RIU-Net: Embarrassingly simple semantic segmentation of 3D LiDAR point cloud

Pierre Biasutti[1,2,3], Aurélie Bugeau[1], Jean-François Aujol[2], Mathieu Brédif[3]
[1]*Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, F-33400, Talence, France*
[2]*Univ. Bordeaux, IMB, INP, CNRS, UMR 5251, F-33400 Talence, France*
[3]*Univ. Paris-Est, LASTIG GEOVIS, IGN, ENSG, F-94160 Saint-Mandé, France*
pierre.biasutti@labri.fr

arXiv:1905.08748v2 [cs.CV] 6 Jun 2019

***Abstract*— This paper proposes RIU-Net (for Range-Image U-Net), the adaptation of a popular semantic segmentation network for the semantic segmentation of a 3D LiDAR point cloud. The point cloud is turned into a 2D range-image by exploiting the topology of the sensor. This image is then used as input to a U-net. This architecture has already proved its efficiency for the task of semantic segmentation of medical images. We demonstrate how it can also be used for the accurate semantic segmentation of a 3D LiDAR point cloud and how it represents a valid bridge between image processing and 3D point cloud processing. Our model is trained on range-images built from KITTI 3D object detection dataset. Experiments show that RIU-Net, despite being very simple, offers results that are comparable to the state-of-the-art of range-image based methods. Finally, we demonstrate that this architecture is able to operate at 90fps on a single GPU, which enables deployment for real-time segmentation.**

I. Introduction

The recent interest for autonomous systems have motivated many computer vision works over the past years. Indeed, the importance of accurate perception models is a crucial step towards systems automation, especially for mobile robots and autonomous driving. Modern systems are equipped with both optical cameras and 3D sensors (namely LiDAR sensors). LiDAR sensors are essential components of modern perception systems as they enable direct space measurements, providing accurate 3D representation of the scene. However, for most automation-related tasks, the raw LiDAR point cloud needs further processing in order to be used. In particular, point cloud with accurate semantic segmentation provides a higher level of representation of the scene that can be used in various applications such as obstacle avoiding, road inventor or object manipulation.

This work focuses on semantic segmentation of 3D LiDAR point clouds. Given a point cloud acquired with a LiDAR sensor, we aim at estimating a label for each point that belongs to objects of interests in urban environment (such as cars, pedestrians and cyclists). The traditional pipelines used to tackle this problem consider ground removal, clustering of remaining structures, and classification based on handcrafted features extracted on each clusters [8], [6]. The segmentation can be improved with variational models [12]. These methods are often hard to tune as handcrafted features usually require tuning many parameters, which is likely to be data dependant and therefore hard to use in a general scenario. Finally, although the use of regularization can lead to visual and qualitative improvements, it often leads to a large increasing of the computational time.

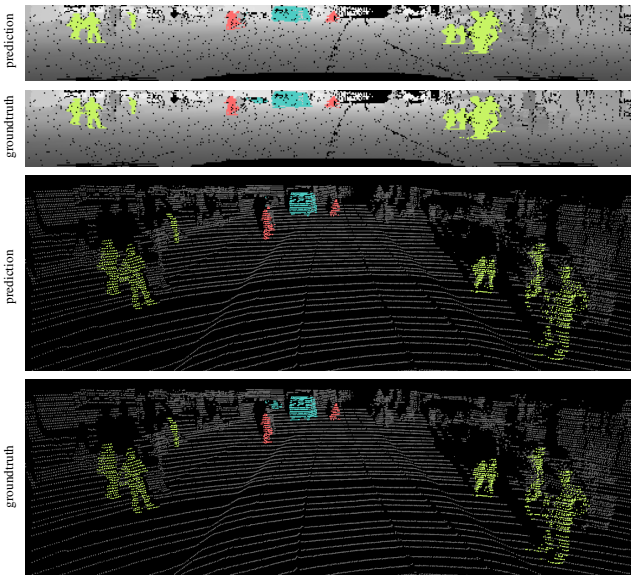

Fig. 1. Result of the range-image semantic segmentation produced by the proposed method. The first two results show the prediction of the proposed model and the groundtruth respectively, seen in the sensor topology. The last two results show the same prediction and groundtruth in 3D.

Recently, deep-learning approaches have been proposed to overcome the problem of the difficulty of tuning handcrafted features. This has become possible with the arrival of large 3D annotated datasets such as the KITTI 3D object detection dataset [7]. Many methods have been proposed to segment the point cloud by directly operating in 3D [16] or on voxel-based representation of the point cloud [22]. However, this type of methods either need very high computational power, or are not able to process a point cloud that corresponds to a full turn of the sensor in a single pass. Recently, faster approaches have been proposed [19], [18]. They rely on a 2D representation of the point cloud, called range-image, that can be used as the input of a convolutional neural network. Thus, the processing time as well as the required computational power can be kept low.

In this paper, we propose RIU-Net (for Range-Image U-

Net), the adaptation of U-Net [17], a very popular semantic segmentation architecture, to the semantic segmentation of 3D LiDAR point clouds. We demonstrate that, beside being a very simple architecture, the results of RIU-Net are comparable state-of-the-art range-image methods, as shown in Figure 1.

The contributions of the paper are the followings: 1) a simple adaptation of the method presented in [17] for the accurate semantic segmentation of 3D LiDAR point cloud, 2) a comparison with state-of-the-art methods in which we show that RIU-Net performs better on the same training set. The paper is organized as follows: first, previous works on point cloud semantic segmentation are presented. After that, the details of the adaptation of the model presented in [17] are explained. Finally, qualitative and quantitative results are shown and a conclusion is drawn.

II. Related works

In this section, previous works on image semantic segmentation as well as 3D point cloud semantic segmentation are presented.

A. Semantic segmentation for images

Semantic segmentation of images has been the subject of many works in the past years. Recently, deep-learning methods have largely outperformed existing methods. The method presented in [15] was the first to propose an accurate end-to-end network for semantic segmentation. This method is based on an encoder in which each scale is used to compute the final segmentation. Only a few month later, the U-net architecture [17] (later generalized in [1]) has been proposed for the semantic segmentation of medical images. This method is an encoder-decoder that is able to reach very fine precision in the segmentation. These two methods have largely influenced recent works such as DeeplabV3+ [5] that uses dilated convolutional layers and spatial pyramid pooling modules in an encoder-decoder structure to improve the quality of the prediction. Other approaches explore multi-scale architectures to produce and fuse segmentations performed at different scales [14], [21]. Most of these methods are able to produce very accurate results, on various types of images (medical, outdoor, indoor). The review [3] of CNNs methods for semantic segmentation provides a deep analysis of some recent techniques. This work demonstrates that a combination of various components would most likely improve segmentation results on wider classes of objects.

B. Semantic segmentation for point clouds

a) 3D-based methods: As mentioned above, the first approaches for point cloud semantic segmentation were done using heavy pipelines, composed of many successive steps such as: ground removal, point cloud clustering, feature extraction as presented in [8], [6]. However, these methods often require many parameters and they are therefore hard to tune. Recently, Landrieu et al. proposed in [11] to extract features of the point cloud using a deep-learning approach. Then, the segmentation is done using a variational regularization. Another approach presented in [16] proposes to directly input the raw 3D LiDAR point cloud to a network composed of a succession of fully-connected layers to classify or segment the point cloud. However, due to the heavy structure of this architecture, it is only suitable for small point clouds. Moreover, processing 3D data often increases the computational time due to the dimension of the data (number of points, number of voxels), and the absence of spatial correlation. To overcome these limitations, the methods presented in [13] and [22] propose to represent the point cloud as a voxel-grid which can be used as the input of 3D CNN. These methods achieve satisfying results for 3D detection. However, semantic segmentation would require a voxel-grid of very high resolution, which would increase the computational cost as well as the memory usage.

b) Range-image based methods: Recently, Wu et al. proposed SqueezeSeg [19], a novel approach for the semantic segmentation of a LiDAR point cloud represented as a spherical range-image [2]. This representation allows to perform the segmentation by using simple 2D convolutions, which lowers the computational cost while keeping good accuracy. The architecture is derived from the SqueezeNet image segmentation method [10]. The intermediate layers are "fire layers", *i.e.* layers made of one squeeze module and one expansion module. Later on, the same authors improved this method in [20] by adding a context aggregation module and by considering focal loss and batch normalization to improve the quality of the segmentation. A similar range-image approach was proposed in [18], where a Atrous Spatial Pyramid Pooling [4] and squeeze reweighting layer [9] are added. These range-image methods succeed in real-time computation but require heavy architectures including post-processing steps.

In next section, we propose RIU-Net: an adaptation of U-net [17] for the semantic segmentation of point clouds represented as range-images.

III. Methodology

In this section, we present RIU-Net, our adaptation of the U-net architecture [17] for the semantic segmentation of LiDAR point clouds. The method consists in feeding the U-net architectures with 2-channels images encoding range and elevation. In the next sub-section, we explain how to build these images, called range-images, that were introduced in [2].

A. Input of the network

As mentioned above, processing raw LiDAR point clouds is computationally expensive. Indeed, these 3D point clouds are stored as unorganized lists of (x, y, z) Cartesian coordinates. Therefore processing such data, or turning them into voxels involve heavy memory costs. Modern LiDAR sensors often acquire 3D points, with a sampling pattern with a few number of scan lines and quasi uniform angular steps between samples, from which we can build a dense image [2], the so-called range-image. Indeed each point is defined by two angles and a depth, (θ, ϕ, d) respectively, with steps

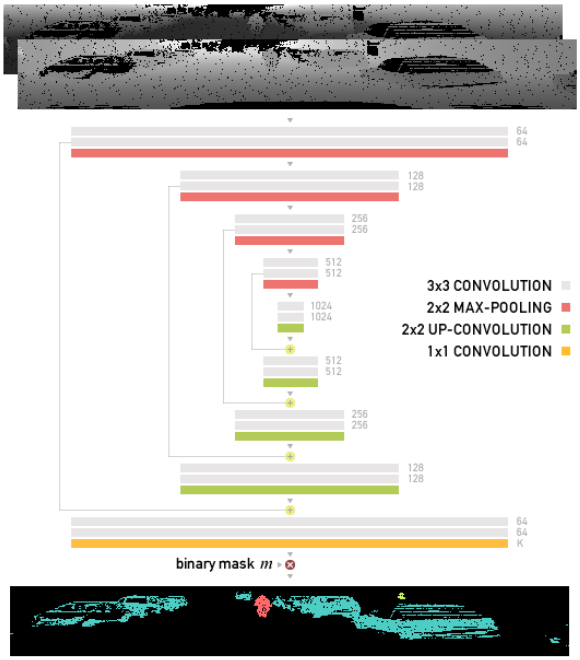

Fig. 2. RIU-Net: U-Net architecture adapted to point cloud semantic segmentation with the depth and elevation channels input (top) and the output segmented image (bottom).

of $(\Delta\theta, \Delta\phi)$ between two consecutive positions. Each point p of the LiDAR point cloud can be mapped to the coordinates (x, y) with $x = \lfloor \frac{\theta}{\Delta\theta} \rfloor, y = \lfloor \frac{\phi}{\Delta\phi} \rfloor$ of a 2D range-image, where each channel represents a modality of the measured point. Note that such processing is not required whenever the raw LiDAR data (with beam number) are available. For the rest of this work, we use a range-image named u of 512×64px with two channels: the depth towards the sensor and the elevation. In perfect conditions, the resulting image is completely dense, without any missing data. However, due to the nature of the acquisition, some measurements are considered invalid by the sensor and they lead to empty pixels (no-data). This happens when the laser beam is highly deviated (*e.g.* when going through a transparent material) or when it does not create any echo (*e.g.* when the beam points in the sky direction). We propose to identify such pixels using a binary mask m equal to 0 for empty pixels and to 1 otherwise. The analysis of multi-echo LiDAR scans is subject to future work.

B. Architecture

The U-net architecture [17] is an encoder-decoder. As illustrated in Figure 2, the first half consists in the repeated application of two 3×3 convolutions followed by a rectified linear unit (ReLU) and a 2×2 max-pooling layer that down-samples the input by a factor 2. Each time a downsampling is done, the number of features is doubled. The second half of the network consists in upsampling blocks where the input is upsampled using 2×2 up-convolutions. Then, concatenation is done between the upsampled feature map and the corresponding feature map of the first half. This allows the network to capture global details while keeping fine details. After that, two 3×3 convolutions are applied followed by a ReLU. This block is repeated until the output of the network matches the dimension of the input. Finally, the last layer consists in a 1x1 convolution that outputs as many features as the wanted number of possible labels K 1-hot encoded.

C. Loss function

The loss function of the semantic segmentation network is defined as the cross-entropy of the softmax of the output of the network. The softmax is defined pixel-wise for each label k as follows:

$$p_k(x) = \frac{\exp(a_k(x))}{\sum_{k'=0}^{K} \exp(a_{k'}(x))}$$

where $a_k(x)$ is the activation for feature k at the pixel position x. After that, we define $l(x)$ the groundtruth label of the x pixel. We then compute the cross-correlation as follows:

$$E = \sum_{x \in \Omega} \mathbb{1}_{\{m(x)>0\}} w(x) \log(p_{l(x)}(x))$$

where Ω is the domain of definition of u, $m(x) > 0$ are the valid pixels and $w(x)$ is a weighting function introduced to give more importance to pixels that are close to a separation between two labels, as defined in [17].

D. Training

We train the network with the Adam stochastic gradient optimizer and a learning rate set to 0.001. We also use batch normalization with a momentum of 0.99 to ensure good convergence of the model. Finally, the batch size is set to 8 and the training is stopped after 10 epochs.

IV. Experiments

To test RIU-Net, we follow the experimental setup of the SqueezeSeg approach [19]. They provide range-images with segmentation labels exported from the 3D object detection challenge of the KITTI dataset [7]. The complete dataset is split into a training set of 8678 images (80%) and a validation set of 2170 images (20%).

Figure 1 shows a segmentation result of RIU-Net and the groundtruth both on the range-image (top) and in 3D (bottom). The segmentation in 3D is obtained by labelling the raw point cloud according to the result on the range-image. More results are shown in Figure 3. They all highlight how visually similar the results obtained with RIU-Net and the groundtruth are.

Similarly to [19] and [20], we use the intersection-over-union metric to evaluate RIU-Net and we compare it with the state-of-the-art:

$$IoU_l = \frac{|\rho_l \bigcap G_l|}{|\rho_l \bigcup G_l|}$$

where ρ_l and G_l denote the predicted and groundtruth sets of points that belongs to label l respectively.

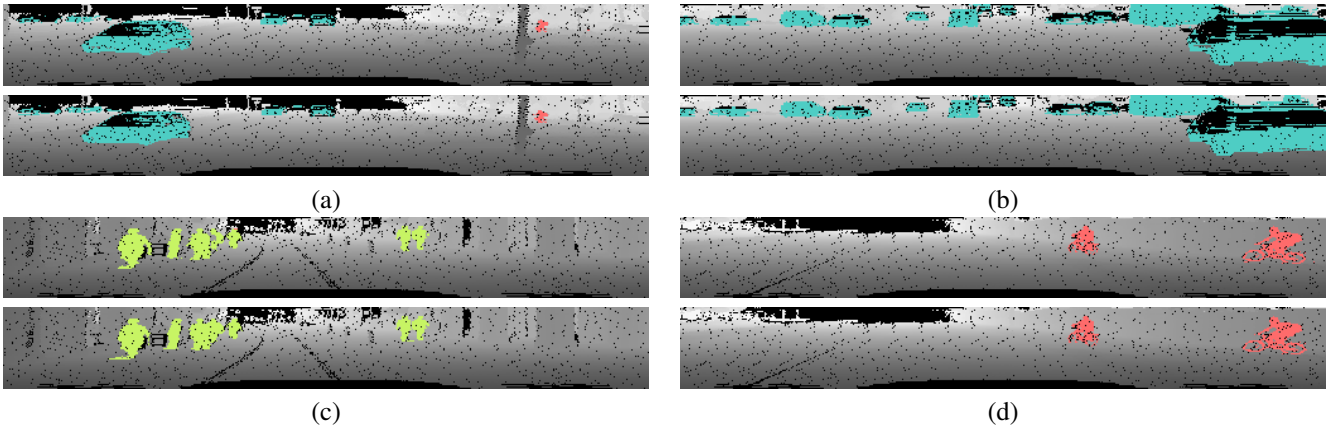

Fig. 3. Results of the semantic segmentation of the proposed method (top) and groundtruth (bottom). Labels are associated to colors as follows: blue for the cars, red for the cyclists and lime for the pedestrians.

Table I presents the results obtained for the segmentation of cars, cyclists and pedestrians, with SqueezeSeg [19], SqueezeSegv2 [20] and PointSeg [18] compared to RIU-Net. The scores of state-of-the-art methods are taken from the corresponding papers, using the same training conditions. We can see that the results of RIU-Net are comparable to the one of the state-of-the-art, despite the architecture being very simple. Indeed, our method achieves better average IoU score compared to the SqueezeSeg [19] architecture. Moreover, it outperforms the model PointSeg [18] for both Cars and Cyclists classes. We believe that the increased number of parameters of our model compensate with the fact that the architecture had not been specifically designed for LiDAR point cloud segmentation in sensor topology, contrary to the other methods. The method SqueezeSegV2 is built on the SqueezeSeg architecture, while proposing several modifications (content aggretation modules and fire layers) driven by the specificity of the data. Therefore, it is reasonable to think that comparable results could be achieved with our model by applying the same modifications, however our goal is to keep the architecture simple and as generic as possible. Visual results of our method against the ground truth are displayed in Figure 3. Finally, we advocate that the proposed model can operate with a frame-rate of 90 frames per second on a single GPU, which is comparable, if not faster, to state-of-the-art methods.

TABLE I

COMPARISON (IOUS, %) OF OUR APPROACH WITH THE STATE-OF-THE-ART FOR THE SEMANTIC SEGMENTATION OF THE KITTI DATASET.

	Cars	Pedestrians	Cyclists	Average
PointSeg [18]	67.4	19.2	32.7	39.8
SqueezeSeg [19]	64.6	21.8	25.1	37.2
SqueezeSegv2 [20]	**73.2**	**27.8**	**33.6**	**44.9**
RIU-Net	65.1	19.1	28.3	37.5

V. CONCLUSION

In this paper, we have shown that applying a U-net architecture on LiDAR data is comptetitive with the state-of-the-art for point cloud semantic segmentation. The U-net architectures is fed with range-image representations of a 3D point cloud. The resulting method, called RIU-Net, is simple and fast, but yet provides results that are comparable to state-of-the-art methods, while proposing an efficient bridge between image processing and 3D point cloud processing. The current method relies on a cross-entropy loss function. We plan to investigate other functions such as the focal loss used in [20] and to study possible spatial regularization schemes. Finally fusion of LiDAR and optical data would probably enable reaching a higher level of accuracy.

VI. ACKNOWLEDGEMENT

This project has also received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 777826.

REFERENCES

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.

[2] P. Biasutti, J-F. Aujol, M. Brédif, and A. Bugeau. Range-Image: Incorporating sensor topology for LiDAR point cloud processing. *Photogrammetric Engineering & Remote Sensing*, 84(6):367–375, 2018.

[3] A. Briot, P. Viswanath, and S. Yogamani. Analysis of efficient CNN design techniques for semantic segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 663–672, 2018.

[4] L-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018.

[5] L-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and A. Hartwig. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proc. of ECCV*, pages 801–808, 2018.

[6] C. Feng, Y. Taguchi, and V. R. Kamat. Fast plane extraction in organized point clouds using agglomerative hierarchical clustering. In *IEEE International Conference on Robotics and Automation*, pages 6218–6225, 2014.

[7] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[8] M. Himmelsbach, A. Mueller, T. Lüttel, and H-J. Wünsche. LIDAR-based 3D object perception. In *Proceedings of international workshop on Cognition for Technical Systems*, pages 1–7, 2008.

[9] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018.

[10] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. Squeezenet: AlexNet-level accuracy with 50x fewer parameters and ¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[11] L. Landrieu and M. Boussaha. Point cloud oversegmentation with graph-structured deep metric learning. *arXiv preprint axXiv:1904.02113*, 2019.

[12] L. Landrieu and M. Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 4558–4567, 2018.

[13] B. Li. 3D fully convolutional network for vehicle detection in point cloud. In *IEEE Trans. on Intelligent Robots and Systems*, pages 1513–1518, 2017.

[14] G. Lin, A. Milan, C. Shen, and I. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1925–1934, 2017.

[15] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[16] C. R Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 652–660, 2017.

[17] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

[18] Y. Wang, T. Shi, P. Yun, L. Tai, and M. Liu. Pointseg: Real-time semantic segmentation based on 3D LiDAR point cloud. *arXiv preprint arXiv:1807.06288*, 2018.

[19] B. Wu, A. Wan, X. Yue, and K. Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3D LiDAR point cloud. In *IEEE International Conference on Robotics and Automation*, pages 1887–1893, 2018.

[20] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer. SqueezesegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud. *arXiv preprint arXiv:1809.08495*, 2018.

[21] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. Icnet for real-time semantic segmentation on high-resolution images. In *Proc. of ECCV*, pages 405–420, 2018.

[22] Y. Zhou and O. Tuzel. Voxelnet: End-to-end learning for point cloud based 3D object detection. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.