# Semantic Segmentation of a Point Clouds of an Urban Scenes

Andrey Dashkevich[0000−0002−9963−0998]

National Technical University "Kharkiv Polytechnic Institute", Kharkiv 61002, Ukraine
dashkewich.a@gmail.com

**Abstract.** Semantic segmentation of images is a challenging task in computer vision. In our paper we present an algorithm for segmentation of images of urban scenes, acquired by methods of structure from motion. Our approach is based on extracting of depth and color features into a reduced parameter space. Our key contribution is a model of scene segmentation based on a $k$-nearest neighbor classifier in reduced color-depth space. Parameter space reduction is provided by splitting a parameter space on a regular grid in each major axis direction. Then we train $k$NN classifier to label pixels of input images as one of three categories: plants, roads and buildings.

**Keywords:** Depth Map, Semantic Segmentation, Structure From Motion, $k$NN Classifier, Color And Depth Parameter Space, Pixel Labeling.

## 1 Introduction

Automatic understanding of urban scenes is a challenging task in computer vision. There are many applications of it in the robotics, autonomous car driving and path planning [1, 2], unmanned aerial vehicles trajectory control [3], geoinformation systems. This leads to the need for development of efficient methods for solving the segmentation, classification and clustering of 3D models of urban environments. This research domain is actively progresses during last decades.

Majority of existing methods are based on correct 3D models reconstructed from data obtained by various techniques such as multi-view stereovision [4], structured light cameras [5, 6] or laser range scanners [2]. Such methods provide the only information about scenes in a form of raw point clouds thus the problem is to extract semantic and topological information from data.Some researches are dealt with segmentation tasks in presence of outliers and uncomplete information [7].

In our paper we present a method of segmentation of urban scenes, acquired by UAVs. Our approach is based on extracting of depth and color features into a parameter space. Our key contribution is a model of scene segmentation based on a $k$-nearest neighbor classifier in reduced color-depth space, such reduction is provided by splitting a parameter space on a regular grid in each axis direction. The goal of our classifier is to label each pixel of input image as one of three categories: plants, roads and buildings.

## 2 Related Work

Tasks of point clouds segmentation of environmental scenes can be divided into several research fields, such as an analysis of urban scenes, road maps, traffic scenes and detecting of obstacles [1,8–13], segmentation of indoor scenes [14–18], scene completion [19], material recognition [20].

The approaches used for segmentation vary from using of probabilistic models [21,22] to deep learning techniques [1,5,12,19,23,24]. Authors of [1] work propose method based on fully convolutional neural net to tasks of segmentation. In [21] authors utilize visual simultaneous and localization and mapping (SLAM) and a conditional random field (CRF) model to reconstruct scene from monocular videos and to label elements of scene for further parsing. Another probabilistic-based model, Markov random field, is used in [22] to build a semantic occupancy octree data structure. In [25,26] a Markov random field is used to contours extraction. Method of extraction of planar surfaces from a single RGB-D image by a template matching with probability map is developed by the authors of [16].

As to methods of geometrical scene representation there are several stixel-based algorithms were developed [10,27]. Another approach is based on voxel representation [8,28,29].

Several methods exist that based on energy minimization [28,30]. In [31] authors propose approach to segmentation based on optimization of ray energies passing through the objects in scene.

In [17] authors propose a pipeline for planar shapes segmentation based on RANSAC plane fitting.

## 3 Method Overview

### 3.1 Problem Statement

Given the image $I$ and corresponding depth map $M$ find a function $f\colon (I, M) \to R$, where $R = \{r_1, ..., r_m\}$ – is a set of disjoint regions of $I$ such that $\cup r_i = I, i = 1..m$.

In our work, we proceed from the assumption that pixels and depth values form a multi-dimensional space and groups of pixels that belong to semantic category are adjacent in the metric parameter space. Therefore, we can find class of pixel via it spatial proximity to one of the semantic clusters. The main goal of our approach is to find classification algorithm, which can classify images with affordable error rate. Another problem is to find corresponding parameter space that provides better error scores.

One of the main problem in neighbor search in metric spaces is the "curse of dimensionality": the volume of parameter space increases exponentially with increasing the number of dimensions. This leads to the data sparsity in high-dimensional spaces, thus we need methods for efficiently traverse such a spaces or ways to decrease the dimensionality of feature space. One of the ways to solve dimensionality problem is using of spatial data structures such as spatial hashes [32].

In our work we exploit the idea of $k$-nearest neighbors classification [33] in reduced search space that is achieved by splitting of the space into regular grid and building of a spatial hash-table for improving of search of nearest neighbors.

### 3.2 Parameter Space And Spatial Hashing

In our approach we assume that all of semantic categories of objects in image have unique color and depth information. Thus we can consider each pixel as a point $P = (r, g, b, d)$ in 4-dimensional space, where $(r, g, b)$ – is a red, green and blue components of pixel and $d$ – is a height level of pixel in corresponding depth map. As the pixel values can form sparse structures in parameter space we propose to split space into cells in each major axis direction in order to reduce search space. The number of cells $t$ is equal for each axis direction. Hence, we discretize space as follows:

$$P_{discr} = \frac{P}{G}, \tag{1}$$

where $G = \frac{\max g_j}{t}$ – is a cell size in each $j$-th axis direction, $g_j$ – values along given axis.

Then we can build spatial hash-table $H = \langle K, V \rangle$, where $K = hash(P_{discr})$ and $V$ – is a list of indices of points that belong to the cell with hash $H$. The hashing function is described as follows:

- For each point $P_{i_{discr}}$ of image $I$ we take a concatenation of string representations of its components $K = str(r_i) + str(g_i) + str(b_i) + str(d_i)$, where $str(\cdot)$ is a function that converts integer value to respective string value.

After finishing of spatial hash-table construction we get a dense spatial representation of initial parameter space. The length of each $V$ in hash-table corresponds to the weight of the cell that is indexed by $K$.

### 3.3 Semantic Labeling And Evaluation Metrics

Let us demonstrate our approach to classify and to label pixels of input image. Our approach consists of two main stages: training of the classifier and its evaluation. In the training stage we set $t$ as the main hyperparameter of our algorithm that determine grid resolution. Then we take manually labeled training images and corresponding depth maps and build separate hash-tables for each of the given object classes. After that we calculate weights of grid cells as a length of corresponding $V$ of hash-tables. Therefore, for each class $C_k$ we get a hash-table $H_k$ with weighted values of the cells.

In the evaluation stage we determine $P_t = P_{test_{discr}}$ for each pixel of test images and its corresponding hash $h = hash(P_t)$. Then we use $h$ as a key to get weights from all trained hash-tables and the class of the pixel is determined by a hash cell with maximal weight, thus we mark this pixel with corresponding label. In the final step we compare result from our algorithm with ground truth images by calculating the mean squared error as follows:

$$\epsilon = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2, \tag{2}$$

where $x_i$ – is a $i$-th pixel value obtained by our algorithm and $y_i$ – is a corresponding pixel value of the ground truth image.

## 4 Experimental Results And Discussion

We trained and tested our algorithm on image and depth map sequences obtained from a video of urban scene by methods of structure from motion. For each of three classes we build hash-tables based on image sequences. First, we test our approach on raw images and depth maps. Then we evaluate work of the algorithm with some preprocessing, in our experiments we smooth images with different kernel sizes, for depth maps we use Laplacian filtering instead of blurring. Image filtering was provided by means of OpenCV library and Python programming language.
Image data and algorithm parameters:

- We have frames of video with 24 Mpx resolution, after depth map calculation we reduce all images and depth maps to the size of $(W \times H) = (1200 \times 800)$ pixels each for decreasing computational cost.
- Grid resolution was taken from the set: $t = 4, 5, 8, 10, 64, 128, 256$.
- Kernel sizes for filtering was taken from the set: $k = 3, 7, 9, 15, 21$.

An example of input data image is provided in Fig. 1. For the learning of the classifier we take a patch from input image of scene and manually annotate it (see Fig. 2). In our experiments algorithm tries to teach one of the predefined class label: "Plants", "Roads", "Buildings", "Cars" and a special class "Undefined" for objects that have equal weights in each hash-table. We made two series of experiments: without classes "Cars" and "Undefined" and with them. Color scheme for annotation and representation of segmentation results: green color is for "Plants" class, red is for "Roads", blue is for "Buildings", cyan is for "Cars" and white is for undefined objects.

In Fig. 3 we demonstrate the examples of segmented images for different algorithm parameters for the experiments without "Cars" and "Undefined" classes. In Fig. 4 the examples of full set of classes segmentation is presented. In Fig. 5 the segmentation error scores are provided.

As it can be seen from results, our algorithm better classifies objects in experiments without classes "Cars" and "Undefined", that can be explained by a accuracy of manual annotation. Another important observation is the increasing of noise level with increasing of grid parameter $t$. Noise can be explained by used features that do not include geometrical characteristics, but only color information and depth value of points. The way of dealing with the shown drawbacks is the using of algorithms of feature extraction to avoid manual feature selection and image annotation.

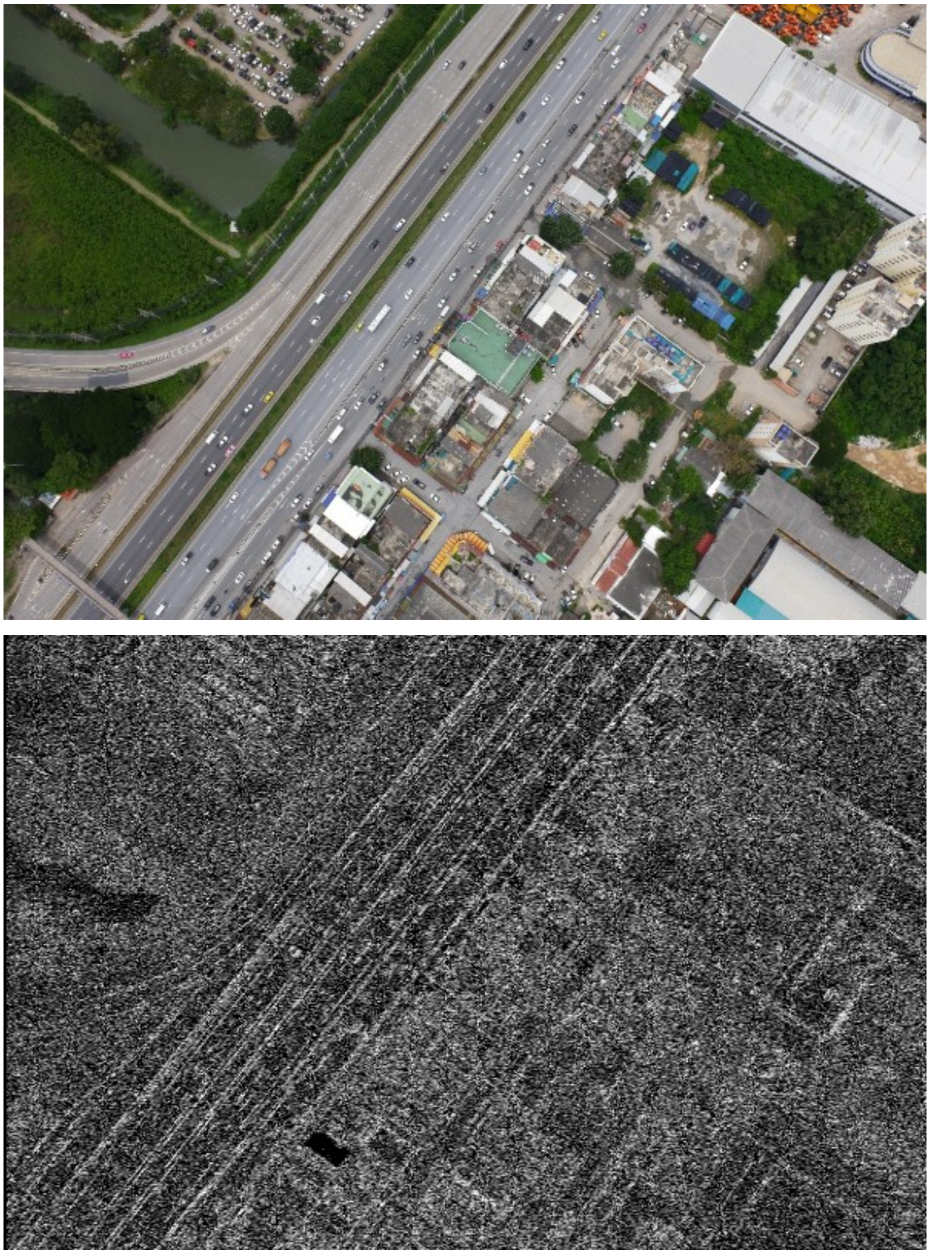


**Fig. 1.** Input frames (top), and corresponding depth map (bottom).

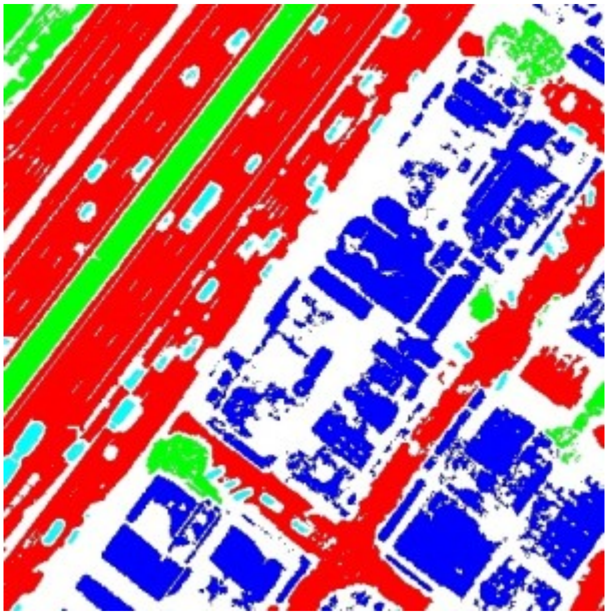
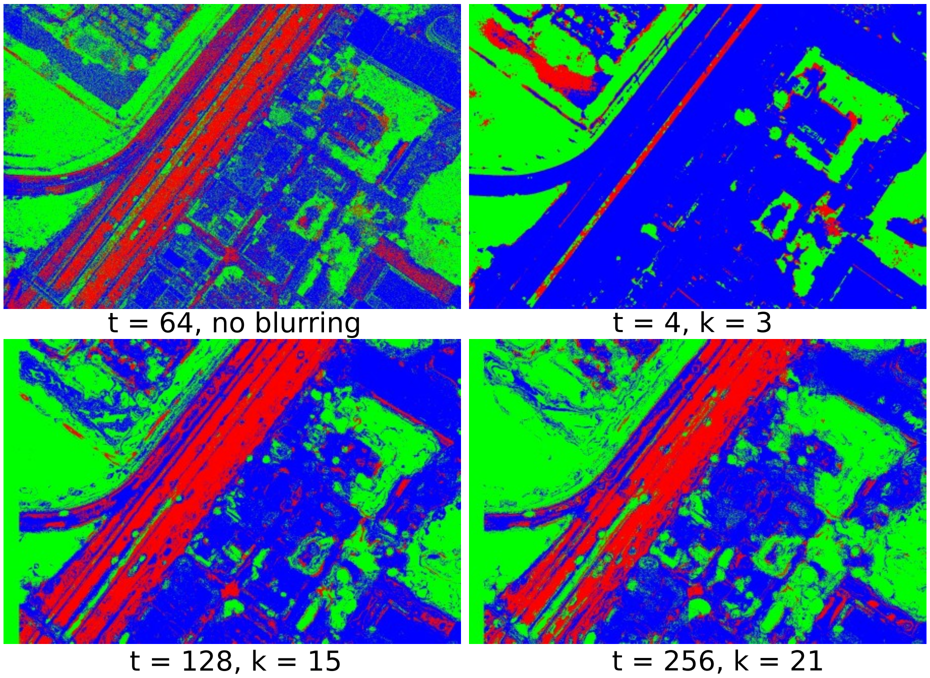


**Fig. 2.** Manually annotated patch of input image.



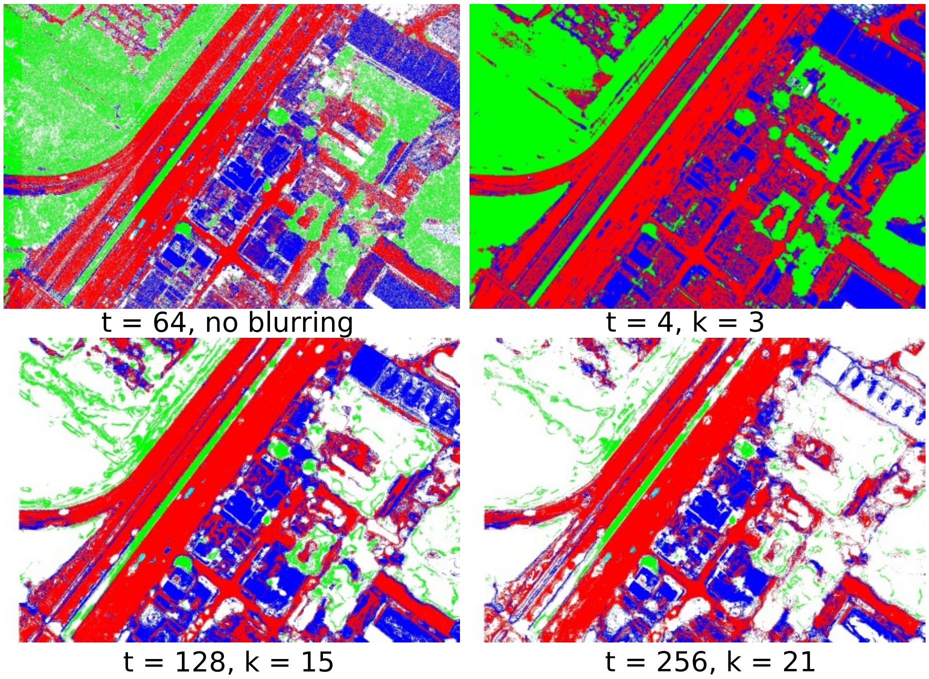
t = 64, no blurring

t = 4, k = 3

t = 128, k = 15

t = 256, k = 21

**Fig. 3.** Segmentation results without "Cars" and "Undefined" classes.



t = 64, no blurring

t = 4, k = 3

t = 128, k = 15

t = 256, k = 21

**Fig. 4.** Segmentation results with full set of classes.

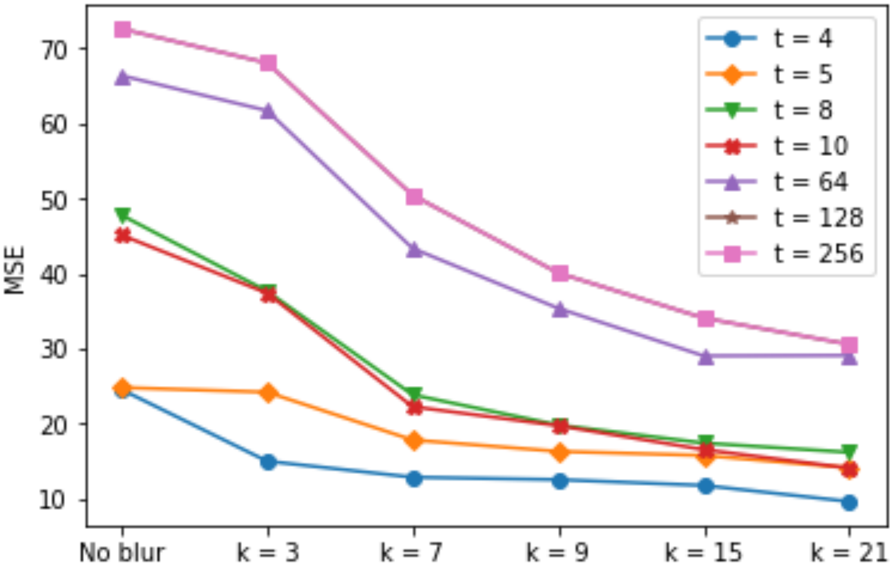


**Fig. 5.** Classification error metrics.

## 5 Conclusions

In our paper we demonstrate an approach for semantic segmentation of images based on a color and depth information by means of nearest neighbor search in parameter space. Our approach exploits the spatial hashing methods for reducing of search space to dense spatial structure and for fast search of points in it. The proposed algorithm is tested under different combination of grid resolution and smoothing kernels and implemented as a program utility. Also, we provide evaluation metrics for the algorithm, which show the ability of the approach to efficiently label images. One of the advantages of our algorithm is the possibility of adding of new classes by simply calculating of additional hash-table for new classes without refreshing of the old ones.

The limitation of our approach is that it not robust to outliers in data and we need to train it to classify as much objects classes as possible, therefore the future work is aimed to improve robustness to outliers. Another disadvantage is the manual feature selection that can be avoided with the use of automatic feature extractors, such as convolution autoencoders.

## References

1. Maturana, D., Chou, P.-W., Uenoyama, M., Scherer, S.: Real-Time Semantic Mapping for Autonomous Off-Road Navigation. In: Hutter, M. and Siegwart, R. (eds.) Field and Service Robotics. pp. 335–350. Springer International Publishing, Cham (2018)
2. Zermas, D., Izzat, I., Papanikolopoulos, N.: Fast segmentation of 3D point clouds: A paradigm on LiDAR data for autonomous vehicle applications. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 5067–5073. IEEE, Singapore, Singapore (2017).
3. Hepp, B., Nießner, M., Hilliges, O.: Plan3D: Viewpoint and Trajectory Optimization for Aerial Multi-View Stereo Reconstruction. ACM Transactions on Graphics. 38, 1–17 (2018). doi:10.1145/3233794.
4. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on. pp. 519–528. IEEE (2006).
5. Song, S., Xiao, J.: Deep Sliding Shapes for Amodal 3D Object Detection in RGB-D Images. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 808–816. IEEE, Las Vegas, NV, USA (2016).
6. Hänsch, R., Kaiser, S., Helwich, O.: Near Real-time Object Detection in RGBD Data: In: Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. pp. 179–186. SCITEPRESS - Science and Technology Publications, Porto, Portugal (2017).
7. Rock, J., Gupta, T., Thorsen, J., Gwak, J., Shin, D., Hoiem, D.: Completing 3D object shape from one depth image. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2484–2493. IEEE, Boston, MA, USA (2015).
8. Kochanov, D., Osep, A., Stuckler, J., Leibe, B.: Scene flow propagation for semantic mapping and object discovery in dynamic street scenes. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1785–1792. IEEE, Daejeon, South Korea (2016).

9. Boyko, A., Funkhouser, T.: Extracting roads from dense point clouds in large scale urban environment. ISPRS Journal of Photogrammetry and Remote Sensing. 66, S2–S12 (2011). doi:10.1016/j.isprsjprs.2011.09.009.
10. Ramos, S., Gehrig, S., Pinggera, P., Franke, U., Rother, C.: Detecting unexpected obstacles for self-driving cars: Fusing deep learning and geometric modeling. In: 2017 IEEE Intelligent Vehicles Symposium (IV). pp. 1025–1032 (2017).
11. Li, X., Ao, H., Belaroussi, R., Gruyer, D.: Fast semi-dense 3D semantic mapping with monocular visual SLAM. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). pp. 385–390 (2017).
12. Wen, C., Sun, X., Li, J., Wang, C., Guo, Y., Habib, A.: A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing. 147, 178–192 (2019). doi:10.1016/j.isprsjprs.2018.10.007.
13. Zia, M.Z., Stark, M., Schindler, K.: Towards Scene Understanding with Detailed 3D Object Representations. International Journal of Computer Vision. 112, 188–203 (2015). doi:10.1007/s11263-014-0780-y.
14. Nakajima, Y., Tateno, K., Tombari, F., Saito, H.: Fast and Accurate Semantic Mapping through Geometric-based Incremental Segmentation. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 385–392. IEEE, Madrid (2018).
15. Gupta, S., Arbeláez, P., Girshick, R., Malik, J.: Indoor Scene Understanding with RGB-D Images: Bottom-up Segmentation, Object Detection and Semantic Segmentation. International Journal of Computer Vision. 112, 133–149 (2015). doi:10.1007/s11263-014-0777-6.
16. Guo, R., Hoiem, D.: Support Surface Prediction in Indoor Scenes. In: 2013 IEEE International Conference on Computer Vision. pp. 2144–2151. IEEE, Sydney, Australia (2013).
17. Shui, W., Liu, J., Ren, P., Maddock, S., Zhou, M.: Automatic planar shape segmentation from indoor point clouds. In: Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry - VRCAI '16. pp. 363–372. ACM Press, Zhuhai, China (2016).
18. Ambrus, R., Claici, S., Wendt, A.: Automatic Room Segmentation From Unstructured 3-D Data of Indoor Environments. IEEE Robotics and Automation Letters. 2, 749–756 (2017). doi:10.1109/LRA.2017.2651939.
19. Liu, S., Hu, Y., Zeng, Y., Tang, Q., Jin, B., Han, Y., Li, X.: See and Think: Disentangling Semantic Scene Completion. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.) Advances in Neural Information Processing Systems 31. pp. 263–274. Curran Associates, Inc. (2018).
20. Degol, J., Golparvar-Fard, M., Hoiem, D.: Geometry-Informed Material Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1554–1562. IEEE, Las Vegas, NV, USA (2016).
21. Kundu, A., Li, Y., Dellaert, F., Li, F., Rehg, J.M.: Joint Semantic Segmentation and 3D Reconstruction from Monocular Video. In: Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 703–718. Springer International Publishing, Cham (2014).
22. Sengupta, S., Sturgess, P.: Semantic octree: Unifying recognition, reconstruction and representation via an octree constrained higher order MRF. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 1874–1879. IEEE, Seattle, WA, USA (2015).

23. Lahoud, J., Ghanem, B.: 2D-Driven 3D Object Detection in RGB-D Images. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 4632–4640. IEEE, Venice (2017).
24. Luo, Q., Ma, H., Wang, Y., Tang, L., Xiong, R.: Single Multi-feature detector for Amodal 3D Object Detection in RGB-D Images. CoRR. abs/1711.00238, (2017).
25. Hackel, T., Wegner, J.D., Schindler, K.: Joint classification and contour extraction of large 3D point clouds. ISPRS Journal of Photogrammetry and Remote Sensing. 130, 231–245 (2017). doi:10.1016/j.isprsjprs.2017.05.012.
26. Hackel, T., Wegner, J.D., Schindler, K.: Contour Detection in Unstructured 3D Point Clouds. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1610–1618. IEEE, Las Vegas, NV, USA (2016).
27. Cordts, M., Rehfeld, T., Schneider, L., Pfeiffer, D., Enzweiler, M., Roth, S., Pollefeys, M., Franke, U.: The Stixel world: A medium-level representation of traffic scenes. Image and Vision Computing. 68, 40–52 (2017). doi:10.1016/j.imavis.2017.01.009.
28. Kim, B.-S., Kohli, P., Savarese, S.: 3D Scene Understanding by Voxel-CRF. In: 2013 IEEE International Conference on Computer Vision. pp. 1425–1432. IEEE, Sydney, Australia (2013).
29. Aijazi, A., Checchin, P., Trassoudaine, L.: Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation. Remote Sensing. 5, 1624–1650 (2013). doi:10.3390/rs5041624.
30. Chen, X., Kundu, K., Zhu, Y., Berneshawi, A.G., Ma, H., Fidler, S., Urtasun, R.: 3D Object Proposals for Accurate Object Class Detection. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., and Garnett, R. (eds.) Advances in Neural Information Processing Systems 28. pp. 424–432. Curran Associates, Inc. (2015).
31. Savinov, N., Ladicky, L., Hane, C., Pollefeys, M.: Discrete optimization of ray potentials for semantic 3D reconstruction. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5511–5518. IEEE, Boston, MA, USA (2015).
32. Eitz, M., Lixu, G.: Hierarchical spatial hashing for real-time collision detection. In: Shape Modeling and Applications, 2007. SMI'07. IEEE International Conference on. pp. 61–70. IEEE (2007).
33. Liu, T., Moore, A.W., Yang, K., Gray, A.G.: An investigation of practical approximate nearest neighbor algorithms. In: Advances in neural information processing systems. pp. 825–832 (2005).