Augmented Radar Points Connectivity based on Image Processing Techniques for Object Detection and Classification

Mohamed Sabry[1][a], Ahmed Hussein[2][b], Amr Elmougy[1][c] and Slim Abdennadher[1][d]

[1] *Computer Engineering Department, German University in Cairo (GUC), Cairo, Egypt*

[2] *IAV GmbH, Intelligent Systems Functions Department, Berlin, Germany*

Keywords: Computer Vision, Image Processing, Radars, Point Clouds, Object Detection and Classification.

Abstract: Perception and scene understanding are complex modules that require data from multiple types of sensors to construct a weather-resilient system that can operate in almost all conditions. This is mainly due to drawbacks of each sensor on its own. The only sensor that is able to work in a variety of conditions is the radar. However, the sparseness of radar pointclouds from open source datasets makes it under-perform in object classification tasks. This is compared to the LiDAR, which after constraints and filtration, produces an average of 22,000 points per frame within a grid map image representation of 120 x 120 meters in the real world. Therefore, in this paper, a preprocessing module is proposed to enable the radar to partially reconnect objects in the scene from a sparse pointcloud. This adapts the radar to object classification tasks rather than the conventional uses in automotive applications, such as Adaptive Cruise Control or object tracking. The proposed module is used as preprocessing step in a Deep Learning pipeline for a classification task. The evaluation was carried out on the nuScenes dataset, as it contained both radar and LiDAR data, which enables the comparison between the performance of both modules. After applying the preprocessing module, this work managed to make the radar-based classification significantly close to the performance of the LiDAR.

1 INTRODUCTION

There have been many studies recently to develop accurate and efficient pointcloud-based object detection and classification modules. More focus has been on the use of LiDAR sensors as they produce dense pointcloud data that is capable of classifying objects in the environment around an ego vehicle. Each object is represented with a fair amount of points which gives descriptive spatial information lying within the geometry of the objects. However, for the automotive industry, LiDARs are still not approved for serial production. This means that automotive manufacturers will need to alter a lot of processes in their production lines to be able to add LiDAR sensors to their serial production vehicles intended for autonomy and Advanced Driver Assistant Systems (ADAS). In addition, the price of the LiDAR sensors is significantly higher than the automotive as shown in (Mohammed et al., 2020). The mentioned price tag significantly adds to the challenges as well. Unlike the LiDAR, automotive Radars are approved for serial production vehicles meant for levels 1 and 2 for autonomy as well as being significantly cheaper as aforementioned. However, the available online data from Radars are relatively sparse compared to the LiDAR (Caesar et al., 2019).

One of the main challenges for dealing with the radar data from public access datasets is the aforementioned sparseness of the Radar data. This has limited the serial production Radars for automotive applications to Adaptive Cruise Control (ACC) and automatic braking. When used in other applications usually meant for the LiDARs and Cameras, such as object detection and classification, Radars lag behind significantly as can be seen on the nuScenes Leaderboard for 3D object detection and classification. Given the Radars capability of working in various weather conditions in which other sensors can fail, such as rain and snow. This resilience can fill in the gap if it can be applied in applications where the other sensors excel, such as object detection and classification.

[a] https://orcid.org/0000-0002-9721-6291
[b] https://orcid.org/0000-0001-6621-4923
[c] https://orcid.org/0000-0003-0250-0984
[d] https://orcid.org/0000-0003-1817-1855

Sabry, M., Hussein, A., Elmougy, A. and Abdennadher, S.
Augmented Radar Points Connectivity based on Image Processing Techniques for Object Detection and Classification.
DOI: 10.5220/0010860600003124
In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 5: VISAPP*, pages 535-542
ISBN: 978-989-758-555-5; ISSN: 2184-4321

To be able to utilize the Radar for 3D object detection and classification task, this paper proposes a preprocessing module that can be applied to Radar data to significantly enhance its object detection and classification capabilities to be merged with other sensors for a better overall perception system with more redundancy and robustness. In addition, this will allow the vehicle to function in the aforementioned weather conditions where the LiDARs and Cameras fail to function.

To the best of our knowledge, this is the first approach to optimize the object classification capabilities of the radar pointcloud based mainly on the Cartesian coordinates.

The remainder of this paper is organized as follows. Section 2 discusses the state of the art. Section 3 discusses the proposed radar preprocessing module. Next, the experimental work is introduced including the implementation details, the dataset used and the evaluation metrics in Section 4. Section 5 shows the proposed algorithm object classification results and the discussion. Finally, Section 6 includes concluding remarks and future work.

2 STATE OF THE ART

2.1 Radar Only Detection and Classification

In (Palffy et al., 2020), the authors presented a Radar based single-frame, multi-class detection method for moving road users (pedestrian, cyclist, car) which is based on feature extension and a Convolutional Neural Network (CNN) for the classification. They utilized low-level Radar cube data. The authors provided the data format and input shape of the data used. However, the dataset was not public which eliminated the opportunity to compare the approach with other ones. In addition, the authors used the Radar Data to detect moving objects only, so stationary objects were excluded from the calculations.

In (Danzer et al., 2019), the authors utilized the Radar data for car detection using Radar data. However, the data contained only one class of objects with is the Car label. In addition, the test set mentioned in the paper does not include a test case were the their target vehicle is stationary.

2.2 Radar-camera Fusion-based Proposals

In (Kim et al., 2020), the authors proposed using a self-produced short-range FMCW Radar with the YOLO (Redmon and Farhadi, 2018) network. The dataset used in this paper was collected by the authors and is not publicly available. In addition, the data used is from a "self-produced short-range FMCW Radar" that produces 512 points. This denotes that this is not off the shelf Radar that gives the points in the Cartesian coordinates directly such as the Radars that are used in a public dataset such as nuScenes.

In (Nabati and Qi, 2020) the authors focus on the problem of Radar and camera sensor fusion and propose a middle-fusion approach to exploit both Radar and camera data for 3D object detection. Their approach first uses a center point detection network to detect objects by identifying their center points on the image then associates the Radar detections to their corresponding object's center point. The associated Radar detections are used to generate Radar-based feature maps to complement the image features, and regress to object properties such as depth, rotation and velocity. The results obtained were better than the state-of-the-art camera-based algorithm by more than 12% in the overall nuScenes Detection Score (NDS).

Several other approaches utilized data from LiDARs, Radars and Cameras for object detection. (Wang et al., 2020) is an example denoting the approach.

In this paper, the variables used in (Danzer et al., 2019) and some of the preprocessing steps used in (Dung, 2020) were modified and applied along side an image processing technique to further develop the object classification capabilities of Radar data for multiple classes as well as dynamic and static objects.

3 METHODOLOGY

In this section, the main components of the preprocessing algorithm are introduced. The flow of the section can be seen in Figure 1.

3.1 Data Concatenation

The first step in the proposed preprocessing module is the concatenation of multiple frames from previous timestamps. This is crucial as, in a single timestamp, the Radars surrounding the ego vehicle from the nuScenes dataset produce a maximum of 625 points (125 points each) with 18 features each. This number of points is very sparse compared to the num-

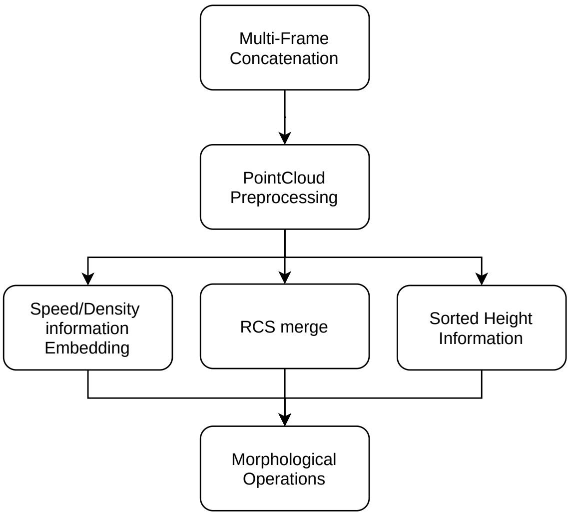

Figure 1: Proposed approach pipeline for Radar improvement for object classification.

ber of points that the LiDAR produces in a single timestamp. From the testing in this work with the nuScenes dataset, the LiDAR produced an average of 22,000 points per frame after preprocessing and filtering, as aforementioned. This is one of the obvious reasons for lack of performance in applications such as object classification. The concatenated frames were extracted and compensated by the ego vehicle's ego pose in each of the previous timestamps directly using the nuScenes devkit.

3.2 Initial Information Extraction

After the data is concatenated, points that are found to be farther than a 60 meter radius from the ego vehicle are removed. The points are further placed in an image grid map representation with the size of 608x608 pixels representing 120 x 120 meters. Following the extraction step, the information is then further processed by the following steps used in (Dung, 2020):

- The pointcloud location data is converted from meters into discrete pixel locations to be fitted into a 608x608 pixels image.
- The discretized location information is used to sort the unique points from the pointclouds. The pointclouds' height information are then normalized to create a heightMap which is a single 608x608 image channel denoting a GridMap representing the normalized height information in the Bird's eye view form.
- Information denoting the density of points are represented as in the previous point but the values within the channel being the normalized counts for each of the unique values in the discretized pointcloud. This generated image channel is utilized as the densityMap.

3.3 Information and Relation Extraction

As aforementioned, the data presented is sparse which significantly reduces the effectiveness of the preprocessing steps used for Radars. Given this observation and the available 18 features for each point in the Radar data, the following features were utilized as a first step to combine data components:

- The compensated velocities of points were added as lines drawn with the speed represented in the color and the direction denoted by the velocity on the densityMap. This is to try and cover a larger area to compensate for the sparseness of the radar data.
- The Radar cross-section (RCS) feature, which is the area of the object the Radar hits (Knott et al., 2004), was extracted from each point and used as an independent channel. This is considered to be a stable feature to use as it can work with both stationary and moving objects. This channel is denoted as the rcsMap.

As a final step for the information extraction from the pointcloud, The heightMap alongside the densityMap with the added compensated velocity information and the rcsMap to create a 3 channel Image with the needed information for the following step.

3.4 Morphological Operations for Object Connectivity

After extracting the mentioned channels, the points produced from the Radar were still relatively sparse with an average of 4500 points from the concatenated data compared to an average of 22,000+ points from the LiDAR in one sample frame only. To overcome the sparseness issue, the second step from the data components combination was applied. Applying Morphological operations on the aforementioned channels to tackle the task (Comer and Delp III, 1999). This helps in overlapping sparse points from radar pointclouds which increases connectivity between points belonging to the same object as a part of a partial scene reconstruction. For this step, the equations used can be seen in (1), (2), (3) and (4).

$$D_r = X \oplus H \tag{1}$$

$$E_r = X \ominus H \tag{2}$$

$$C = (X \oplus H) \ominus H \quad (3)$$

$$O = (X \ominus H) \oplus H \quad (4)$$

where H is the structuring element used for the morphological operations, X is the original color image, E_r is the eroded image, D_r is the dilated image, O is the result of the opening operation on the image and C is the result of the closing operation on the image. An example is shown in Figure 2

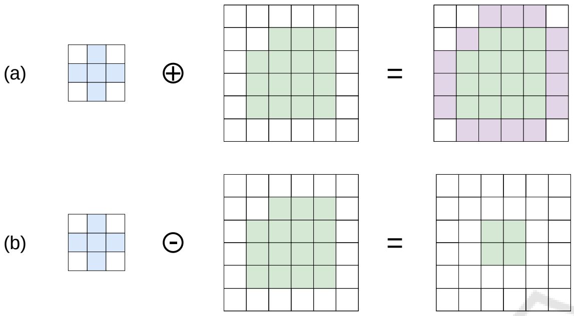

Figure 2: The top row shows an example of the dilation operation applied on a binary image (a). On the left most is the kernel used for the morphological operation, the middle is the image which the operation is to be applied on and on the right most the result of the operation. Furthermore, the bottom row shows an example of the erosion operation on an image.

In this paper, the morphological operations were applied on 3 channels for a color image, with each channel having a values from 0 to 255.

By trial and error, it was found that applying a closing operation with 5 dilation iterations followed by 3 erosion iterations then a final dilation operation with yet again 5 iterations yields the best results in this work. This helps maximize the values of objects in the scene compared to static infrastructure objects in the scene.

3.5 Deep Neural Network Prediction

After preprocessing the data using the aforementioned steps, the data is used with the Deep Learning (DL) network from (Dung, 2020), which is based on the work of (Li et al., 2020). The network was originally tested on the KITTI dataset (Geiger et al., 2012) and has been modified to run on the nuScenes dataset. The main purpose of the paper is to compare LiDAR and the modified Radar data head to head, comparing the same labels to shed the light on the performance gain of the Radar data after applying the proposed preprocessing module.

4 EXPERIMENTAL WORK

4.1 Implementation

The proposed approach is implemented in Python using OpenCV, numpy and pytorch to manipulate the data. All experiments and tests were carried out on a computer with an Intel i7-8800K 6-core processor using 32GB of RAM, running Ubuntu version 18.04, with a RTX 2080Ti GPU.

4.2 Dataset

As aforementioned, nuScenes was selected as the publicly available dataset (Caesar et al., 2019) which contains the needed sensor setup for the testing. The full dataset provides ground-truth labels for object detection tasks for 1000 scenes with more than 30,000 samples including the training, testing and validation. The ground-truth is provided as a list containing the translation, size, orientation, velocity, attributes and detection names for each object. For the test sequences, evaluation results are obtained by submitting to the nuScenes website. The training was applied on half the trainval dataset consisting of just over 16,000 samples. The training took around 35 hours for 300 epochs.

For the target of this work which is restricted to the improvement of the Radar object detection and classification, the classification was restricted to three classes, namely vehicles, pedestrians and bicycles. The nuscenes devkit was used to access the data (Caesar et al., 2019).

As aforementioned, the network from (Dung, 2020) was utilized and modified to fit the nuScenes dataset. The trainval dataset was used for the Network training. The training was set for 300 epochs.

4.3 Metrics

For all tests, the evaluation was done by computing the mean average precision across classes. For each class, the average precision, transnational, scaling and orientation errors as well as the nuScenes detection score (NDS, weighted sum of the individual scores) were calculated as well. These metrics were extracted from the nuScenes devkit. These metrics were strictly used to compare the Radar's performance against the LiDAR for moving and stationary objects without taking velocity into consideration.

5 RESULTS AND DISCUSSION

The evaluation was done on the mini version of the dataset as proof of concept. The results based on the LiDAR data, the concatenated Radar data and the Radar data with the full proposed module can be seen in Figure 5, Table 1 and Table 2.

Table 1: A comparison between the results of the LiDAR, the normal concatenated Radar, and the proposed module.

Approach	Object Class	AP	ATE	ASE	AOE
LiDAR	Pedestrian	0.543	0.384	0.781	0.576
Concatenated Radar		0.000	1.000	1.000	1.000
Proposed Approach		0.475	0.446	0.782	0.685
LiDAR	Vehicle	0.685	0.404	0.780	0.320
Concatenated Radar		0.000	1.000	1.000	1.000
Proposed Approach		0.635	0.427	0.781	0.383
LiDAR	Cyclist	0.361	0.420	0.797	0.246
Concatenated Radar		0.000	1.000	1.000	1.000
Proposed Approach		0.338	0.449	0.794	0.493

Table 2: Overview quantitative metrics results.

Approach	mAP	mATE	mASE	mAOE	NDS
LiDAR	0.5297	0.4027	0.7862	0.3804	0.6079
Concatenated Data	0.0001	1.0000	1.0000	1.0000	0.0001
Proposed Approach	0.4827	0.4405	0.7858	0.5201	0.5667

No statistics were done on the radar data alone without any further preprocessing as the visual results were very poor on their own based on the network predictions with weights taken from the LiDAR data as well as the weights trained on the Radar data without any preprocessing as seen in Figure 3. For the visual results of the comparisons refer to Figure 4.

The proposed approach shows a significant improvement over the Radar data without the proposed module as well as a very similar performance to the LiDAR based predictions based on the used testing in this paper. As can be seen in the tables, the overall results of the proposed approach are satisfactory even in crowded environments.

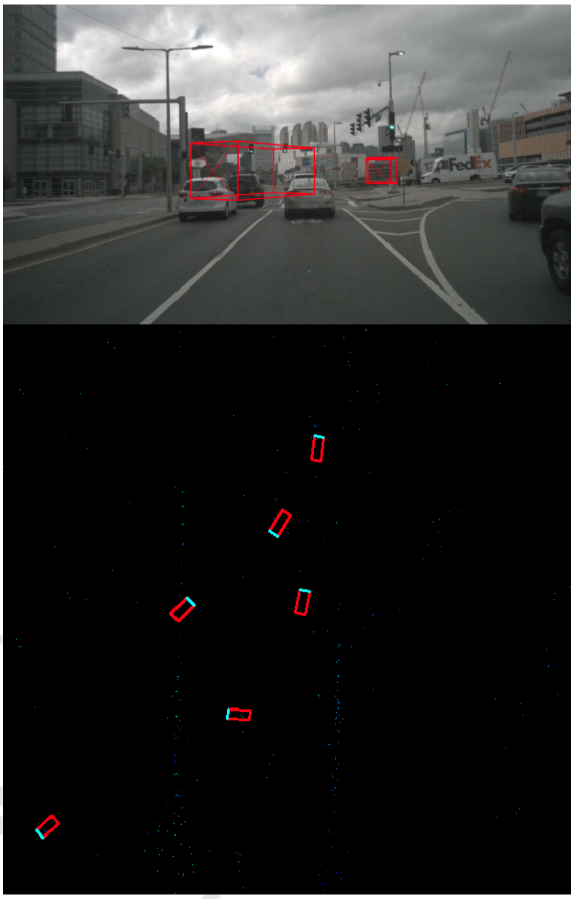

Figure 3: Radar alone without any thing training on Radar data.

5.1 Discussion

The main contribution of the proposed approach is the introduction of new preprocessing module in the the object detection and classification pipeline meant for radar pointclouds to enhance the overall results. It is worth noting that since the main aim of the paper is to reduce the gap between the Radar performance and the LiDAR performance in object detection and classification based on 3 classes as a proof of concept, the comparison to the ranking system of the nuScenes dataset is not straightforward since comparisons are based on 8 classes in addition to the ability to use all available sensors from the dataset to be able to produce the overall performance of the object detection and classification. However, to the authors knowledge, this is the first paper to address the use of Radar data pointclouds the same way as the LiDAR pointclouds and get a very near performance. Through further fusion with automotive cameras with this algorithm, the performance can surpass the LiDAR performance. This concludes that serial production vehic-

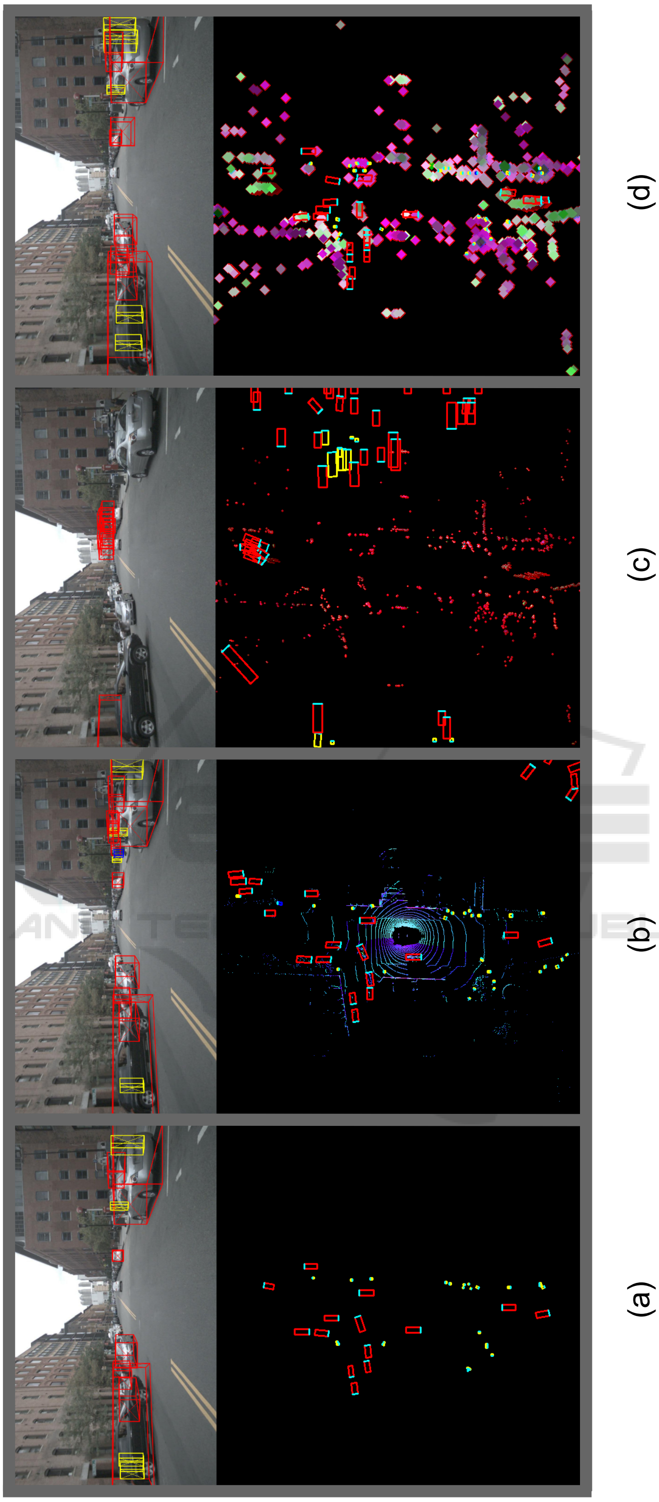

Figure 4: A visualization for the produced network predictions compared to the ground truth. The top row denotes the bounding box projections and the bottom row represents a top view of the labels/pointcloud data. (a) represents the ground truth, (b) represents the predictions on the LiDAR data, (c) represents the prediction on the Radar data based on concatenation and speed information only and (d) represents the network prediction on the Radar data after the proposed preprocessing module. The prediction is applied on the pointcloud data from the nuScenes dataset.

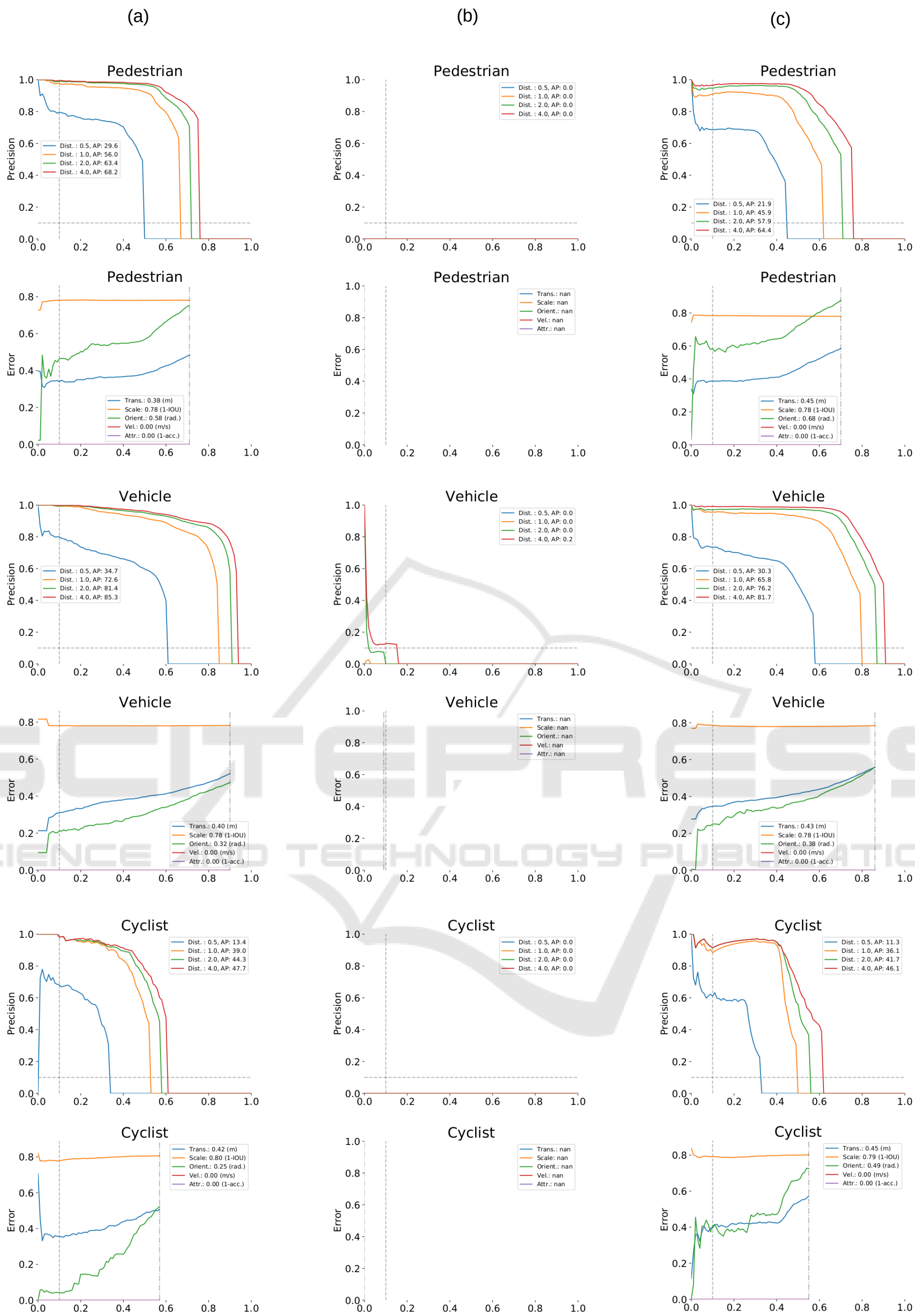

Figure 5: A comparison between the results of (a) the LiDAR , (b) the normal concatenated Radar and (c) the Radar after the proposed module. The data was extracted from the mini dataset version from nuScenes to test the proposed concept.

les can get a LiDAR like performance, if not better, using the serial production automotive Radars and Cameras.

6 CONCLUSION

In this paper, an enhancement preprocessing module has been proposed for radar data to be able to enhance the object classification performance. The proposed theory was tested on the network architecture based on fpn_resnet from (Dung, 2020) and the results showed that indeed the proposed module provided a surge in performance compared to just the radar data without the module and a significant close-in on the LiDAR performance compared to the ground-truth data.

As for future work, the proposed algorithm is to be extended to run on the 8 classes on the Leaderboard of the nuScenes evaluation server. Furthermore, in order to improve the accuracy of the object classification to surpass the LiDAR performance, the radar data is to be fused with the camera to surpass the LiDAR performance on its own.

REFERENCES

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2019). nuscenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*.

Comer, M. L. and Delp III, E. J. (1999). Morphological operations for color image processing. *Journal of electronic imaging*, 8(3):279–289.

Danzer, A., Griebel, T., Bach, M., and Dietmayer, K. (2019). 2d car detection in radar data with pointnets. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 61–66. IEEE.

Dung, N. M. (2020). Super-fast-accurate-3d-object-detection-pytorch. https://github.com/maudzung/Super-Fast-Accurate-3D-Object-Detection.

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE.

Kim, W., Cho, H., Kim, J., Kim, B., and Lee, S. (2020). Yolo-based simultaneous target detection and classification in automotive fmcw radar systems. *Sensors*, 20(10):2897.

Knott, E. F., Schaeffer, J. F., and Tulley, M. T. (2004). *Radar cross section*. SciTech Publishing.

Li, P., Zhao, H., Liu, P., and Cao, F. (2020). Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. *arXiv preprint arXiv:2001.03343*, 2.

Mohammed, A. S., Amamou, A., Ayevide, F. K., Kelouwani, S., Agbossou, K., and Zioui, N. (2020). The perception system of intelligent ground vehicles in all weather conditions: a systematic literature review. *Sensors*, 20(22):6532.

Nabati, R. and Qi, H. (2020). Centerfusion: Center-based radar and camera fusion for 3d object detection. *arXiv preprint arXiv:2011.04841*.

Palffy, A., Dong, J., Kooij, J. F., and Gavrila, D. M. (2020). Cnn based road user detection using the 3d radar cube. *IEEE Robotics and Automation Letters*, 5(2):1263–1270.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Wang, L., Chen, T., Anklam, C., and Goldluecke, B. (2020). High dimensional frustum pointnet for 3d object detection from camera, lidar, and radar. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1621–1628. IEEE.