Vehicle Detection and Localization using 3D LIDAR Point Cloud and Image Semantic Segmentation

Rafael Barea[1], Carlos Pérez[1], Luis M. Bergasa[1], Elena López-Guillén[1], Eduardo Romera[1], Eduardo Molinos[1],
Manuel Ocaña[1], Joaquín López[2]

Abstract— **This paper presents a real-time approach to detect and localize surrounding vehicles in urban driving scenes. We propose a multimodal fusion framework that processes both 3D LIDAR point cloud and RGB image to obtain robust vehicle position and size in a Bird's Eye View (BEV). Semantic segmentation from RGB images is obtained using our efficient Convolutional Neural Network (CNN) architecture called ERFNet. Our proposal takes advantage of accurate depth information provided by LIDAR and detailed semantic information processed from a camera. The method has been tested using the KITTI object detection benchmark. Experiments show that our approach outperforms or is on par with other state-of-the-art proposals but our CNN was trained in another dataset, showing a good generalization capability to any domain, a key point for autonomous driving.**

I. Introduction

Perception in dynamic environment plays a pivotal role to autonomous driving. 3D object detection and localization has become an increasing research topic because it supposes an important challenge for vehicles, pedestrians and cyclists recognition on roads. This data can be used to generate objects' trajectories and to predict their motion. Based on this information, different high-level driving behaviors can be implemented, such as: avoiding collision, overtaking others vehicles, stopping on crosswalks, etc. Nowadays, modern self-driving vehicles are equipped with multiple and high-precision sensors such as cameras and LIDAR.

LIDAR-based detection methods measure the distances to several points in the surroundings and create 3D point clouds [1]. They have the advantage of getting accurate depth information and obtaining robust results in location, independently of the environment lighting conditions. Their main drawbacks are sparsity of data, their price and their integration in commercial vehicles, where body car aesthetic plays an important role. On the other hand, camera-based methods provide much more detailed semantic information [2]. However, their performance degrade in scenes with difficult lighting conditions (sun-rise, night-time, etc.) and with the distance. On the other hand, further processing steps are required to obtain objects' ground positions (projection of detection results from images to ground). LIDAR and cameras should be used complementary to achieve higher performance and safety systems that compensate drawbacks in one modality [3].

In recent years, Convolutional Neural Networks (CNN) have achieved great success in object detection and recognition tasks achieving the top ranked results on public benchmarks as KITTI [4]. Most of them are focused on 2D detection and recognition in images [5] [6]. Some proposals tackle the 3D object detection in images doing 2D detection and 3D pose estimation [7] [8]. Some few approaches use 3D [9] or 2D [3] object detection in point cloud and the last approaches exploit multiple modalities of data [3] [10]. All of these supervised proposals present results based on KITTI dataset splitting data between training and validation set.

This paper presents a method to fuse 3D LIDAR point cloud with image semantic segmentation, obtained through a RGB-based CNN, to detect vehicles in images and localize them in a Bird's Eye View (BEV) point cloud projection. Our goal is to take advantage of the complementarity of these two sensors to achieve a high-precision and robust 3D object detection and location that permits driver-less navigation in urban environments. We propose an architecture that obtains semantic information from RGB images through a CNN, and projects it over a 3D point cloud, obtained from a LIDAR, reaching a coloring point cloud segmentation. The proposed CNN has been designed to get robust segmentation in unseen domains and to maximize its performance for real-time operation. Information from the two sensors are fused to detect 3D vehicle models (pose and size) in a BEV.

We have tested our proposal on the KITTI object detection benchmark [4]. Results are evaluated based on the average precision (AP) of the vehicles detection on the images as well as their localization accuracy on the ground plane. Experiments show that our proposal outperforms or is on par with other state-of-the-art results in terms of AP but using a CNN trained in another dataset, showing the generalization capability of our method, a key point for real autonomous navigation applications. Besides, we have studied pose and size errors of the detected vehicles showing that our estimations are good enough for autonomous driving.

*This work has been funded in part from the Spanish MINECO/FEDER through the SmartElderlyCar project (TRA2015-70501-C2-1-R, TRA2015-70501-C2-2-R), the Spanish DGT through the SERMON project (SPIP2017-02305) and from the RoboCity2030-III-CM project (S2013/MIT-2748), funded by Programas de actividades I+D (CAM) and cofunded by EU Structural Funds.

[1] Rafael Barea, Carlos Pérez, Luis M. Bergasa, Elena López-Guillén, Eduardo Romera, Eduardo Molinos and Manuel Ocaña are with the Electronics Department, University of Alcalá (UAH), Spain {rafael.barea, luism.bergasa, elena.lopezg, manuel.ocanna}@uah.es,{carlos.perzrivas, eduardo.romera, eduardo.molinos,}@edu.uah.es

[2] Joaquín López is with the Department of Systems Engineering and Automation, University of Vigo, Pontevedra, Spain joaquin@uvigo.es

II. Related Works

This section briefly reviews most important works of the literature on object detection using LIDAR point cloud, image and fusion of them.

A. LIDAR-based Object Detection

Most existing methods encode 3D point cloud with voxel grid representation and use feature detectors for classification. Rusu et al. in [11] developed the Point Feature Histograms (PFH) and Viewpoint Feature Histograms (VFH) that use the geometrical structure of neighboring points to compute the features and obtain a descriptor. Some works use SVM classifiers on 3D clusters encoded with geometry features, such as Vote3D [12]. In [13] authors investigate volumetric and multi-view representation for 3D object classification. Recently, some works propose to improve feature representation by using 3D convolutional networks [9] [14]. In VeloFCN [15] point cloud is projected to the front view, a convolutional network is applied on the 2D point map and 3D boxes are predicted from the convolutional feature map.

Most commonly used methods discretize point clouds into a 3D grid. Since the detection is performed in 3D space directly, object size variations are limited to real size variation of the objects, and positions of the detected objects can be obtained directly. However, the point density may vary as a function of the distance and classifiers must work with both dense and sparse data [3]. In practice, classifiers usually work well in short-distance (dense data) and hardly work properly in long-distance (sparse data). Our proposal incorporates semantic information to the point cloud to improve 3D classification specially at long-distance.

B. Image-based Object Detection

Most of these approaches employ detectors to do 2D detection and then do 3D pose estimation. Some of them use monocular images to generate 3D object proposals [7] and others use stereo images for accurate objects detection [16]. Hough Transform and 3D SURF have also been used for robust 3D classification [17]. 3DVP [18] introduces ACF detectors to estimate 3D voxels and 3DOP [8] reconstructs a depth image from stereo images and uses energy minimization to generate 3D proposals.

In the last years, CNN-based object detection has played an important role in image classification. Most popular methods use Fast R-CNN [19] for vehicle detection, which has a two stage detection framework. In the first stage, some region proposals that are likely to contain objects are generated. In the second stage, the CNN is applied on the region proposals to classify the object and refine its locations. In 3DOP method the 3D box proposals are fed to an R-CNN pipeline for vehicle recognition. Mono3D [7] shares the same pipeline with 3DOP but it generates 3D proposals from monocular images.

The main drawback of these methods is that they usually rely on accurate depth estimation but, in practice, camera models are not so accurate. Our proposal incorporates LIDAR point cloud to improve 3D localization.

C. Multimodal fusion

Only a few works exploit multimodal fusion in the context of vehicle detection. However, the fusion of multiple data can provide complementary information and increase the accuracy of the decision making process in autonomous driving [20]. The most common fusion strategy consists of merging both LIDAR and images. [21] describes a framework for multimodal information fusion for urban scene understanding.

Recently, multi-view networks have been proposed for 3D object detection in the field of autonomous driving. [22] presents a multi-modal sensor registration for vehicle perception via deep neural networks and [23] describes a vehicle detection system based on LIDAR and camera fusion. [20] introduces a Multi-View object detection network (MV3D) that takes both LIDAR point cloud and RGB image as input and predicts oriented 3D bounding boxes.

Our work is based on a multimodal fusion but differs from previous works in the applied fusion method, because it incorporates visual semantic information to 3D LIDAR point cloud and use a double 2D/3D validation check to improve 3D vehicle detection.

III. Architecture for Vehicle Detection

We propose an architecture based on a multimodal fusion from two complement sensors as are a 3D LIDAR point cloud and a RGB image. Semantic segmentation is obtained from the RGB image using a CNN developed by the authors (ERFNet) [24]. Through the labeling of categories of the image at the pixel-level, the minimum rectangle hulls of the vehicle blobs (blue color) in the segmented image are taken as the 2D bounding boxes proposals. On the other hand, 2D semantic information is projected in the 3D LIDAR point cloud obtaining a 3D colored cloud. Based on the 3D semantic point cloud, 3D bounding boxes proposals are done applying clustering. These proposals are projected in the front view for easy merging. Proposals from the image and from the LIDAR are fused obtaining validated 2D boxes for vehicles in the image. These boxes are projected back in the 3D ground plane and are seen in the BEV. Fig.1 shows an overview diagram of the explained architecture.

A. 2D Vehicle Detection from RGB Images

Differently to most of the approaches of the state of the art, we base our 2D vehicle detection in semantic segmentation obtained from the RGB image using our Efficient Residual Factorized ConvNet for Real-time Semantic Segmentation (ERFNet) and a complete data augmentation strategy. This is a deep architecture able to run in real-time while providing accurate semantic segmentation. The core of our architecture is a novel layer that uses residual connections and factorized convolutions in order to remain efficient while retaining remarkable accuracy [24]. We use this approach for two main reasons. The first one is that semantic segmentation provides a global understanding of the traffic scene and can be used to add semantic information to the LIDAR point cloud and to detect main objects in the scene (vehicles in our case) in

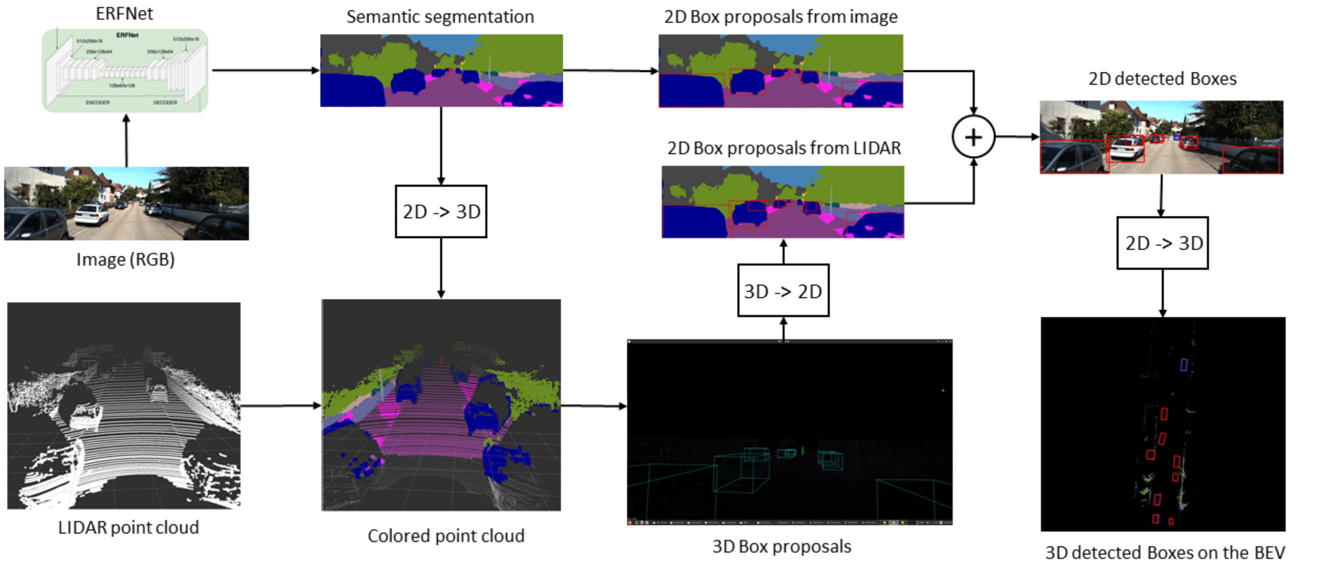


Fig. 1: Overview diagram of our architecture proposal for vehicle detection and localization.

a direct way. The second one is that our goal is to achieve robustness in any domain. This is the reason because our ERFNet is trained on Cityscapes [25] with 19 classes instead of being trained in the own training set of KITTI. Fig.1 shows our semantic segmentation where cars are detected in dark blue, pedestrians in red, road in magenta, sky in light blue, etc.

Since our CNN provides semantic segmentation of the image at the pixel-level, it is easy to detect different objects in the scene by using color codification and connectivity of the pixels. Imposing some geometric restrictions about size and form, the minimum rectangle hulls of the vehicle blobs (blue color) in the segmented image are taken as the 2D bounding boxes proposals. Some results can be seen in Fig.1.

On the other hand, 2D semantic information is related to 3D LIDAR coordinates according to the intrinsic calibration parameters of the camera (P), and the translation (T) and rotation (R) matrices of the camera with respect to the 3D LIDAR position. In this way it is possible to estimate the 3D position of a pixel in the world coordinates (x_{3D}) from its projection in the image in pixels (x_{2D}) and the opposite through equations 1 and 2.

$$x_{3D} = (RT)^{-1}P^{-1}x_{2D} \tag{1}$$

$$x_{2D} = PRTx_{3D} \tag{2}$$

B. 3D Vehicle Detection from the LIDAR point cloud

Taken the semantic information obtained by the CNN (RGB) and the LIDAR point cloud (PointXYZ) it is possible to obtain a 3D colored point cloud (PointXYZRGB) where different objects in the scene are classified by color (see Fig.1). To do that, 3D point cloud is projected to 2D semantic image applying equation 2. In this way, each point of the point cloud is colored according to the color of the object class on which it is projected. When semantic information has been added to the point cloud, we proceed with classification stage.

Classification is carried out by color filtering because points with the same color belong to the same class. However, there can be some cases where different objects of a class are connected (see cars in Fig.1) and additional processing is necessary to detect them in a separate way. A clustering of the 3D point cloud positions with the same color allows to detect the different objects in the scene for each class. Our clustering algorithm is based on euclidean distance and was proposed by [26]. This algorithm uses a Kd-tree structure for finding the nearest neighbors. We have modified this algorithm to take into account the difference between horizontal and vertical angular resolution in LIDAR data. This difference is very important, especially when the distance is large. Our algorithm adapts euclidean resolution as a function of the object distance and the vertical scan.

Once the different objects have been detected, we assign a geometric model to each of them. We opted to fit each cluster into the 3D bounding box that best suits the shape of the cluster. Length of the boxes are discretized to 5 different values and height is fixed to 1.6m. This method presented many problems due to bad observations and occlusions: shadows caused by other obstacles or by itself, occlusions of the rear part of the vehicles by the front parts, etc., producing errors mainly in the orientation of the boxes. In order to improve our object estimation pose (position and orientation), we project orthogonally the 3D point cloud to the 2D ground plane (z=0) and fit a 2D box to each object. For the points inside the box we apply the Hough Transform to get the main directions of the projected points that correspond with the correct orientation of the box. After that, the 3D bounding box is fit with this orientation angle. Resulting 3D box proposals can be seen in Fig.1.

C. Fusion from LIDAR and Image proposals

Once the vehicles have been detected separately in 2D and 3D through the 2D and 3D box proposals, they are merged

TABLE I: 3D localization performance: Average Precision (AP) in % of 3D boxes on KITTI validation set

Method	**Method**	**IoU=0.5**			**IoU=0.7**		
		Easy	**Moderate**	**Hard**	**Easy**	**Moderate**	**Hard**
Mono3D [7]	Mono	30.5	22.39	19.16	5.22	5.19	4.13
3DOP [8]	Stereo	55.04	41.25	34.55	12.63	9.49	7.59
VeloFCN [15]	LIDAR	79.68	63.82	62.80	40.14	32.08	30.47
MV3D [20]	LIDAR+mono	96.52	89.56	88.94	86.55	78.10	76.67
Ours	LIDAR+mono	**79.77**	**65.76**	**63.14**	**57.24**	**43.08**	**39.00**

to validate common detections and to complement detections carried out only for one of the sensors. The idea is to use complementarity of the two sensors to improve detection. To fuse data in a correct way, 3D box proposals are projected to the image plane obtaining some red boxes (see Fig.1) that can be easily matched with the red boxes obtained from the semantic image. If a proposal overlaps in the two domains and follows some geometric restrictions, it is validated and the LIDAR box is passed to the output image as a detected vehicle. On the other hand, if a proposal appears only in one domain it is validated depending of the sensor and the distance where it was found.

Vehicle detection with point cloud works quite well and with high precision for objects located at short distance ($< 50m$). For long distance, LIDAR data is very sparse and object detection is very difficult. In addition, to strengthen detection a minimum threshold size of 10 colored points were included. Long distance gap is covered by our semantic proposals, which can easily detect vehicles at long distance ($> 50m$) although localization precision quadratically decreases with this parameter. Finally, validated 2D boxes are projected back to the ground plane, as it can be seen in the BEV image of Fig.1. In this case, 8 cars were correctly detected by the LIDAR and the camera (red boxes) and one additional car located at 65 m was detected by the image process (blue box).

IV. Experimental Results

A. Experiments

We evaluate our vehicle detection and localization proposal on the challenging KITTI object detection benchmark [4]. The dataset provides 7,481 images for training with ground truth annotations and 7,518 images for online testing without ground truth. As the online testing only evaluates 2D detection, we conduct our evaluation on the training set. To evaluate localization, we use point cloud in the range of [0, 70] x [-40, 40] meters. Ground truth labels are transformed to the LIDAR coordinates with the transformation matrices provided by the dataset.

For implementation, we use our ERFNet pre-trained on the Imagenet dataset and trained on Cityscapes with 19 classes. A complete set of data augmentation is carried out to get robust semantic segmentation in any domain. Evaluation is carried out on the KITTI training set for the vehicle class, taken into account that these images have not been seen before for our CNN. Main research efforts are being put on enlarging deep architectures to achieve accuracy boosts in KITTI (usually training data is split into a training and validation set), forgetting that these algorithms must be deployed in a real vehicle with images that were not seen during training. In this paper one of our goals is achieving robustness in any domain. A deeper explanation of the domain adaptation capability of our proposal can be found in [27].

About the CNN training setup, we train all models in the same conditions using Adam optimization with an initial Learning Rate (LR) of 1e-4 and Weight Decay (WD) of 2e-4, decreasing LR exponentially until cross-entropy loss converges. For more details about optimal training setup or architecture details please refer to ERFNet papers [28] [24].

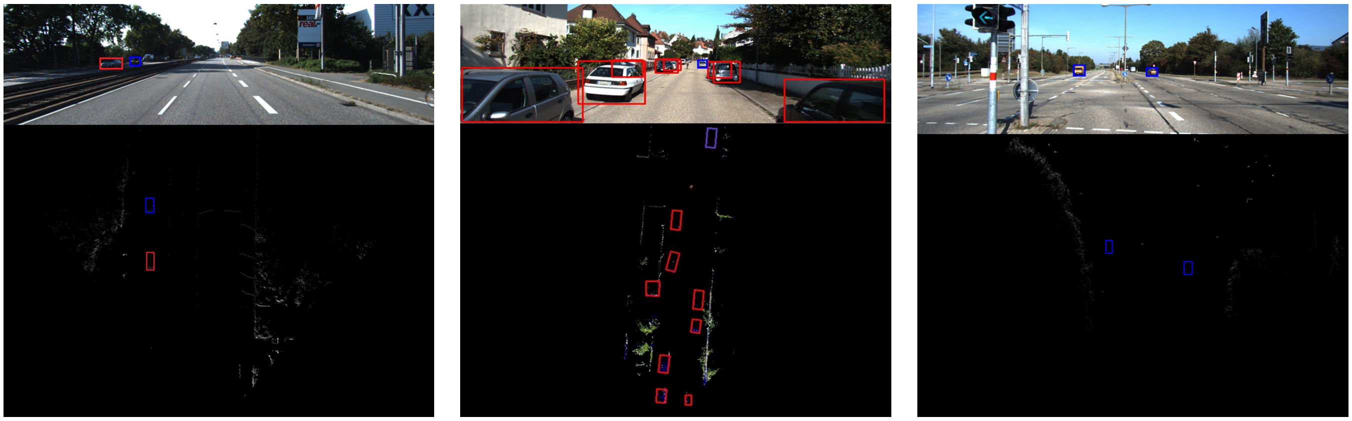
B. Metrics

We validate our proposal in both the image space and the world space using Average Precision (AP) in the following metrics:

1) **Bounding box overlap on the image plane**. This is the original metric of the KITTI benchmark. The 3D LIDAR bounding box proposal is projected to the image plane and the minimum rectangle hull of the projection is taken as the 2D bounding boxes after a fusion with the 2D box proposals taken from the semantic image. Following the KITTI convention, Intersection over Union (IoU) threshold is set to 0.7 for 2D boxes. This metric evaluate vehicle detection.
2) **Bounding box overlap on the ground plane**. The 3D bounding box detection, obtained by projecting back the 2D detected box to the LIDAR coordinates, is projected onto the 2D ground plane orthogonally. A detection is accepted if the overlap area IoU with the ground truth is larger than a certain threshold of 0.5 and 0.7. Since coordinates on BEV images represent ground coordinates positions, vehicle localization performance is evaluated with the BEV bounding box. This metric reflects autonomous driving demand, in which vertical localization is less important than the horizontal.

C. Baseline for comparison

As this work aims at 2D vehicle detection and 3D vehicle localization, we compare our approach to representative LIDAR-based methods as VeloFCN [15] and Vote3D [12], representative image-based methods as 3DOP [8] and Mono3D [7], as well as a reference of the multimodal methods (LIDAR + image) as is the MV3D [20]. For 3D localization evaluation, we compare with Mono3D [7], 3DOP [8], VeloFCN [15] and MV3D [20] since they provide results on the validation set for a IoU of 0.5 and 0.7 on BEV images.



(a) *Image 000004* **(b)** *Image 000025* **(c)** *Image 000012*

Fig. 2: Examples of Vehicle Detection and Localization Results: 2D boxes detection in images and 3D boxes projected to the BEV

For Vote 3D, which have no results publicly available, we only do comparison on 2D detection.

D. *Performance of 3D Vehicle Localization*

Table I shows AP on KITTI validation set using bounding box overlap on the ground plane for a IoU threshold of 0.5 and 0.7. Since LIDAR sensors obtain distance measurements directly, the LIDAR-based method (VeloFCN) performs better than image-based methods (Mono3D, 3DOP). Best results are obtained for the fusion proposals (MV3D and our). Our method outperforms LIDAR-based and image-based proposals for the IoU=0.5 and IoU=0.7 and for the easy, moderate and hard regime. By combining with visual semantic, our approach is further improved. MV3D performs much better than our proposal but our method is the only one that has not been trained with KITTI images and validation is carried out over the whole training images and not in a subset of it. We visualize the localization results of some examples in Fig.2.

On the other hand, IoU is not well suited for evaluation of vehicle localization [3] and they propose to evaluate center offset and size errors separately as evaluation criteria. For the center offset error the parameters are (x_{error}, y_{error}, z_{error}), where x_{error} is the offset in the heading direction of the vehicle, y_{error} is the offset in the direction orthogonal to the heading, and z_{error} is the vertical (height) offset. For the size error the parameters are width (orthogonal to the heading direction, length (parallel to heading direction) and height (vertical direction) respectively.

We evaluate our method following this criteria and results are shown in Fig.3 for an IoU threshold of 0.7. 99.85% of the detection results are localized within an error of 40 cm for (x_{error}, y_{error}) and 50% of the detections have an error lower than 20 cm for the same parameters. With the exception of some single outliers that go above a localization error of 1 m, results can be consider good enough. The size error of the bounding box also shows good results but they are worse than localization error. This means that our method to calculate box orientation can be improved. An error of less than 20 cm for width, less than 40 cm for height and less than 50 cm for length was accomplished for more than 50% of the detections. Height estimation shows a larger error than width estimation due to only a discrete value is allowed for this parameter. The largest error is obtained for length due to this parameter has only some discrete values and it is difficult to estimate from front view images where many times rear parts of the vehicles are not seen or are occluded.

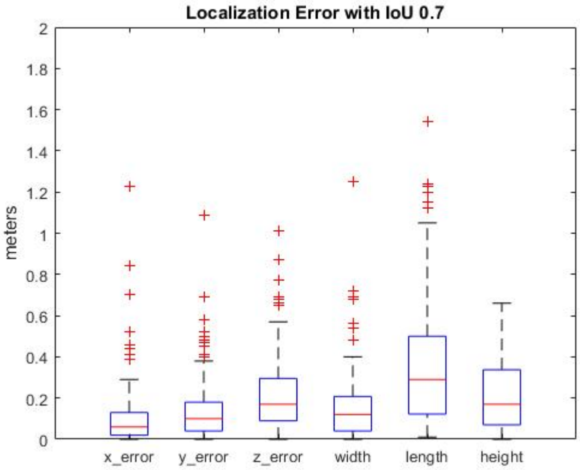


Fig. 3: Localization error and size error of bonding box

E. *Performance of 2D Vehicle Detection*

Table II shows 2D detection performance for the car class for an IoU=0.7 on KITTI test set except for our approach, where training set is used due to this last set has not been used for training this CNN. Image-based methods (Mono3D, 3DOP) perform better than LIDAR-based methods (VeloFCN, Vote3D) in terms of 2D vehicle detection. This is due to image-based methods directly optimize 2D boxes while LIDAR-based methods optimize 3D boxes. Fusion proposals (MV3D and ours) get intermediate results because they optimize both 2D/3D boxes. Our method outperforms LIDAR-based for the easy, moderate and hard regime but gives worse results than image-based for the three regime. However, this is not representative for autonomous driving applications where 3D vehicle detection and location and not 2D vehicle detection is the key parameter. Our proposal shows comparable results than MV3D on AP performance and a good capability to generalize to diverse domains because Cityscapes and KITTI datasets are quite different.

TABLE II: 2D Detection performance: Average Precision (AP) in % for car class on KITTY test set excepts for our proposal where training set is used.

Method	Data	Easy	Moderate	Hard
Mono3D [7]	Mono	92.33	88.66	78.96
3DOP [8]	Stereo	93.04	88.64	79.10
VeloFCN [15]	LIDAR	71.06	53.59	46.92
Vote3D [12]	LIDAR	56.80	47.99	42.57
MV3D [20]	LIDAR+Mono	89.11	87.67	79.54
Ours	LIDAR+Mono	**90.45**	**78.28**	**73.20**

V. Conclusions and Future Works

In this paper we proposed a method to fuse 3D LIDAR point cloud with image semantic segmentation, obtained through our RGB-based CNN called ERFNet, to detect vehicles in images and localize them in a Bird's Eye View (BEV) point cloud projection. Our method takes advantage of both LIDAR point cloud and images. Our approach outperforms existing LIDAR-based and image-based methods for 3D vehicle localization and is on par with other state-of-the-art multimodal proposals for 2D vehicle detection on KITTI benchmark. Besides, our CNN was trained in another dataset, showing a good generalization capability to any domain. Additionally, we studied pose and size errors of the detected vehicles showing that, although our box orientation estimation is still suboptimal, results are promising for autonomous driving.

As future work we plan to use CNN for estimating box poses and size in the ground plane and integrate the method in our open source electric vehicle prototype to be evaluated in real conditions.

References

[1] Y. Ye, L. Fu, and B. Li, "Object detection and tracking using multi-layer laser for autonomous urban driving," in *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*, pp. 259–264, IEEE, 2016.

[2] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.

[3] S.-L. Yu, T. Westfechtel, R. Hamada, K. Ohno, and S. Tadokoro, "Vehicle detection and localization on bird's eye view elevation images using convolutional neural network," in *Safety, Security and Rescue Robotics (SSRR), 2017 IEEE 17th International Symposium on*, pp. 102–109, IEEE, 2017.

[4] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 3354–3361, IEEE, 2012.

[5] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[6] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[7] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3d object detection for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2147–2156, 2016.

[8] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *Advances in Neural Information Processing Systems*, pp. 424–432, 2015.

[9] B. Li, "3d fully convolutional network for vehicle detection in point cloud," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pp. 1513–1518, IEEE, 2017.

[10] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. Waslander, "Joint 3d proposal generation and object detection from view aggregation," *arXiv preprint arXiv:1712.02294*, 2017.

[11] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 3212–3217, IEEE, 2009.

[12] D. Z. Wang and I. Posner, "Voting for voting in online point cloud object detection.," in *Robotics: Science and Systems*, vol. 1, p. 5, 2015.

[13] V. Hegde and R. Zadeh, "Fusionnet: 3d object classification using multiple data representations," *arXiv preprint arXiv:1607.05695*, 2016.

[14] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proceedings of the IEEE international conference on computer vision*, pp. 945–953, 2015.

[15] B. Li, T. Zhang, and T. Xia, "Vehicle detection from 3d lidar using fully convolutional network," *arXiv preprint arXiv:1608.07916*, 2016.

[16] X. Chen, K. Kundu, Y. Zhu, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals using stereo imagery for accurate object class detection," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[17] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool, "Hough transform and 3d surf for robust three dimensional classification," in *European Conference on Computer Vision*, pp. 589–602, Springer, 2010.

[18] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Data-driven 3d voxel patterns for object category recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1903–1911, 2015.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[20] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *IEEE CVPR*, vol. 1, p. 3, 2017.

[21] P. Xu, F. Davoine, J.-B. Bordes, H. Zhao, and T. Denœux, "Multimodal information fusion for urban scene understanding," *Machine Vision and Applications*, vol. 27, no. 3, pp. 331–349, 2016.

[22] M. Giering, V. Venugopalan, and K. Reddy, "Multi-modal sensor registration for vehicle perception via deep neural networks," in *High Performance Extreme Computing Conference (HPEC), 2015 IEEE*, pp. 1–6, IEEE, 2015.

[23] F. Zhang, D. Clarke, and A. Knoll, "Vehicle detection based on lidar and camera fusion," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pp. 1620–1625, IEEE, 2014.

[24] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Erfnet: Efficient residual factorized convnet for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 263–272, 2018.

[25] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213–3223, 2016.

[26] R. B. Rusu, "Semantic 3d object maps for everyday manipulation in human living environments," *KI-Künstliche Intelligenz*, vol. 24, no. 4, pp. 345–348, 2010.

[27] E. Romera, L. M. Bergasa, J. M. Alvarez, and M. Trivedi, "Train here, deploy there: Robust segmentation in unseen domains," in *Proceedings of the IEEE conference on Intelligent Vehicles Symposium, p. to appear, IEEE ITS*, 2018.

[28] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Efficient convnet for real-time semantic segmentation," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*, pp. 1789–1794, IEEE, 2017.