Region Merging Driven by Deep Learning for RGB-D Segmentation and Labeling

Umberto Michieli, Maria Camporese, Andrea Agiollo, Giampaolo Pagnutti, Pietro Zanuttigh
umberto.michieli@dei.unipd.it
Department of Information Engineering, University of Padova, Italy

ABSTRACT

Among the various segmentation techniques, a widely used family of approaches are the ones based on region merging, where an initial oversegmentation is progressively refined by joining segments with similar characteristics. Instead of using deterministic approaches to decide which segments are going to be merged we propose to exploit a convolutional neural network which takes a couple of segments as input and decides whether to join or not the segments. We fitted this idea into an existent iterative semantic segmentation scheme for RGB-D data. We were able to lower the number of free parameters and to greatly speedup the procedure while achieving comparable or even higher results, thus allowing for its usage in free navigation systems. Furthermore, our method could be extended straightforwardly to other fields where region merging strategies are exploited.

CCS CONCEPTS

• **Computing methodologies → Scene understanding**; **Image segmentation**; *Neural networks.*

KEYWORDS

Region Merging, Convolutional Neural Networks, Semantic Segmentation, Deep Learning.

ACM Reference Format:
Umberto Michieli, Maria Camporese, Andrea Agiollo, Giampaolo Pagnutti, Pietro Zanuttigh. 2019. Region Merging Driven by Deep Learning for RGB-D Segmentation and Labeling. In *13th International Conference on Distributed Smart Cameras (ICDSC 2019), September 9–11, 2019, Trento, Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3349801.3349810

1 INTRODUCTION

One of the key tools for free navigation and autonomous driving systems is a fast and accurate semantic segmentation method able to recognize the different objects and structures in the environment to be explored. This task encompasses two main components, the segmentation of the image into the different regions and objects it contains and the semantic labeling assigning each region to one of the possible classes.

Region merging is one of the most widely used families of algorithms for image segmentation. Many different approaches have been proposed for this task, mostly based on the idea of starting from an initial oversegmentation and then use some deterministic similarity criteria to decide which couples of segments are going to be combined during the steps of the merging algorithm. The reliability of the similarity criteria is the key issue for these methods. Several different clues, e.g., color, texture or edge information (and also depth or surface normals if 3D data is available) can be used for the task, but it is very challenging to combine all this information to provide reliable decisions on which segments need to be merged.

In this work we propose a different approach to guide the region merging process and we train a Convolutional Neural Network (CNN) in order to build a classifier able to select which couples of segments need to be fused. We sampled random couples of segments from a large dataset (i.e., the NYUDv2 dataset [27] containing both color and depth information) and we used the ground truth segmentation information to compute the labels for the training data indicating if the two segments need to be merged. In this way we obtain a CNN classifier able to control the merging process. We fitted it into a region merging framework derived from [23]. First of all an initial oversegmentation is performed from color and depth data [8]. Additionally, an initial semantic labeling that will be used to aid the segmentation has been computed with a simple CNN. The network for this task has been derived from the work of [21], however any semantic labeling network can be used. Notice that the aim of this work is not to present an advanced semantic classifier based on deep learning, but instead it focuses on the region merging process. Then a list of couples of segments candidate to be merged is built and sorted on the basis of the similarity of the semantic labeling. Finally the algorithm iteratively extracts a couple of segments from the list and uses the CNN classifier to decide whether they are going to be merged or not; the procedure stops when the list is empty. This leads to a final segmentation and semantic labeling which improves the accuracy of the initial semantic classification. This approach could be beneficial in many real world situations such as free navigation, grasping objects in robotics or in facing road scenarios in autonomous driving [19].

2 RELATED WORK

Image segmentation is a long-term research problem that remains challenging despite the huge number of proposed approaches. Recently, due to the diffusion of depth cameras, various works exploited the idea of using an associated depth map (see [32] for a review) as done also in this paper.

Segmentation of RGB-D data with region splitting and merging has been proposed in various works [11, 14, 22, 24, 28]. In [14] a statistical planar region merging scheme is used, while [24] exploits

ICDSC 2019, September 9–11, 2019, Trento, Italy

ACM ISBN 978-1-4503-7189-6/19/09…$15.00
https://doi.org/10.1145/3349801.3349810

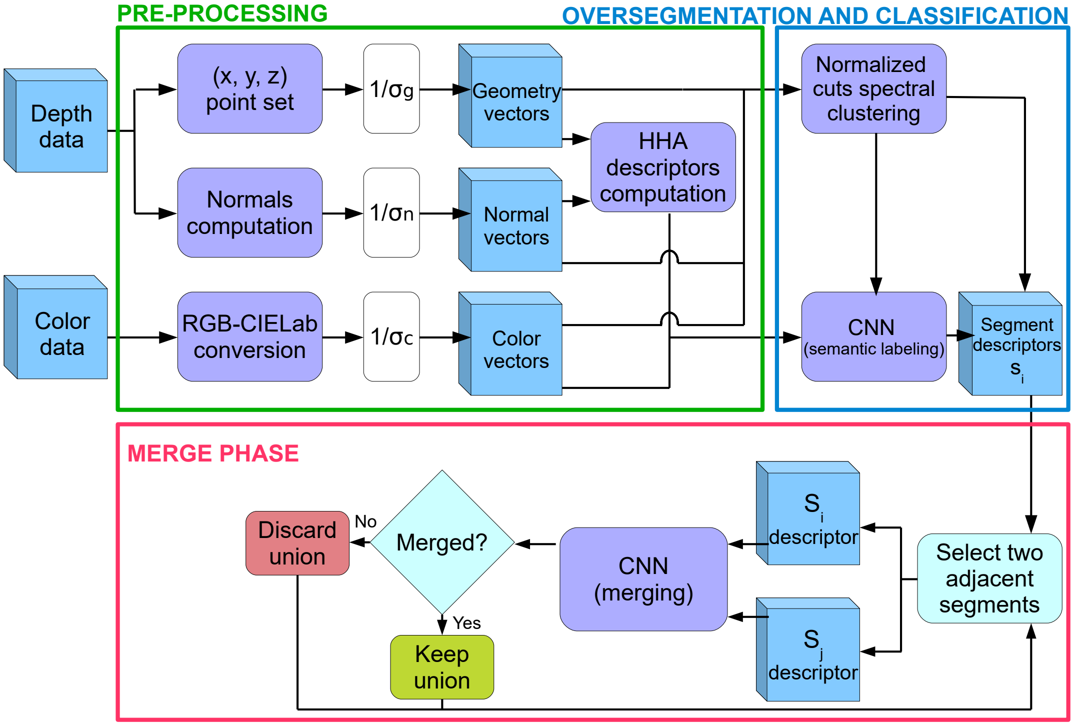

Figure 1: Overview of the proposed approach.

a Markov Random Fields (MRFs) model and a tree-structured segmentation. The work of [22] recursively splits segments that do not represent a single surface, while [23] uses the same criteria in a region merging scheme (that will be exploited as starting point for this work) starting from an initial oversegmentation.

Semantic segmentation, i.e., the combination of segmentation and semantic labeling, is one of the most challenging high-level tasks toward the direction of complete scene understanding and it is typically solved by using machine learning approaches [5, 16, 30, 33]. In [29] multiple segmentations are employed to generate the regions to be used for object detection and recognition. Multiple segmentations are used also by [4] that deals with the problem of object segmentation using a sequence of constrained parametric min-cut problems. Markov Random Fields (MRFs) and Conditional Random Fields (CRFs) have been exploited in various works [9, 15, 24]. Better performance, however, have been obtained with recent deep learning algorithms [2, 3, 7, 10, 13, 17, 25, 31]. In [7, 10] a scheme involving CNN at multiple scales has been adopted. The method of Wang et al. [31] exploits two different CNNs for color and depth, and a feature transformation network. Finally, remarkably good performance can be obtained with Fully Convolutional Networks (FCNs) [18, 25], auto-encoders [2] or residual networks [17].

3 ARCHITECTURE OF THE PROPOSED APPROACH

The proposed algorithm encompasses three main steps as depicted in Fig. 1. In the first, a 9-dimensional representation is built from the input information for each pixel p_k containing the CIELab color values $[L_k, a_k, b_k]$, the 3D coordinates $[x_k, y_k, z_k]$ and orientation data (i.e., the normal vectors $\mathbf{n_k} = [n_k^x, n_k^y, n_k^z]$). In the second step, an initial oversegmentation is computed using the approach of [8], that is an extended version of the normalized cuts algorithm [26]. Then, a simple CNN is used to get an initial semantic segmentation. We employed the deep learning architecture presented in [21] in order to allow a direct comparison with this approach, that uses a deterministic segment merging strategy inside the same framework used in this paper. Notice that, as expected, more advanced deep learning architectures have better performance, but proposing novel deep learning architectures for semantic labeling is not the aim of this work, that is focused on the region merging procedure. The last step, that is the main contribution of this work, is the iterative region merging procedure. The algorithm finds all the couples of adjacent segments, on the basis of the initial oversegmentation, and places them in the list $\mathcal{L}$ of merging candidates. Let us denote with S_i and S_j a generic couple of segments to be merged. As proposed in [20, 21] it is possible to use the contents of the last layer of the CNN classifier as per-pixel descriptors $\mathbf{c_k}$ (by interpolating the data from the reduced resolution of the CNN output to the original image resolution) that can then be averaged on each segment obtaining the segment-wise descriptors $\mathbf{s_i}$ and $\mathbf{s_j}$. The two descriptors can be viewed as Probability Density Functions (PDFs) and the similarity of the two segments can be estimated by computing the Bhattacharyya coefficient between $\mathbf{s_i}$ and $\mathbf{s_j}$, i.e.,: $b_{i,j} = \sum_t \sqrt{s_i^t s_j^t}$, where t indexes the elements of the last layer (i.e., the class scores). The list $\mathcal{L}$ is then sorted on the basis of $b_{i,j}$, with more similar segments at the top of the list. The entries with

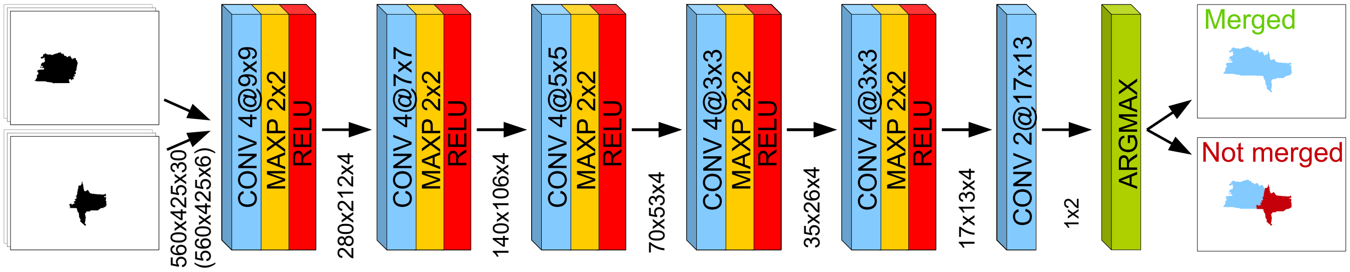

Figure 2: Architecture of the proposed neural network. The dimensionality of the input data is different depending on the usage of I_{PDF} or I_n.

similarity $b_{i,j}$ smaller than a threshold T_{sim} are discarded and will not be considered for the merging operations.

Finally, the iterative procedure extracts the first couple of segments from the list $\mathcal{L}$ and feeds it to the CNN classifier that decides whether it is going to be merged or not. We exploited and compared two variants of the deep network, one using normal information $\mathbf{n_k}$ and the other using pixel descriptors $\mathbf{c_k}$, see Section 4 for details on how the classification is performed. If the couple is not going to be fused it is simply removed from $\mathcal{L}$, otherwise a new segment corresponding to the union of the two is created. In this case, all the couples containing S_i and S_j are removed from $\mathcal{L}$ while the adjacency between the newly created segment and the contiguous ones is checked and eventually new couples are added to the list in the position corresponding to their similarity values. The procedure is repeated iteratively in a tree structure until no more merging operations are possible.

4 NEURAL NETWORK FOR REGION MERGING

In order to select the couples of segments to be merged we replaced the commonly used deterministic similarity metrics with a classifier based on deep neural networks. We developed a CNN that takes in input features coming from two segments candidate for merging and outputs a single binary value indicating if the two segments need to be fused or not.

To produce the input data for the training we used the NYUDv2 dataset [27]. First of all we performed an initial oversegmentation of the images and depth maps in the dataset using the approach of Section 3. Then for each segmented image we randomly extracted 10 couples of adjacent segments, leading to a total of 14490 data samples.

We considered two possible types of input features. In the first, for each pixel of a candidate segment we get the array $\mathbf{c_k}$ containing the output of the last layer of the semantic classifier (the output of the softmax before taking the argmax) interpolated to the image resolution, i.e., the same data used for the computation of the $b_{i,j}$ values used to sort the list $\mathcal{L}$. The dataset labels belong to 15 classes (including the *unknown* and the *unlabeled* classes), consequently in this case the input data has 15 channels. The values for all the other pixels not contained in the considered segment are set to 0. This leads to a matrix I_{PDF} of size 560x425x30 (15 channels for the representation of the first candidate segment and 15 for the second candidate segment) that represents the input data for the neural network.

The second option, instead, is to use the normal information $\mathbf{n}_k$ in place of the descriptors $\mathbf{c_k}$. Using the same approach we obtain a matrix I_n of size 560x425x6 (there are 3 components of the normal vector for each of the two segments). We selected the orientation data since it led to a higher accuracy than color or 3D positions in the experiments.

Notice that the first approach leads to a more descriptive representation, but has a higher dimensionality and introduces a dependency on the other deep network used for the semantic part. Using normal information is faster and simpler with a limited impact on the final accuracy.

The ground truth labels for the training have been computed by analyzing the region in the ground truth segmentation corresponding to the two segments and setting the label to true if more than 85% of the region belongs to a single segment (notice that the boundaries in the oversegmentation, in general, could not accurately match the ground truth ones).

The input matrix I_{PDF} (or I_n) is then fed to the neural network. The CNN is made of 6 convolutional layers (symmetrical padding is used for the boundaries), each followed by a ReLU activation and by a 2x2 max pooling except the last layer. The architecture of the network is depicted in Fig. 2, notice that larger convolutions are used at the beginning and smaller ones in the last layers. All the layers have 4 filters except the last one, while the pooling operations are designed in order to progressively reduce the resolution until a single feature is produced at the end of the network. The proposed architecture is quite simple, but notice that the available amount of training data is limited and the segments have a quite scarce information content: many of them have quite uniform properties and the key insights to be exploited are the differences between the data in the two segments. We also tried more complex architectures but they led to overfitting issues.

The networks have been trained for 50 epochs using the TensorFlow framework with a batch size of 32 samples. We used the ADAM optimizer for the loss composed by the sum of a cross-entropy term and a L2-regularization term. For the neural network based on the PDFs we used a similarity threshold $T_{sim} = 0.8$, a learning rate of 10^{-4} and a regularization constant of 10^{-3}. For the neural network based on the normal vectors we used $T_{sim} = 0.75$, a learning rate of 10^{-3} and a regularization constant of $5e^{-5}$. The training took around 3 hours with the version based on normals and

11 hours for the one exploiting PDFs (where the input vectors have a higher dimensionality) on a NVIDIA Titan X GPU. Inspecting the behavior of the algorithm it is possible to see that the accuracy is good on large segments but there are some issues on couples with small segments, which have a limited associated information and are intrinsically difficult to classify.

5 EXPERIMENTAL RESULTS

The performance of the proposed approach have been evaluated on the NYUDv2 dataset [27]. This dataset contains 1449 depth maps and color images of indoor scenes acquired with a first generation Kinect sensor divided into a training set with 795 scenes and a test set with the remaining 654 scenes. For results evaluation we used the ground truth labels from [12], and we clustered the original 894 categories into 15 classes using the mapping of [6]. As done in competing works [6, 10, 21] we excluded from the evaluation of the results the *unknown* and *unlabeled* classes.

Approach	*Pixel Accuracy*	*Class Accuracy*
Couprie et al. [7]	52.4%	36.2%
Hickson et al. [16]	53.0%	47.6%
A. Wang et al. [30]	46.3%	42.2%
J. Wang et al. [31]	54.8%	52.7%
A. Hermans et al. [15]	54.2%	48.0%
D. Eigen et al. [10]	75.4%	66.9%
Pagnutti et al. [21]	67.2%	54.4%
Semantic CNN	64.4%	51.7%
Our method (normals)	66.6%	53.6%
Our method (PDFs)	67.2%	54.5%

Table 1: Average pixel and class accuracies on the test set of the NYUDv2 dataset for some state-of-the-art methods from the literature and for the proposed method.

The numerical results for the semantic segmentation task are shown in Table 1. We report both the per-pixel accuracy and the average class accuracy (the latter is smaller since less frequent classes are also harder to recognize in many cases). The proposed approach using the PDFs obtains a mean pixel accuracy of 67.2% and a class accuracy of 54.5%.

To evaluate this result, first of all notice that the output of the classification performed by the initial semantic CNN of Section 3 has a mean pixel accuracy of 64.4% and a mean class accuracy of 51.7%. The improved accuracy shows how the segmentation procedure allows to refine the classification leading to more accurate boundaries and removing artifacts and noise from the original classification.

A second interesting comparison is with the approach of [21], that exploits the same initial semantic CNN together with a complex deterministic region merging procedure based on surface fitting clues. That approach has an average pixel accuracy of 67.2% and a class accuracy of 54.4%. Notice that the proposed method has very similar performance, indeed it achieves even slightly higher class accuracy, but it is much simpler, faster and, especially, does not rely on several hand-tuned thresholds as [21].

These considerations can also be evaluated visually by comparing the images in Fig. 3, where the results on four sample scenes from the test set are shown. In particular, it is possible to appreciate how the merging procedure refines the semantic segmentation output leading to more accurate boundaries (e.g., look at the bed in the second row or the table in the fourth row). Furthermore by comparing the third and the fourth columns it is possible to see that the proposed approach achieves similar results compared to [21]. There are some minor refinements but there is no clear winner, although the proposed approach is much simpler and faster.

Table 1 shows also the comparison with some competing approaches for which the results in the 13 classes setting are available. Notice how the proposed method is able to get good results and compete with more complex deep learning architectures even starting from the initial classification performed by a simple CNN. Recent more complex deep learning architectures, e.g., [10], have a higher semantic accuracy, however we do not aim at proposing advanced deep learning models for semantic segmentation. The target of the proposed work, instead, is to show how a deep neural network can efficiently control a region merging process and how this idea can be used to improve the accuracy of an initial semantic segmentation, even if performed by simple and not extremely powerful approaches. It is noticeable that by refining the boundaries with segmentation information it is possible to obtain an accurate representation of the shapes without using multi-resolution networks, skip connections, auto-encoders or other advanced deep network models.

It is possible to evaluate the performance of the proposed approach also using segmentation metrics (i.e., looking only at the segments' shapes without considering the class labels). Two commonly used metrics are the Rand Index (RI) and the Variation of Information (VoI) (see [1] for details on the metrics). The mean RI score (higher is better) on the test set increased from 0.82 of the CNN output to 0.87, while the mean VoI (lower is better) decreased from 2.77 to 2.00.

Finally we present some observation on the computation time, since one of the main claims is that this approach is faster than the previous deterministic method. We focus on the iterative region merging procedure: the proposed deep network can be evaluated in 22ms if normals are employed or 101ms if PDFs are used on an Intel Core i7-8700K CPU @3.70GHz. By using the GPU for the inference the computation time can be strongly reduced: for example, on a NVIDIA GeForce GTX 1070 the inference call requires on average 2ms in case of normals, and 10ms in case of PDFs. As a comparison the method of [21] based on surface fitting required on average 58ms for each evaluation. Notice that the oversegmentation and semantic classification steps are the same for both methods (and can be replaced with other superpixel segmentation schemes or different deep networks). Another interesting aspect is related to the stability of the proposed approach, which always requires the same amount of time for each couple of segments, while the computation time of [21] is heavily dependent on the area to be fitted.

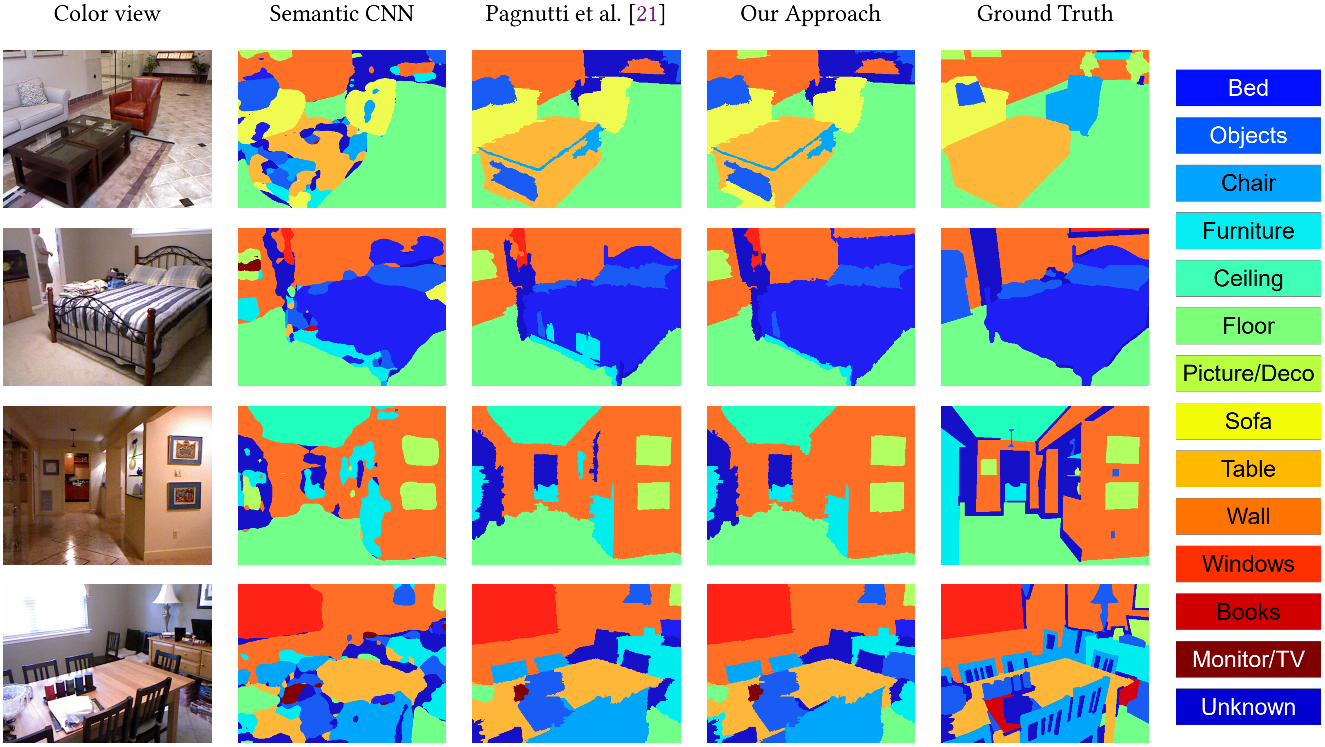

Figure 3: Semantic labeling of 4 sample scenes from the NYUDv2 dataset (images 465, 947, 1256 and 1349). The figure shows the color images, the labeling from the semantic CNN, the refined labeling obtained with [21] and by exploiting the proposed approach and finally the ground truth labels (*best viewed in color*).

6 CONCLUSION AND FUTURE WORK

In this paper we proposed a novel region merging strategy for RGB-D data segmentation where the decision on the segments to be merged is driven by a CNN binary classifier that replaces deterministic criteria, along with several free parameters, used up to now. We showed how the proposed classifier is able to reliably select the merging operations to be performed and we fitted it into an iterative region merging framework for semantic segmentation, although the framework allows wider applications where candidate segments, of arbitrary nature, need to be evaluated for merging. Experimental results show how it obtains the same performance of complex deterministic schemes with a smaller computation time and without using several hand-tuned thresholds. The faster computation time and the better generalization properties allow to use this approach in challenging tasks where a reliable semantic understanding of the scene is required, like in autonomous driving or in free navigation systems.

Further research will be devoted to combining the proposed approach with state-of-the-art deep learning approaches, to better focus the attention of the CNN on the boundary between the candidate segments and to its application in different fields where region merging strategies can be exploited. We will also extend the approach to video information, that is typical of autonomous navigation applications, by introducing temporal constraints into the proposed framework.

7 ACKNOWLEDGMENTS

We gratefully acknowledge NVIDIA for the donation of the GPU used for training the CNN and Gianluca Agresti for the insightful discussion.

REFERENCES

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. 2011. Contour Detection and Hierarchical Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 5 (2011), 898–916.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. 2017. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 12 (2017), 2481–2495.

[3] Matteo Biasetton, Umberto Michieli, Gianluca Agresti, and Pietro Zanuttigh. 2019. Unsupervised Domain Adaptation from Synthetic Data for Autonomous Vehicle Scenarios. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Workshop on Autonomous Driving (WAD), Long Beach (US)* (2019).

[4] Joao Carreira and Cristian Sminchisescu. 2012. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 7 (2012), 1312–1328.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2018. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 4 (2018), 834–848.

[6] C. Couprie, C. Farabet, L. Najman, and Y. LeCun. 2013. Indoor semantic segmentation using depth information. In *International Conference on Learning Representations (ICLR)*.

[7] C. Couprie, C. Farabet, L. Najman, and Y. Lecun. 2014. Convolutional nets and watershed cuts for real-time semantic Labeling of RGBD videos. *Journal of Machine Learning Research* 15, 1 (2014), 3489–3511.

[8] C. Dal Mutto, P. Zanuttigh, and G.M. Cortelazzo. 2012. Fusion of geometry and color information for scene segmentation. *IEEE Journal of Selected Topics in Signal Processing* 6, 5 (2012), 505–521.

[9] Z. Deng, S. Todorovic, and L. Jan Latecki. 2015. Semantic Segmentation of RGBD Images with Mutex Constraints. In *Proceedings of International Conference on Computer Vision (ICCV)*. 1733–1741.

[10] D. Eigen and R. Fergus. 2015. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of International Conference on Computer Vision (ICCV)*. 2650–2658.

[11] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik. 2015. Indoor scene understanding with RGB-D images: Bottom-up segmentation, object detection and semantic segmentation. *International Journal of Computer Vision* 112, 2 (2015), 133–149.

[12] S. Gupta, P. Arbelaez, and J. Malik. 2013. Perceptual Organization and Recognition of Indoor Scenes from RGB-D Images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[13] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. 2014. Learning rich features from RGB-D images for object detection and segmentation. In *Proceedings of European Conference on Computer Vision (ECCV)*. Springer, 345–360.

[14] M. Hasnat, O. Alata, and A. Tremeau. 2016. Joint Color-Spatial-Directional clustering and Region Merging (JCSD-RM) for unsupervised RGB-D image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (2016), 2255–2268. Issue 11.

[15] A. Hermans, G. Floros, and B. Leibe. 2014. Dense 3D semantic mapping of indoor scenes from rgb-d images. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*. IEEE, 2631–2638.

[16] S. Hickson, I. Essa, and H. Christensen. 2015. Semantic Instance Labeling Leveraging Hierarchical Segmentation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*. 1068–1075.

[17] G. Lin, A. Milan, C. Shen, and I. Reid. 2017. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[18] J. Long, E. Shelhamer, and T. Darrell. 2015. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[19] U. Michieli and L. Badia. 2018. Game Theoretic Analysis of Road User Safety Scenarios Involving Autonomous Vehicles. In *2018, IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*. 1377–1381.

[20] L. Minto, G. Pagnutti, and P. Zanuttigh. 2016. Scene segmentation driven by deep learning and surface fitting. In *Proceedings of European Conference on Computer Vision Workshops (ECCVW)*. Springer.

[21] G. Pagnutti, L. Minto, and P. Zanuttigh. 2017. Segmentation and semantic labelling of RGBD data with convolutional neural networks and surface fitting. *IET Computer Vision (IETCV)* 11, 8 (2017), 633–642.

[22] G. Pagnutti and P. Zanuttigh. 2014. Scene segmentation from depth and color data driven by surface fitting. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*. 4407–4411.

[23] G. Pagnutti and P. Zanuttigh. 2016. Joint Color and Depth Segmentation based on Region Merging and Surface Fitting. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*. 93–100.

[24] X. Ren, L. Bo, and D. Fox. 2012. RGB-D scene labeling: Features and algorithms. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[25] E. Shelhamer, J. Long, and T. Darrell. 2016. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (2016), 640–651. Issue 4.

[26] J. Shi and J. Malik. 2000. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 8 (2000), 888–905.

[27] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. 2012. Indoor segmentation and support inference from RGBD images. In *Proceedings of European Conference on Computer Vision (ECCV)*. Springer.

[28] N. Srinivasan and F. Dellaert. 2014. A Rao-Blackwellized MCMC Algorithm for Recovering Piecewise Planar 3D model from Multiple View RGBD Images. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*.

[29] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. 2013. Selective search for object recognition. *International Journal of Computer Vision* 104, 2 (2013), 154–171.

[30] A. Wang, J. Lu, G. Wang, J. Cai, and T. Cham. 2014. Multi-modal unsupervised feature learning for RGB-D scene labeling. In *Proceedings of European Conference on Computer Vision (ECCV)*. 453–467.

[31] J. Wang, Z. Wang, D. Tao, S. See, and G. Wang. 2016. Learning Common and Specific Features for RGB-D Semantic Segmentation with Deconvolutional Networks. In *Proceedings of European Conference on Computer Vision (ECCV)*. 664–679.

[32] P. Zanuttigh, G. Marin, C. Dal Mutto, F. Dominio, L. Minto, and G.M. Cortelazzo. 2016. *Time-of-flight and structured light depth cameras*. Springer.

[33] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. 2017. Pyramid scene parsing network. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2881–2890.