An Experimental Study on 3D Person Localization in Traffic Scenes

Joram R. van der Sluis*,1, Ewoud A.I. Pool*,1 and Dariu M. Gavrila1

***Abstract*— This paper presents an experimental study on 3D person localization (i.e. pedestrians, cyclists) in traffic scenes, using monocular vision and LiDAR data. We first analyze the detection performance of two top-ranking methods (PointPillars and AVOD) on the KITTI benchmark, with respect to varying Intersection over Union (IoU) settings and the underlying parameters of 3D bounding box location, extent and orientation. Given that the KITTI dataset contains relatively few 3D person instances, we also consider the new EuroCity Persons 2.5D (ECP2.5D) dataset, which is one order of magnitude larger. We perform domain transfer experiments between the KITTI and ECP2.5D datasets, to examine how these datasets generalize with respect to each other.**

I. Introduction

According to a recent report of the World Health Organisation, about half of the 1.3 million people killed yearly in traffic worldwide involve Vulnerable Road Users (VRUs), i.e. pedestrians, cyclists and other riders. For much of the past two decades, vision was the dominant sensor modality for intelligent vehicles to detect VRU. Strong progress has been made on 2D image-based VRU detection facilitated by novel (deep learning) methods, faster processors and more data (including benchmarks, e.g. [1], [2], [3]). 3D localization from 2D detections can subsequently be achieved by back-projection, disparity computation [4], and/or association with radar targets. Vision-based VRU detection is meanwhile incorporated in active safety systems of various premium vehicles on the market.

Still, current active VRU safety systems are deployed in the context of driver assistance. With the advent of fully self-driving vehicles, performance needs to be significantly upped, as a driver is no longer available as a back-up. The LiDAR sensor is an attractive sensor for self-driving vehicles, stemming from its capabilities to directly and accurately measure distances and to deal with low-light environments. KITTI [5] meanwhile offers a 3D object detection benchmark, including one for pedestrians. The latter leader-board currently lists a 3D Average Precision (AP_{3D}) of 51% and around 40% for the *easy* and *all* targets, respectively (in contrast, the state-of-the-art in 2D object detection attains an Average Precision (AP) of 75% overall).

This paper presents an experimental study on 3D person localization (i.e. pedestrians, cyclists) in traffic scenes, using monocular vision and LiDAR data. We consider two 3D object detection methods, PointPillars [6] and AVOD [7], which are among the top performers on the KITTI benchmark, see fig. 1. We investigate the effect of the varying IoU setting on detection performance and quantify the various errors in terms of 3D bounding box location, extent and orientation. Given that the KITTI benchmark contains relatively few 3D person instances, we also perform experiments on a large subset of the new EuroCity Persons 2.5D (ECP2.5D) dataset [8]. Apart from being one order of magnitude larger than KITTI, ECP2.5D has advantages in terms of diversity (e.g. geographical coverage, time of day/season, weather conditions) and by being devoid of privacy-driven image blurring. This makes ECP2.5D also attractive when compared to other recent dataset additions, see table II. Finally, we perform domain transfer experiments between KITTI and ECP2.5D, to examine how these datasets relate to each other.

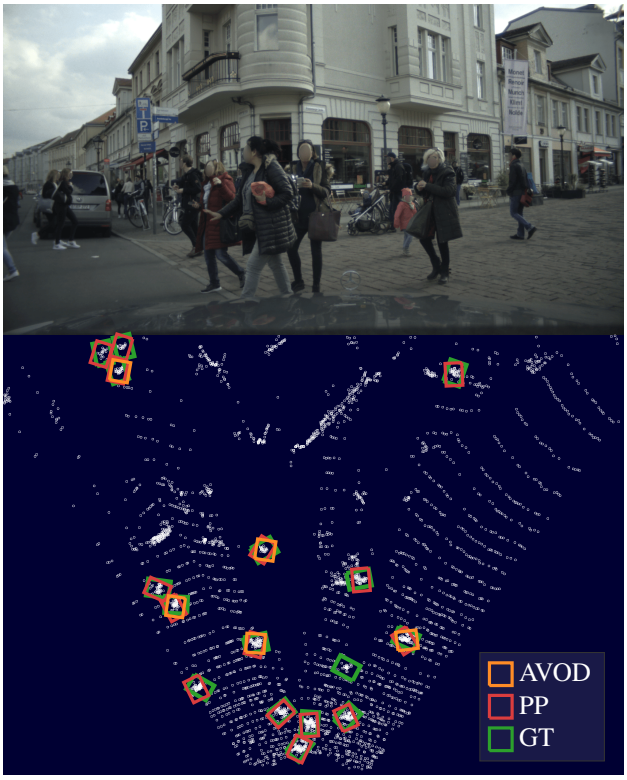


Fig. 1. An example of the predicted bounding boxes of PointPillars [6] (PP) and AVOD [7] on a scene from the EuroCity Persons 2.5D [8], along with the annotated ground truth (GT).

II. Previous Work

We focus our discussion on previous 3D object detection methods that use neural network architectures, as they are the current best performers in the various benchmarks.

One way to categorize this work is by sensor modality, i.e. either a single modality or a fusion of multiple modalities. The commonly used sensors used are (monocular) camera

∗) Authors contributed equally
1) Intelligent Vehicles group, TU Delft, The Netherlands

TABLE I

COMPARISON OF AVOD AND POINTPILLARS.

	AVOD	PointPillars
Modality	LiDAR + image	LiDAR
Stages	Two-stage	Single-stage
Bounding box regression	four corners, heights, orientation	3D center point, length, width, height, orientation

and LiDAR. However, the RGB-only methods (e.g. Shift R-CNN [9]) are generally outperformed by methods that instead use LiDAR information. These LiDAR-only networks map the point cloud to either a 2D or a 3D representation. Examples of 2D representations are Birds Eye View (used by *e.g.* HDNet [10]) and Range View (*e.g.* LaserNet [11]). Networks can also map the point cloud to 3D representations like Voxels (*e.g.* Voxelnet [12]), Pillars (*e.g.* PointPillars [6]), or Stixels (*e.g.* SCNet [13]).

Multi-sensor modality networks, or fusion networks, use both camera and LiDAR. Here, all the previously mentioned LiDAR mappings can be used to fuse with the camera data. How they are fused exactly falls into four categories. The first category is early fusion, where the modalities are concatenated before being passed into a neural network. An example of early fusion is MVX-Net PointFusion [14] where the pointcloud is projected onto a RGB-image and then concatenated. Secondly, deep fusion networks fuse the modalities after they have already been processed by a part of the network, for example PointFusion [15]. Here, the features from a PointNet [16] and a ResNet-50 are concatenated. With deep fusion, it is also possible to fuse the various modalities at multiple stages, as is done with AVOD [7]. Within such a deep fusion network, the performance is dependent on the feature encoder used [17]. Thirdly, late fusion takes the output of two or more independent networks and fuses the class probabilities [18]. Lastly, sequential fusion processes the sensor modalities in sequence. For example, Frustum PointNets [19] and Frustrum Convnet [20] use a 2D image detector to select frustums in a pointcloud, which is then processed separately.

Another way of categorizing previous 3D object detection methods is by the number of stages used by the network. Two-stage approaches utilize a Region Proposal Network (RPN) to generate bounding boxes which are individually evaluated (*e.g.* STD [21]). Single-state approaches instead evaluate predetermined bounding boxes (*e.g.* PointPainting [22]), also called anchor boxes.

Table I highlights the differences between PointPillars [6] and AVOD [7], two of the best performing LiDAR and fusion networks, respectively, with code available at the time of writing. These will be used later in the experiments.

In terms of existing datasets, one of the first 3D object detection benchmarks was an extension to KITTI [5], released in 2017, which contains around 9400 pedestrians (of which half in the publicly available training set). Since then, KITTI has become the de facto standard for 3D object detection. However, because of the relatively small dataset size, performances can differ a lot on the validation and test set. More recent dataset additions to KITTI are significantly larger and more diverse, see table II.

TABLE II

OVERVIEW OF TRAFFIC-RELATED 3D PERSONS DATASETS. A DASH DENOTES THAT THE INFORMATION COULD NOT BE DETERMINED.

Dataset	**Waymo** [23]	**nuScenes** [24]	**Argoverse** [25]	**Lyft** [26]	**KITTI** [5]	**ECP2.5D** [8]
# Countries	1	2	1	1	1	12
# Cities	2	2	2	1	1	30
# Imgs	800K	34K	350K	55K	15K	46K
# Peds	2.8M	222K	132K	25K	9.4K	123K
# Riders	67K	24K	11K	22K	3.3K	13K
# Seasons	-	-	1	1	1	4
Weather	dry, rain	dry, rain	dry	dry	dry	dry, rain
Time of day	day, night	day, night	day, night	-	day	day, night
Unblurred	✗	✗	✗	✗	✓	✓

This paper presents an experimental study on monocular and LiDAR-based 3D person detection. Its specific contributions are:

- A performance analysis of two state-of-the-art methods (PointPillars and AVOD) on KITTI, with respect to varying IoU and the underlying parameters of 3D bounding box location, extent and orientation.
- Results from domain transfer experiments between KITTI and ECP2.5D.

III. METHODOLOGY

The goal of 3D person detectors is to detect the bounding boxes of VRUs in the scene. In KITTI, these bounding boxes have seven degrees of freedom (fig. 2). The 3D position is given in a coordinate system with respect to the egovehicle, where x is the position of the bounding box center lateral to the vehicle, z is the position longitudinal to the vehicle (i.e. depth), and y determines the altitude of the bounding box center. The bounding box dimensions are specified by a width w, length l and height h, and finally each bounding box has a yaw rotation θ. The top and bottom face of the bounding box are assumed to be parallel to the $y = 0$ plane. The predicted bounding boxes will also have a detection score d related to them.

A. Intersection over Union (IoU)

To evaluate the performance of an object detector, one needs to count a predicted bounding box as valid or non-valid (i.e. true positive or false positive). In 3D (as well as 2D) object detection, the method to assess if a proposed bounding box is a true- or false-positive is based on IoU. It is defined as the intersection (or overlap) of a 3D bounding box prediction (B_p) and ground truth (B_{gt}) divided by the union of the prediction and ground truth. When both bounding boxes only have a yaw rotation, this can be written as [27]:

$$\text{IoU} = \frac{B_p \cap B_{gt}}{B_p \cup B_{gt}} = \frac{A_o \times h_o}{V_{gt} + V_p - A_o \times h_o} \tag{1}$$

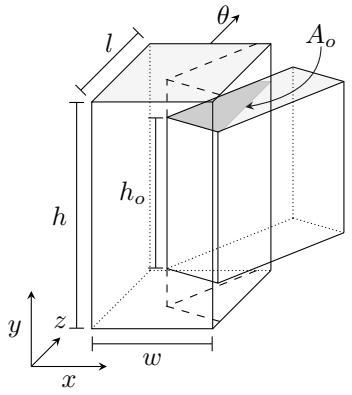


Fig. 2. A visualization of the parameters relevant for computing the IoU of a ground truth and predicted bounding box. The darker shaded area indicates the overlap area A_o. In this figure, the overlapping height h_o is equal to the height of the smaller bounding box.

Where V_p and V_{gt} are the volume of the predicted and ground truth bounding box. The overlap of volumes can be computed from the overlapping top-view area A_o and the overlapping height (h_o), see fig. 2. In KITTI, a predicted bounding box is seen as a true positive if it has an IoU of more than 0.5. Only one predicted bounding box can be marked as a true positive for any ground truth bounding box.

B. Performance metrics

After the true positives have been determined, it is possible to compute the two metrics as defined in the KITTI benchmark for 3D object detection: 3D Average Precision (AP_{3D}) and Average Orientation Similarity (AOS) [5].

The AP_{3D} averages the maximum attained precision s with at least a recall r for a fixed range of recall values [28]:

$$\mathrm{A}P_{3D} = \frac{1}{40} \sum_{r \in \{\frac{1}{40}, \frac{2}{40}, \ldots, 1\}} \max_{\tilde{r}: \tilde{r} \geq r} s(\tilde{r}) \tag{2}$$

As precision and recall both depend on the amount of true positives, the AP_{3D} strongly depends on the IoU threshold.

Where the AP_{3D} verifies whether the bounding boxes are in the correct place, the AOS additionally verifies the correctness of their orientations:

$$AOS = \frac{1}{40} \sum_{r \in \{\frac{1}{40}, \frac{2}{40}, \ldots, 1\}} \max_{\tilde{r}: \tilde{r} \geq r} \tilde{s}(\tilde{r}) \tag{3}$$

$$\tilde{s}(r) = \frac{1}{|\mathcal{D}(r)|} \sum_{i \in \mathcal{D}(r)} \frac{1 + \cos \Delta_\theta^{(i)}}{2} \delta_i \tag{4}$$

Where $\mathcal{D}(r)$ denotes the set of all objects at a specific recall rate r and $\Delta_\theta^{(i)}$ the difference between the estimated and the real orientation. The indicator δ_i is one if the predicted bounding box is seen as a true positive, and zero otherwise. If every true positive predicted bounding box has an orientation error of 0, eq. (4) reduces to the precision at that recall rate.

IV. Experiments

Experiments were performed with the codebase of the authors of AVOD[1] and the codebase recommended by the authors of PointPillars[2] as is, using the best performing network as reported in their papers. Thus for AVOD, we use AVOD-FPN, and for PointPillars, we use a spatial resolution of 0.16×0.16 m^2.

A. Datasets overview

Figure 3 shows the distribution of the VRUs locations relative to the vehicle, for the publicly available part of both KITTI and ECP2.5D. The bulk of the detections in the KITTI dataset lies within 30 m distance of the ego-vehicle. Both datasets have a bias towards VRUs being on the right side of the ego-vehicle.

We are using the same KITTI 1:1 train/validation split as specified by the AVOD and PP codebases. The KITTI dataset contains 2.2K/0.7K and 2.3K/0.9K pedestrian/cyclist annotations for the train and validation split respectively. The validation split can be divided in three parts which are “easy”, “moderate” and “hard”, as defined by KITTI. The ECP2.5D dataset has a larger amount of annotations for the 3D position and orientation, but lacks width, length and height annotation. We will use the median bounding box dimensions of the train split of the KITTI dataset so both networks can still regress a full bounding box. This paper uses the ”Day” subset of ECP2.5D as its basis. Additionally, the underlying EuroCity Persons (ECP) dataset misses an orientation label for 386 pedestrians and 144 riders, these are set to “Don’t Care”. This results in 62.3K/7.3K pedestrian/cyclist annotations in the training split, and 12.6K/1.3K pedestrian/cyclist annotations in the validation split. The test set ground truth annotations of both datasets is not made public, so all evaluations done in the rest of this paper are done using the validation splits of either dataset as mentioned here.

Both datasets use the Velodyne HDL-64E (LiDAR) sensor. The intensity of the LiDAR points in KITTI fall in 100 discrete bins of between 0 and 1. ECP2.5D has an intensity on a continuous range between 1.0 and 255.

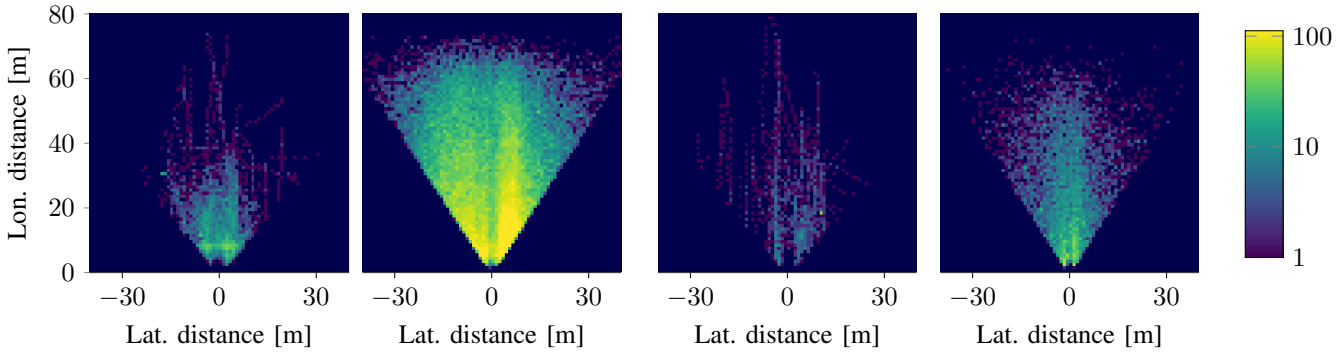
B. Effect of IoU on performance and error analysis

Performance with lower IoU constraints: Table III shows the performance of PointPillars and AVOD on KITTI for the cyclist and the pedestrian classes. PointPillars has a higher AP_{3D} than AVOD, even though their scores on the moderate test split on the KITTI benchmark differ less than one percent. However, the results we find for AVOD are comparable to those found on the validation split in the comparison study of [17]. Lowering the IoU threshold increases the AP_{3D} by a large margin. For example, the AP_{3D} of PointPillars on pedestrians increases from 55.8 to 77.2 (21 %).

This is further visualized in fig. 4, which shows a histogram of the IoU found for all true positive detections

[1] https://github.com/kujason/avod

[2] https://github.com/traveller59/second.pytorch



(a) KITTI pedestrians (b) ECP2.5D pedestrians (c) KITTI cyclists (d) ECP2.5D cyclists

Fig. 3. The overall distribution over location of pedestrians and cyclists in both the KITTI and ECP2.5D dataset as a log plot. In all figures, the egovehicle is positioned at $(0, 0)$, looking upwards. Each pixel in the image corresponds to a $1x1$ square meter area. The darkest blue region indicates areas with zero pedestrians.

TABLE III
AOS AND AP_{3D} PERFORMANCE OF POINTPILLARS (PP) AND AVOD, TRAINED ON KITTI AND EVALUATED ON THE MODERATE PART OF THE KITTI VALIDATION SPLIT.

	Pedestrian		Cyclist	
IoU	AP_{3D}	AOS	AP_{3D}	AOS
PP				
0.5	55.8	27.0	58.5	5.8
0.4	71.5	34.5	63.7	6.9
0.3	76.5	37.1	64.9	7.1
0.2	77.1	37.4	66.0	7.2
0.1	77.2	37.5	66.0	7.2
AVOD				
0.5	41.2	32.3	35.1	34.8
0.4	50.0	38.3	36.3	35.9
0.3	52.5	40.1	36.3	35.9
0.2	52.7	40.2	36.3	35.9
0.1	52.7	40.3	36.3	35.9

at an IoU threshold of 0.1. This histogram shows that for pedestrians more than 15% of the detections of PointPillars and 10% of the detections of AVOD had an IoU between 0.4 and 0.5, just outside the normal IoU threshold. A similar effect is seen for cyclists, albeit less strongly.

The upper bound of the AOS is the AP_{3D}, as mentioned in section III-B. Table III shows that even though the general detection accuracy of AVOD is lower than PointPillars, its AOS is almost perfect, especially for cyclists. The AOS of PointPillars is far worse than the AOS noted on the online KITTI benchmark. A closer inspection of the distribution of the orientation error (fig. 5) shows that for PointPillars, the orientation error peaks around 0 or 180 degrees. In the paper of PointPillars, the authors state that the orientation loss used cannot distinguish between flipped boxes, for which they use an additional binary classification loss. The orientation errors of PointPillars shown in fig. 5 seem to indicate that while the original overall orientation loss works as expected, there might be an implementation issue with the binary classification loss in the codebase of SECOND. As for

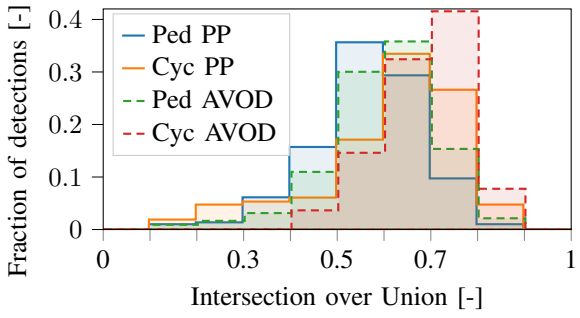


Fig. 4. PointPillars and AVOD trained on KITTI: a histogram of what fraction of true positive detections had what IoU (IoU threshold of 0.1).

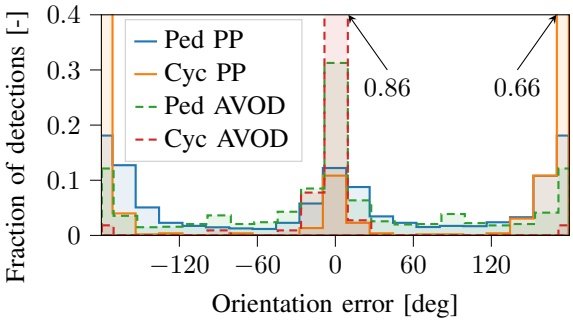


Fig. 5. PointPillars and AVOD trained on KITTI: A histogram of the orientation error. The arrows indicate the fraction of detections of the two bars outside of the y axis range. Most orientation errors lie either between -40 and 40 degrees, or between 140 and -140 degrees.

AVOD, almost all of the orientation estimates indeed have an error closer to 0 degrees as was expected from their AOS.

Error analysis of bounding box estimation: Figure 6 shows the error made in position and size of the predicted bounding boxes on pedestrians by PointPillars. The smallest errors are made on the x and the z estimation: the lateral and longitudinal position. The largest error is made on the width and length estimation. These depend on the stride of a pedestrian, as well as the location of their arms, which can be difficult to infer at larger distances.

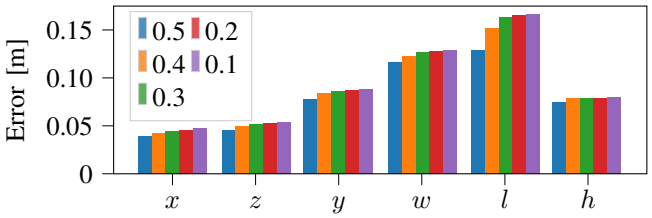


Fig. 6. PointPillars trained on KITTI: the average error between the prediction and the ground truth for the pedestrian detections on x, z, y, w, l and h, at different IoUs thresholds. The largest error is made on the altitude estimation, together with the bounding box width and length.

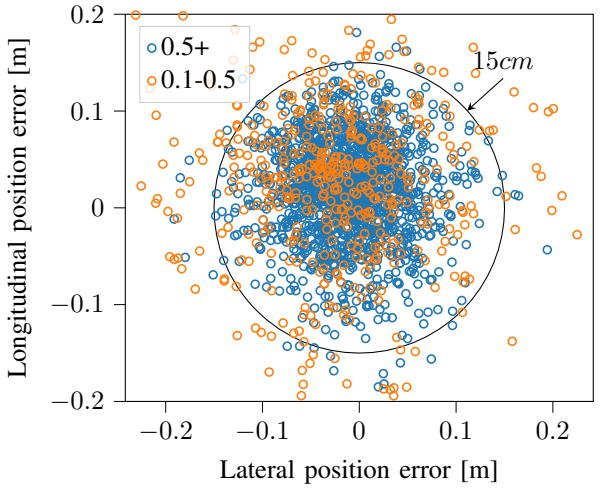


Fig. 7. PointPillars trained on KITTI: The localization error made by true positive detections of **pedestrians**, from a bird's eye viewpoint. Of the true positive detections with an IoU of over 0.5, 1462 out of 1494 detections lie within a radius of 15 cm. Of the true positive detections with an IoU between 0.1 and 0.5, 349 out of 478 lie within that radius.

The relatively small error in x and the z position (essentially a top-down position estimate) is visualized in fig. 7. It shows the x and z position error made for the true positive detections for the original IoU threshold, as well as the detections between an IoU of 0.1 and 0.5. A lot of the detections with an IoU below 0.5 are still accurate at estimating the position. For an IoU threshold of 0.5, nearly all of the true positive detections (1462 of the 1494) lie within a radius of 15 cm. When looking at the detections found with an IoU threshold of 0.1, a total of 1811 detections lie within a radius of 15 cm. In other words, using a radius of 15 cm for as a metric to determine true positives instead of an IoU of at least 0.5 shows an increase in detections of 23 %. The same data is put more succinctly in fig. 8, with cyclists added as well. It shows the amount of true positive detections which fall below a specific Euclidean position error. Cyclists see a smaller benefit, but as their annotated bounding boxes are larger, it is possible to make a larger position error without affecting the IoU as much.

Accuracy evaluation using fixed bounding boxes during training: The relatively large errors in width and length suggest that these two 3D object detectors are not able to properly estimate these. To investigate the influence of the bounding boxes dimensions, we train on a version of the KITTI dataset train split where we fix the dimensions of each VRU. The dimensions are fixed to the median dimensions of their respective class. Then, we evaluate on the original KITTI dataset validation split with the correct dimensions. See Table IV. Where at an IoU of 0.5, the performance of PointPillars on the pedestrian class drops with 1.2 %, the performance of the cyclist class even increases with 3.9 %. Next to that, AVOD shows an increase for both the pedestrian and the cyclist class.

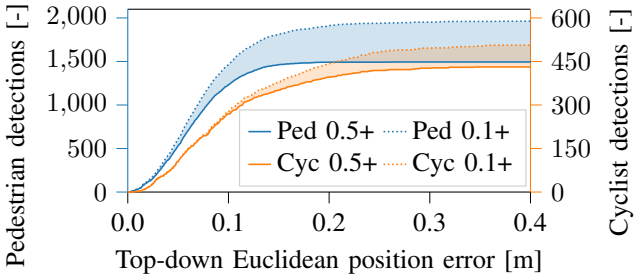


Fig. 8. PointPillars trained on KITTI: given a certain Euclidean position error threshold, how many detections would be inside. The solid line shows what Euclidean error is made by detections using the current default IoU threshold of 0.5. The dotted line shows the amount of detections within a given Euclidean error for an IoU threshold of 0.1. The shaded area then shows the amount of detections added.

TABLE IV

AP_{3D} PERFORMANCE OF POINTPILLARS (PP) AND AVOD FOR TWO IOU THRESHOLDS, EVALUATED ON THE MODERATE PART OF THE KITTI VALIDATION SPLIT. THE NETWORKS WERE TRAINED ON THE ORIGINAL KITTI GROUND TRUTH (ORIG.) OR THE GROUND TRUTH WITH FIXED BOUNDING BOX DIMENSIONS (FIXED).

	Pedestrian		Cyclist	
IoU	Orig.	Fixed	Orig.	Fixed
PP				
0.5	55.8	54.6	58.5	62.6
0.1	77.2	73.3	66.0	68.1
AVOD				
0.5	41.2	46.0	35.1	35.5
0.1	52.7	59.6	36.3	38.8

C. Cross-dataset Evaluations

To see how well each dataset generalizes, we train both networks on the one dataset and evaluate them on the other. Because the original PointPillars uses the intensity information of the points in the point cloud as well, we train it once *with* this intensity information present, and once *without*. AVOD does not use the intensity information, and therefore only needs to be trained once on each dataset. To ensure the datasets are compatible, we linearly rescaled the LiDAR intensity values in each dataset to the same range.

Table V shows the resulting AP_{3D}. PointPillars using LiDAR intensity data and the ("native") training sets, corresponding to the datasets tested, has the best performance on both datasets. When not using LiDAR intensity data, PointPillars' performance slightly drops, but still clearly

TABLE V

AP_{3D} PERFORMANCE OF POINTPILLARS (PP) AND AVOD FOR AN IOU OF 0.1 ON THE MODERATE VALIDATION SPLIT OF KITTI AND ECP2.5D. BOLD INDICATES HIGHEST PERFORMANCE IN THAT COLUMN.

	AP_{3D}	
Trained network	ECP2.5D	KITTI
with intensity :		
PP on ECP2.5D	**34.1**	46.7
PP on KITTI	6.9	**77.2**
w/o intensity :		
PP on ECP2.5D	32.8	55.4
PP on KITTI	26.0	67.5
AVOD on ECP2.5D	26.8	34.0
AVOD on KITTI	5.0	52.7

outperforms AVOD on both datasets, when using the native training sets. Performance of both methods was significantly lower on ECP2.5D vs. KITTI,

When non-native training sets are used, performances degrade significantly for both methods, both when moving from KITTI to ECP2D, and vice versa. The resulting performance degradation for PointPillars is less severe when the LiDAR intensity data is not used. More research is needed to improve cross-domain adaptation (e.g. [29]).

V. CONCLUSION

This paper presented an experimental study on 3D person localization in traffic scenes, on the basis of monocular vision and LiDAR data. In experiments on KITTI, we found that whereas headline results (AP_{3D}) results might seem low, the 3D box center localization accuracies are in fact quite high. The errors lowering AP_{3D} are mostly related to the estimates of the bounding box extents (especially, width and length).

PointPillars clearly outperformed AVOD (AP_{3D} of 68% vs. 53% and 33% vs. 27%, for KITTI and ECP2.5D respectively, when not using LiDAR intensity information). Performance of both methods was significantly lower on ECP2.5D vs. KITTI, we attribute this to a larger prevalence of distant persons with fewer LiDAR points in ECP2.5D.

Domain transfer experiments indicated the two datasets have quite different biases, in the sense that training on one and testing to the other leads to significantly degraded performance (upwards of AP_{3D} of 6.8%). Further research is needed on cross-domain adaptation.

ACKNOWLEDGMENT

This work received support from the Dutch Science Foundation NWO-TTW within the SafeVRU project (nr. 14667).

REFERENCES

[1] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. on PAMI*, vol. 34, no. 4, pp. 743–761, 2011.

[2] X. Li, F. Flohr, Y. Yang, H. Xiong, M. Braun, S. Pan, K. Li, and D. M. Gavrila, "A new benchmark for vision-based cyclist detection," in *Proc. of the IEEE IV*, 2016, pp. 1028–1033.

[3] M. Braun, S. Krebs, F. B. Flohr, and D. M. Gavrila, "EuroCity Persons: A novel benchmark for person detection in traffic scenes," *IEEE Trans. on PAMI*, vol. 41, no. 8, pp. 1844–1861, Aug 2019.

[4] C. G. Keller, T. Dang, H. Fritz, A. Joos, C. Rabe, and D. M. Gavrila, "Active pedestrian safety by automatic braking and evasive steering," *IEEE Trans. on ITS*, vol. 12, no. 4, pp. 1292–1304, 2011.

[5] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. of the IEEE CVPR*, 2012.

[6] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. of the IEEE CVPR*, 2019, pp. 12 697–12 705.

[7] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D Proposal Generation and Object Detection from View Aggregation," *IEEE IROS*, pp. 5750–5757, October 2018.

[8] M. Braun, S. Krebs, and D. M. Gavrila, "ECP2.5D - Person localization in traffic scenes," in *Proc. of the IEEE IV*, 2020.

[9] A. Naiden, V. Paunescu, G. Kim, B. Jeon, and M. Leordeanu, "Shift R-CNN: Deep monocular 3D object detection with closed-form geometric constraints," in *Proc. of the ICIP*, Sep. 2019, pp. 61–65.

[10] B. Yang, M. Liang, and R. Urtasun, "HDNET: Exploiting HD maps for 3D object detection," in *Proc. of the CoRL*, vol. 87, Oct 2018, pp. 146–155.

[11] G. P. Meyer, A. Laddha, E. Kee, C. Vallespi-Gonzalez, and C. K. Wellington, "LaserNet: An efficient probabilistic 3D object detector for autonomous driving," in *Proc. of the IEEE CVPR*, June 2019.

[12] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. of the IEEE CVPR*, June 2018.

[13] Z. Wang, H. Fu, L. Wang, L. Xiao, and B. Dai, "SCNet: Subdivision coding network for object detection based on 3D point cloud," *IEEE Access*, vol. 7, pp. 120 449–120 462, 2019.

[14] V. A. Sindagi, Y. Zhou, and O. Tuzel, "MVX-Net: Multimodal voxelnet for 3D object detection," in *IEEE ICRA*, May 2019, pp. 7276–7282.

[15] D. Xu, D. Anguelov, and A. Jain, "PointFusion: Deep Sensor Fusion for 3D Bounding Box Estimation," in *Proc. of the IEEE CVPR*, June 2018.

[16] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and Segmentation," in *Proc. of the IEEE CVPR*, July 2017.

[17] M. Roth, D. Jargot, and D. M. Gavrila, "Deep end-to-end 3D person detection from camera and lidar," in *Proc of the IEEE ITSC*. IEEE, 2019, pp. 521–527.

[18] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "Multimodal vehicle detection: fusing 3D-LIDAR and color camera data," *Pattern Recognition Letters*, vol. 115, pp. 20–29, 2018.

[19] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D object detection from RGB-D data," in *Proc. of the IEEE CVPR*, June 2018, pp. 918–927.

[20] Z. Wang and K. Jia, "Frustum ConvNet: Sliding frustums to aggregate local point-wise features for amodal 3D object detection," in *IEEE IROS*. IEEE, 2019.

[21] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "STD: Sparse-to-dense 3D object detector for point cloud," in *Proc. of the IEEE ICCV*, October 2019.

[22] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "PointPainting: Sequential fusion for 3D object detection," 2019.

[23] Waymo, "Waymo Open dataset: An autonomous driving dataset," 2019.

[24] H. Caesar *et al.*, "nuScenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.

[25] M.-F. Chang *et al.*, "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. of the IEEE CVPR*, 2019.

[26] R. Kesten *et al.*, "Lyft level 5 AV dataset 2019," https://level5.lyft.com/dataset/, 2019.

[27] D. Zhou *et al.*, "IoU loss for 2D/3D object detection," in *International Conference on 3D Vision*. IEEE, 2019, pp. 85–94.

[28] A. Simonelli, S. R. R. Bulò, L. Porzi, M. López-Antequera, and P. Kontschieder, "Disentangling monocular 3D object detection," *arXiv preprint arXiv:1905.12365*, 2019.

[29] C. Rist, M. Enzweiler, and D. M. Gavrila, "Cross-sensor deep domain adaptation for LiDAR detection and segmentation," in *Proc. of the IEEE IV*, 2019.