and other artificial objects are frequently in contact or otherwise overlap from the lidar's perspective in higher regions of the lidar capture. This means that neighboring trees and objects are segmented into the same cluster by the region growing method, which makes it impossible to segment each tree after the segmentation process. However, in this study, a section of a tree trunk, with a height of 0.5 to 1.0 m, was first focused on and the initial clusters were made. The tree trunk is only the part that does not touch the neighboring trees and objects. By utilizing this characteristic, each tree could be segmented. In the segmentation process, each tree could be separated from the 3D images accurately (Fig. 1, panel [iv]). Our target is the tree, and other objects are classified into one group. A convolutional neural network is known for its high classification accuracy. When the neural network is trained for the two-class classification of trees and other objects, training is difficult regarding the features of the "others." Because all objects other than trees belong to the "others," the network cannot cover all their features, resulting in a lower classification accuracy. In this study, we used a one-class SVM to exclude the non-tree objects and extract only trees from the segmented objects (Fig. 2, panel [i]). After the 2D projection in Fig. 2 [ii], feature extraction was performed via Alexnet (Fig. 2 [iii]). The pre-trained network is also useful for extracting features, and it outperformed many conventional hand-crafted feature techniques [26]. When the feature extraction process was skipped (i.e., the process in Fig. 2 [iii] was not done), and each pixel value of the image was input for its training data, the classification accuracy was 82.2%, which is significantly lower than the accuracy with the feature extraction. Therefore, we can say that feature extraction with Alexnet contributed to a high classification accuracy.

A similar prior work is available in [16]; however, this prior work includes only the procedure from the ground extraction to the clustering and growing of each region, as shown in Fig. 1. As a result, an object that can constitute a cluster, as shown in Fig. 1 [iii], was regarded as a tree. Then, all objects in the study site such as pedestrians, cars, poles, and other artificial objects are recognized as a tree. On the other hand, the segmented objects, as shown in Fig. 1 [iv], were classified using one-class SVM via feature extraction with Alexnet; therefore, the objects included in the study site could be discriminated accurately,

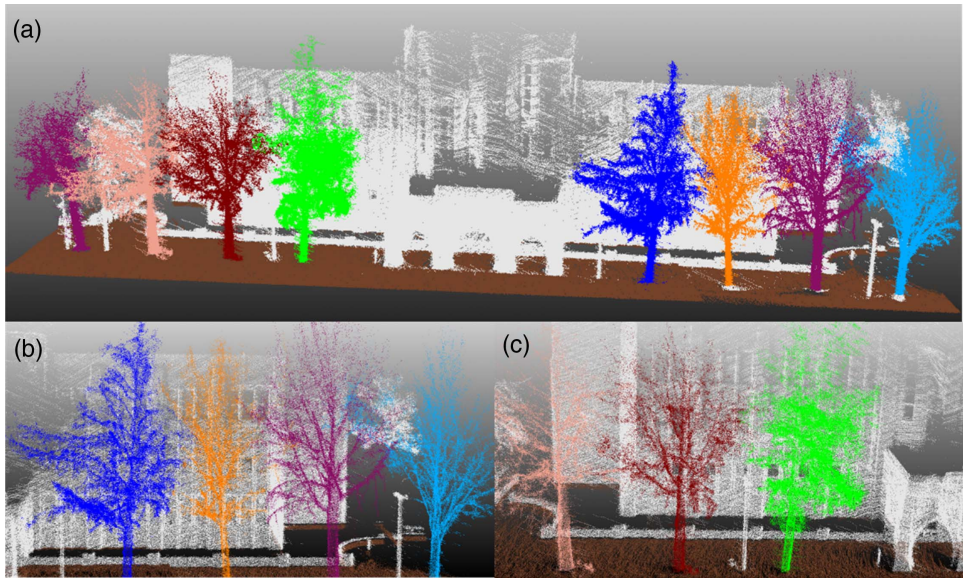

Fig. 3. Result of automatic tree detection from 3D images. The ground was colored brown, and each detected tree is illustrated by different colors, meaning different trees. The other objects are represented by white. (a) Overview of the target, (b) and (c) close up view.

and the tree detection was successfully done. In the prior work [16], most of the objects in the study site were trees, and other objects like a small shrub were cut off by simple thresholding. While the prior method cannot distinguish a tree from a non-tree object, the proposed method in this study enables tree detection in the presence of non-tree objects. Therefore, with the proposed method, the location, size, and other information related to tree shape can be automatically obtained even if there are many non-tree objects. We could investigate the trees in urban spaces and forest instead of conventional manual investigation. The feature extraction via Alexnet and classification with one-class SVM, as shown in Fig. 2, which is the additional process compared with the prior work [16], can be done fast. For example, to classify the total of 180 test data, it takes within 4 s with a consumer-based PC, meaning the time to classify one sample is less than about 0.02 s. Therefore, the processing time of this method and the previous study [16] was not significantly different. This method needs training trees unlike in the prior work [16]; however, the lidar measurement can be done while moving on a platform, which leads to the large-scale measurement. Therefore, significant cost for collecting the training data is not required.

Figure 3 shows an example of the automatic tree detection results using a 3D image. The ground was colored brown, and each detected tree is illustrated by different colors. The different colors represent different tree numbers. The objects that are not trees are represented by white. Panel [a] in Fig. 3 shows the overview of the target, and panels [b] and [c] show its close-up view. From the figures, we can say that the trees can be detected and segmented accurately, and the poles in this figure were not recognized as trees. For instance, in panel [c], a pole can be seen at the center of the image, and it is colored white, meaning it is recognized as a non-tree object after the classification by the one-class SVM (Fig. 2 [iv]). Moreover, as shown in panels [b] and [c], the branches and leaves were segmented accurately.

Methods for classifying objects in 3D images using a deep learning technique are available [27,28]. However, they need substantial training data and time for training because they try to segment all the objects observed in a scene. The method in this study is specific to the automatic detection of trees, and the features of other numerous objects are not learned. Therefore, this technique is simple and effective for tree detection from 3D images, and it offers high accuracy and lower costs for obtaining results. The mis-classification occurred when the projected image of the trees was similar to a pole because of the lack of leaves and branches. In future work, a method to exclude the non-tree objects from the segmented objects with higher accuracy is desirable, e.g., by combining a 3D deep learning method with this method. Also, we will try other tree species and investigate the impact on the classification accuracy.

Funding. ACT-I, Japan Science and Technology Agency (JST) (JPMJPR18U4).

REFERENCES

1. J. Morgenroth and C. Gomez, "Assessment of tree structure using a 3D image analysis technique—a proof of concept," Urban Forestry Urban Green. **13**, 198–203 (2014).
2. H. Oshio, T. Asawa, A. Hoyano, and S. Miyasaka, "Estimation of the leaf area density distribution of individual trees using high-resolution and multi-return airborne LiDAR data," Remote Sens. Environ. **166**, 116–125 (2015).
3. M. Dassot, T. Constant, and M. Fournier, "The use of terrestrial LiDAR technology in forest science: application fields, benefits and challenges," Ann. Fore. Sci. **68**, 959–974 (2011).
4. F. Hosoi and K. Omasa, "Estimation of vertical plant area density profiles in a rice canopy at different growth stages by high-resolution portable scanning lidar with a lightweight mirror," J. Photogram. Remote Sens. **74**, 11–19 (2012).
5. F. Hosoi, Y. Nakai, and K. Omasa, "3-D voxel-based solid modeling of a broad-leaved tree for accurate volume estimation using portable scanning lidar," J. Photogram. Remote Sens. **82**, 41–48 (2013).

6. H. Huang, Z. Li, P. Gong, X. Cheng, N. Clinton, C. Cao, W. Ni, and L. Wang, "Automated methods for measuring DBH and tree heights with a commercial scanning lidar," Photogram. Eng. Remote Sens. **77**, 219–227 (2011).
7. J. L. Lovell, D. L. B. Jupp, G. J. Newnham, and D. S. Culvenor, "Measuring tree stem diameters using intensity profiles from ground-based scanning lidar from a fixed viewpoint," J. Photogram. Remote Sens. **66**, 46–55 (2011).
8. Y. Pan, K. Kuo, and F. Hosoi, "A study on estimation of tree trunk diameters and heights from three-dimensional point cloud images obtained by SLAM," Eco Eng. **29**, 17–22 (2017).
9. M. Simonse, T. Aschoff, H. Spiecker, and M. Thies, "Automatic determination of forest inventory parameters using terrestrial laser scanning," in *Proceedings of the Scand Laser Scientific Workshop on Airborne Laser Scanning of Forests*, Umeå (2003), pp. 251–257.
10. J. G. Henning and P. J. Radtke, "Detailed stem measurements of standing trees from ground-based scanning lidar," Forest Sci. **52**, 67–80 (2006).
11. A. Bienert, S. Scheller, E. Keane, F. Mohan, and C. Nugent, "Tree detection and diameter estimations by analysis of forest terrestrial laser scanner point clouds," in *ISPRS Workshop on Laser, Scanning 2007 and SilviLaser 2007*, Espoo, Finland, September12–14, 2007, pp. 50–55.
12. K. Tansey, N. Selmes, A. Anstee, N. J. Tate, and A. Denniss, "Estimating tree and stand variables in a Corsican pine woodland from terrestrial laser scanner data," Int. J. Remote Sens. **30**, 5195–5209 (2009).
13. M. Thies and H. Spiecker, "Evaluation and future prospects of terrestrial laser scanning for standardized forest inventories," Forest **2**, 1 (2004).
14. P. H. S. Torr and A. Zisserman, "MLESAC: a new robust estimator with application to estimating image geometry," Comput. Vis. Img. Understand. **78**, 138–156 (2000).
15. M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," Commun. ACM **24**, 381–395 (1981).
16. K. Itakura and F. Hosoi, "Automatic individual tree detection and canopy segmentation from three-dimensional point cloud images obtained from ground-based lidar," J. Agri. Meteorol. **74**, 109–113 (2018).
17. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Adv. Neural Inf. Process. Syst. **60**, 1097–1105 (2012).
18. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2009), 248–255.
19. L. M. Manevitz and M. Yousef, "One-class SVMs for document classification," J. Mach. Learn. Res. **2**, 139–154 (2001).
20. Y. Chen, X. S. Zhou, and T. S. Huang, "One-class SVM for learning in image retrieval," in *International Conference on Image Processing* (IEEE, 2001), pp. 34–37.
21. B. Schölkopf, J. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," Neural Comput. **13**, 1443–1471 (2001).
22. B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," Neural Comput. **12**, 1207–1245 (2000).
23. J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Handbook of Advances in Kernel Methods—Support Vector Learning*, C. J. C. Burges, B. Schölkopf, and A. J. Smola, eds. (MIT, 1999), pp. 185–208.
24. O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, N. Navab, J. Hornegger, W. Wells, and A. Frangi, eds. (Springer, 2015), pp. 234–241.
25. R. Adams and L. Bischof, "Seeded region growing," IEEE Trans. Pattern Anal. Mach. Intell. **16**, 641–647 (1994).
26. M. G. Alaslani and L. A. Elrefaei, "Convolutional neural network-based feature extraction for iris recognition," Int. J. Comp. Sci. Info. Tech. **10**, 65–78 (2018).
27. G. Floros and B. Leibe, "Joint 2d-3d temporally consistent semantic segmentation of street scenes," in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2012), pp. 2823–2830.
28. Y. Zhou and O. Tuzel, "Voxelnet: end-to-end learning for point cloud based 3D object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 4490–4499.