

Ghost Target Detection in 3D Radar Data using Point Cloud based Deep Neural Network

Mahdi Chamseddine,
Jason Rambach,
and Didier Stricker

Augmented Vision Department
German Research for Artificial Intelligence
Kaiserslautern, Germany

{mahdi.chamseddine, jason.rambach, didier.stricker}@dfki.de

Oliver Wasenmüller
Hochschule Mannheim
Mannheim, Germany

Abstract—Ghost targets are targets that appear at wrong locations in radar data and are caused by the presence of multiple indirect reflections between the target and the sensor. In this work, we introduce the first point based deep learning approach for ghost target detection in 3D radar point clouds. This is done by extending the PointNet network architecture by modifying its input to include radar point features beyond location and introducing skip connections. We compare different input modalities and analyze the effects of the changes we introduced. We also propose an approach for automatic labeling of ghost targets 3D radar data using lidar as reference. The algorithm is trained and tested on real data in various driving scenarios and the tests show promising results in classifying real and ghost radar targets.

I. INTRODUCTION

The field of autonomous driving is of an ever rising importance in the automotive industry. To achieve higher levels of autonomy, it is essential to be able to robustly detect and track obstacles and other road users such as pedestrians, cyclists, and other motorists. For this reason, vehicles are being equipped with an increasing number of sensors to learn more about their surroundings. Radar sensors have been an integral part of driver assistant systems but are now being tasked in adding more autonomy to vehicles and detect more complex driving scenarios. Radar sensors are desirable not only because of their robustness to different weather conditions and low production costs, but also for their ability to instantly measure the radial velocity of detected targets.

A radar sensor emits a signal and parts of it are reflected and returned as an echo upon hitting a target. From the echo, the distance (range) to the detected target can be calculated based on the time of flight, and the angle of arrival (azimuth angle) is calculated by having multiple receiving antennas with a known distance and calculating the time delay of receiving the echo at each antenna. The relative radial velocity (Doppler velocity) of the target is measured using the frequency shift between the incident and reflected waves [1], [2].

Since a radar sensor usually emits waves in all directions within its field of view, and since target locations are calculated based on the angle of arrival of the echos, the possibility that the reflected signal takes an indirect path between the sensor

and the target arises. Such an indirect reflection path leads to detections at wrong positions either with a different range, azimuth, or both. Those detections are commonly referred to in the literature as ghost targets or multi-path reflections. It is essential to detect the ghost targets to ensure a reliable and robust performance of automotive radars. Fig. 1 shows an example where radar waves are reflected off the target vehicle to the ego vehicle over the guardrail, thus resulting in the detection of a ghost target on the opposite side of the rail.

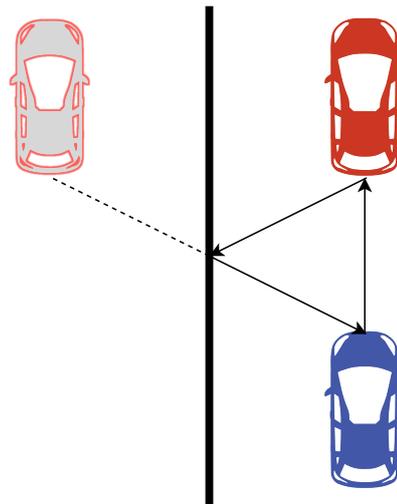


Fig. 1. The ego vehicle equipped with the radar sensor (blue), detects a ghost target (red silhouette) on the wrong side of the guardrail due to the indirect reflection of the radar signals off the target vehicle (red)

The presence of ghost targets in radar data reduces the reliability of the sensors and the robustness of its detections. Robust radar detection is essential for the safe operation of autonomous and highly automated vehicles.

The unpredictability of modern driving scenarios lead to a complexity in the ghost target problem, especially in urban environments where numerous and unknown reflecting surfaces are present. This has made it extremely challenging to solve ghost target problem using conventional, model based

methods that either require the model of the target [3] or the environment as done in through-wall radar imaging [4]–[6].

Furthermore, labeling radar data for ghost targets is a challenging task because many surfaces can reflect electromagnetic waves and it can become a more daunting and error prone task the more complex the scenario is, and the more densely the data is acquired.

Due to the ability of deep learning algorithms to recognize hidden patterns in data, and taking into consideration that ghost target detection is a challenging and unsolved problem, we present in this paper the following contributions:

- the first point based deep learning approach for ghost target detection in dense 3D radar point clouds to the best of our knowledge
- an automatic labeling approach that uses lidar for real/ghost target estimation to overcome the problem of missing data annotations
- an evaluation of different network architectures and input modalities to solve the problem

In Section II we present the literature review concerning the ghost target detection problem. Then in Section III the proposed approach is presented, including the network architecture, dataset, and the reference calculation method. The experiments and results are explained in Section IV followed by the conclusion in Section V.

II. RELATED WORK

A. Ghost Target Detection

To classify ghost targets in radar data, a model based approach was proposed by Roos et al. [3]. The measured velocity vector orientation is compared to the orientation in the vehicle model fit to the data where a mismatch between the orientations indicates a ghost target. The method however requires at least two radars to properly estimate the complete orientation vector. Model based approaches can be very robust in detecting ghost targets when the measurement fits the model, but the complexity of multipath reflection scenarios and the accuracy of the models can limit the effectiveness of those approaches.

As opposed to model based approaches, data driven approaches are less restricted by the measurement scenarios and can tackle more complexity. Ryu et al. [7] placed a radar at an intersection to measure the traffic. Different features of detections such as track lifetime, displacement, and velocity were used to train a multilayer perceptron for ghost target detection. The measurements however, coming from a fixed radar, are too simplistic and unrepresentative of a real automotive scenario with changing environment and driving conditions.

Prophet et al. [8] tested different machine learning based methods for ghost target classification. The authors concluded that random forests classifiers outperformed support vector machines and k-nearest neighbor algorithms for this task. Garcia et al. [9] furthered their work by using a deep convolutional neural network with an encoder-decoder architecture to classify ghost targets in 2D. The results show the effectiveness

of data driven techniques in detecting ghost targets in radar data. The methods applied are limited to moving ghost targets in 2D data, low resolution radars, and with no elevation information considered.

B. Point-Based Deep Learning

For object detection and point-wise segmentation in 3D point clouds, several network architectures have been proposed. PointNet by Qi et al. [10], uses spatial information and optionally other local or global features, to process point clouds. The network can process on unordered data, is invariant to transformations, and can capture interactions among points. The authors extended their work in PointNet++ [11] by allowing the network to process point clouds at multiple scales, to capture more details in the data. VoxelNet by Zhou et al. [12] processes point clouds as voxels using 3D convolutions, and Lang et al. [13] built on the ideas from PointNet and VoxelNet by treating the point space as vertical voxels, or pillars, each processed using a PointNet.

Schumann et al. [14] based their work on PointNet++ to perform semantic segmentation on 2D radar point clouds. The work showed promising results for detecting different classes of road users in hand-labeled data. They also showed that using more features from the sensor measurements improved the overall performance of the algorithm. In another work by [15] the PointNet architecture is used to detect objects in sparse 2D radar data. The authors used the same classification and segmentation networks in PointNet and augmented them with a 2D bounding box estimation network.

C. Non-Line-of-Sight Radar Approaches

Scheiner et al. [16] use the phenomenon of multipath in radar to track targets around corners in automotive scenarios. The work focuses on the detection of pedestrians to improve road safety. A lidar sensor is used to measure the reflecting surface and thus identify candidate multipath points. The authors demonstrate in this paper the importance of identifying ghost targets in automotive scenarios.

Other applications for non-line-of-sight radar is through-wall tracking of people. Setlur et al. [4], [5] tackle this problem using ultra-wideband radar while knowing the dimensions of the lab environment and the possible reflections that can occur. While this technology uses the physical phenomenon of multipath reflections, it is only limited to controlled environments.

In contrast to the existing work described, we present a solution that classifies **3D radar points** using a **point-based deep neural network** that works on **unordered point cloud data**.

III. PROPOSED APPROACH

In section III-A we describe our proposed a deep neural network architecture for ghost target detection in 3D radar point clouds. We also discuss different variations of the network. The dataset used is explained next in section III-B. And finally, we present our suggestion to tackle the labeling problem and the approach we used in section III-C.

A. Network Architecture

For the classification of real and ghost targets, a neural network designed for processing of point cloud data was trained. The network used is based on PointNet architecture [10] and modified to the problem at hand as can be seen in Fig. 2. We selected PointNet as our base architecture because unlike VoxelNet [12] and PointPillars [13] which are more suited for object detection problems, PointNet processes points individually and thus can learn individual point features that are needed for the identification of ghost targets. It is also able to learn global scene features and append them to the local point features, thus allowing for the use of both local and global features for point segmentation.

Since no scene classification is required, the classification output of the network from the original implementation is removed and the network is reduced to the point by point segmentation part only. The original implementation took as input a vector of size $n \times 3$ where n corresponds to the number of input points and 3 corresponds to the point dimensions (x, y, z) . In our version we extended the input dimensions to $n \times 10$ where the first 3 dimensions are (x, y, z) similar to the original implementation. The second 3 dimensions correspond to the spherical coordinates per point (ρ, ϕ, θ) . The inputs 7 through 10 are the Doppler velocity per point v_r , the reflection magnitude mag , the velocity of the ego vehicle v_s , and the turn rate of the ego vehicle yaw respectively.

The spherical coordinates are included to allow the network to learn connections between points that can be easier to represent in the spherical coordinate system. For example two points with the same azimuth and elevation would only differ in depth, such a relationship is linear and easier to learn, unlike in the cartesian system where this relationship is a function of the square root of the sum of squares.

An additional input transformation network is added to modify the spherical inputs similar to the cartesian inputs. The outputs of the transformation networks are then concatenated with the non-spatial inputs (v_r, mag, v_s, yaw) and an $n \times 10$ input vector is formed. The dimensions of the shared multi-layer perceptron layers are increased to take into consideration the increase in input dimensions and thus allow for better encoding of features. As a result, the feature transformation network dimensions are also increased.

A skip connection is introduced from the input until after the feature transform as can be seen in Fig. 2. The non-spatial inputs (v_r, mag, v_s, yaw) are appended to the $n \times 128$ feature vector thus increasing its dimensions to $n \times 132$. The reasoning behind this skip connection is to allow for a bigger influence of the non-spatial inputs as differentiating features. Due to the use of batch normalization for each mlp, no dropout was used. The output of the network was reduced to $n \times 1$ since we are tackling a binary classification problem (real vs ghost).

The most important features of the network are the transformation networks (T-Net) which learn different transformations and apply them to the input point clouds and the features. This helps the network generalize to different rotations and orien-

tations of point cloud features. The skip connection allows for a bigger influence of the inputs on the final classification output. Then the global feature vector is appended to the local features to use both local and global features in the point feature extraction and ghost target detection as can be seen in Fig. 2.

B. Dataset

The data used is measured by an Astyx 6455 HiRes radar sensor, a sample of which is provided by Meyer et al. [17]. The radar has a 110 degrees by 10 degrees horizontal by vertical field of view respectively, and a range of 100 meters. For reference, a 64 beam lidar sensor is provided. The dataset comprises of 9,321 frames divided over 5 measurement sequences each recorded on different roads. The sequences cover various scenarios and include both urban and highway environments, as well as open roads and tunnels.

Each measurement sequence contains the radar data (x, y, z, v_r, mag) , the lidar measurements, and vehicle state measurements such as the speedometer value v_s and the turn rate yaw .

Astyx 6455 HiRes radar sensor measures around 1000 3D points per frame on average, as opposed to 100 2D points per radar frame provided by other types of series production radar [17].

C. Ground Truth Generation

Since there are not many public datasets for automotive radar, and because the existing datasets do not label ghost targets and rather only have labels for visible objects [17], [18] or ego motion ground truth [19], [20], it is very challenging to obtain labeled data for the ghost target detection problem. Taking into account that manually labeling thousands of frames for ghost targets is very time consuming and error prone due to the complexity of driving scenarios, we devise an approach to generate the required annotations to our unlabeled dataset.

As lidar is a directed time of flight sensor, it is much less susceptible to the reflections problem; that is the light being reflected off multiple surfaces before returning to the lidar. Thus it can be used to reliably measure objects and surfaces around the ego vehicle.

We propose to use lidar as a reference for automatic labeling of the radar data to be used as ground truth for training the deep neural network. We project the lidar points to the 2D image space and use depth completion based on Ku et al. [21] to obtain a dense depth map of the scene. We then project the radar data to 2D image space and compare with the lidar data as follows: if the depth value of the radar point/pixel is within a certain threshold of a lidar depth pixel, it is considered to be real, however if the radar depth value is outside the specified threshold, or no corresponding lidar data exists, it is then considered to be a ghost target. The algorithm is summarized in Fig. 3. Fig 4 shows a sample output of the labeling algorithm.

The concept stems from the fact that since ghost targets are measured over an indirect reflection path, their depth

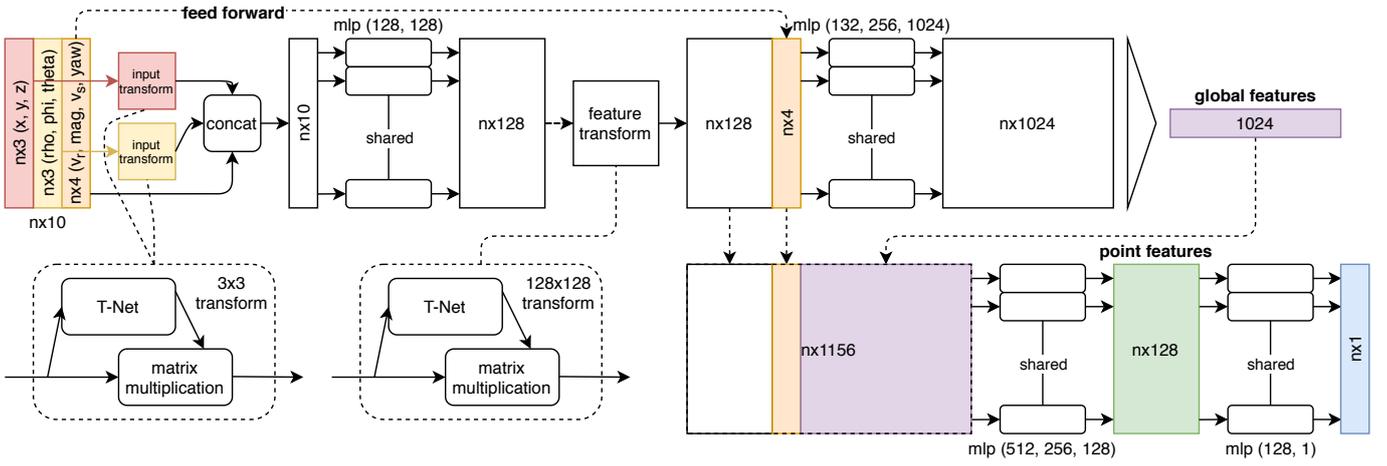


Fig. 2. The architecture of the deep neural network based on PointNet. The input is a ten dimensional vector of points split into 3 groups (x, y, z), (ρ, ϕ, θ), and (v_r, mag, v_s, yaw). The first 2 groups are spatial features in cartesian and spherical coordinates passed through an input transform network, while the non-spatial features are forwarded down the network. The output is a vector of per-point binary classification as real or ghost targets.

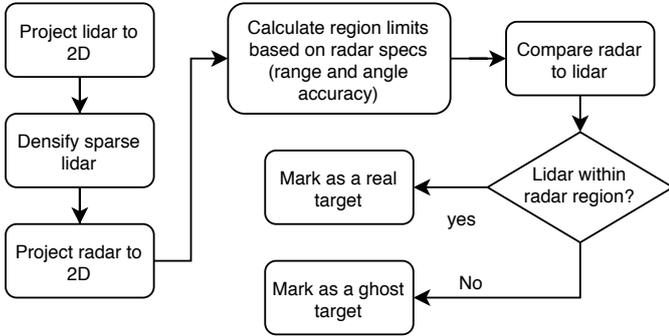


Fig. 3. A flowchart summarizing the main steps for the automatic labeling approach suggested.

value will place them behind the obstacle that causes the last reflection. The threshold is chosen based on the sum of errors expected in the radar and lidar measurements and on the maximum resolutions in azimuth and range. The error value was estimated to be between 0.2 to 1.8 meters in the 5 to 50 meters range, thus a fixed threshold value of 1 meter is chosen for the selection of real and ghost targets.

In our experiments, we manually labeled a selection of frames from multiple sequences and compared the output of the algorithm to the manually labeled frames, the resulting accuracy was 93%.

IV. EXPERIMENTS AND RESULTS

We trained a deep neural network to detect ghost targets in 3D radar point clouds. Data annotation for the radar data is done by our suggested labeling approach using lidar data as reference. The evaluation of the network is done using cross validation where the data is split into 4 parts and the network is trained 4 times each with different split and then results are averaged.

A. Implementation and Training

The network has been trained and evaluated using cross validation. Since the dataset is provided in different sequence measurements, in each training run, one sequence is used for validation while the rest of the sequences are used for training. The accuracy results are calculated based on the weighted average of all the sequences since they have different lengths. The inputs are normalized to zero mean and unit standard deviation over the whole training set, and the values from the training set are used for the normalization of the validation set.

The input to the network was fixed to 2500 radar points chosen randomly and with repetition from each frame. The network was trained using a batch size of 32, over 250 epochs, and using a learning rate of 10^{-3} reduced by half every 125 epochs.

To tackle imbalances between the classes, we used the algorithm proposed by Cui et al. [22] for a class-balanced loss based on the effective number of samples and calculated the loss with a $\beta = 0.999$. The algorithm uses a weighting scheme that utilizes the effective number of sample per class to re-balance the loss.

To evaluate the effectiveness of the changes introduced, we do an ablation study on different network architectures with the unmodified PointNet as a baseline. A PointNet with the inputs extended to a 5 dimensional (x, y, z, v_r, mag) point vector is trained and all subsequent changes are compared to the results of this network.

We tested three modifications of the original PointNet architecture

- A network with $n \times 10$ input. This network show the importance of using additional input features.
- A network with $n \times 10$ input and a skip connection. This is the network architecture presented in Fig. 2 and evaluates the usefulness of the skip connection.

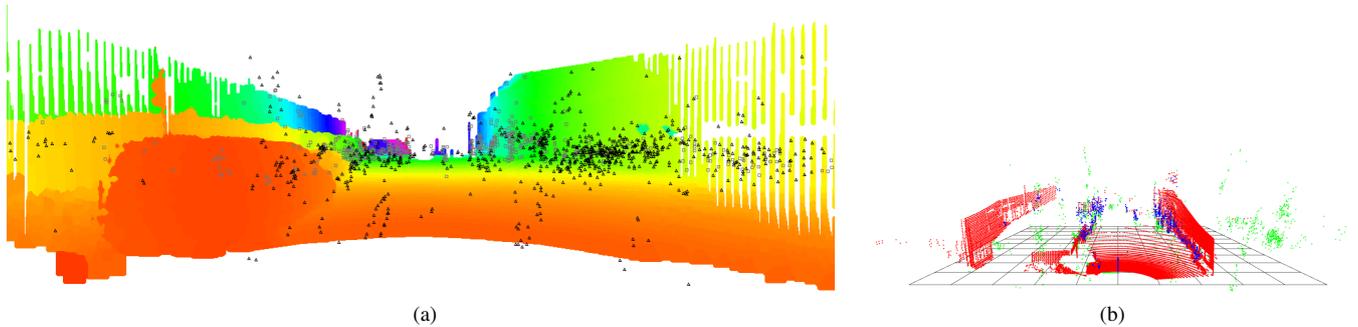


Fig. 4. (4a) The dense depth map calculated based on the lidar data with an overlay of a sample of radar points. The light-colored squares correspond to radar points labeled as "real" based on their depth value, conversely the dark-colored triangles correspond to the points labeled as "ghost". (4b) The 3D perspective showing the points labeled as "real" in blue, coinciding with surfaces detected by the lidar. The "ghost" points are in green and can be seen behind lidar surfaces or missing a reference lidar measurement.

- A network with $n \times 7$ input and a skip connection. In this architecture we removed the spherical coordinates input to evaluate their impact on the overall result.

B. Results

Evaluating our results compared to other approaches was not possible due to the following:

- our approach is tackling 3D data, while the other approaches are tackling 2D data [8], [9].
- the density complexity of the data used in our work makes it a not possible to be reduced to 2D.
- due to the unavailability of the data being used in other approaches, it was not possible to evaluate it on our approach.

For the evaluation of the results, we used the intersection over union IoU metric.

$$IoU = \frac{TP}{TP + FP + FN} \quad (1)$$

where TP is the true positives of a class, FP and FN are the false positives and false negatives respectively. The overall performance is then the mean intersection over union $mIoU$. Using those metrics allows us to both evaluate the per class and overall performance of the network.

As seen in table I, the added number of the input features had a significant impact on the output of the classification. The addition of a skip connection slightly improved the results. And the inclusion of the spherical spatial features improved the results as well. We can attribute the biggest improvement to the vehicle velocity v_s and the vehicle turn rate yaw . The detection of real data benefited more from the implemented changes, this can indicate that they are easier to learn than the ghost targets.

The high contribution of the vehicle information is expected due to the compensation effects that can be learned from the inclusion of those features.

During the training, we noticed that some data sequences were more challenging than others, and in some cases there was less similarity between the validation and training data. We are interested in applying our proposed architecture to more diverse data.

Fig. 5 shows some results of the detection algorithm on validation data in comparison to the lidar data as reference. A qualitative analysis shows the effectiveness of the algorithm in detecting ghost targets in complex reflection scenarios such as in tunnels as seen in Fig. 5c and 5d.

As can be seen in the data, the radar points exhibit a column like distribution visible in Fig. 4, 5a, and 5c, This might be caused by the degraded resolution in the elevation direction, or an artifact of the signal processing. It is worth noting that the network was also successful at removing those column structures since they do not represent a physical structure and thus have no lidar reflections.

In our observations we noticed that the network is effective at removing ghost targets caused by the ground and thus they fall below the ground level. We also observed that it is more successful at detecting ghost targets that fall behind bigger surfaces such as walls, busses, or transport vehicles. Since the data has more real targets than ghost targets, we expect the prediction accuracy to improve the more balanced the data is, and the more data is used.

V. CONCLUSION

In this paper, we presented the first point based deep neural network for classification of real and ghost targets in 3D radar data. We extended the PointNet architecture with expanded inputs and input forwarding to make it more suitable for the radar detection problem. We also presented an approach for automatic radar data labeling using lidar data. The work has shown promising results in complex real measurement scenarios. We achieved a mean detection score of 65%, a significant improvement over the baseline, and evaluated the effect of different modifications over the original network. It would be interesting to investigate further changes to the architecture. Data from previous time steps could be used to add temporal information, deeper and more complex network architecture can help reveal more relationships between the data, and improved ego vehicle information could lead to more reliable features. To investigate the suggested improvements more data is needed to train the neural networks while higher radar data frequency could also prove beneficial as well.

TABLE I
NETWORK CLASSIFICATION PERFORMANCE

Network	mIoU	IoU Ghost	IoU Real	F1 Ghost
Baseline	61.41%	55.91%	66.90%	71.72%
$n \times 10$ input	65.13%	58.53%	71.73%	73.84%
$n \times 10$ input with skip	65.38%	58.63%	72.13%	73.92%
$n \times 7$ input with skip	64.52%	57.76%	71.29%	73.23%

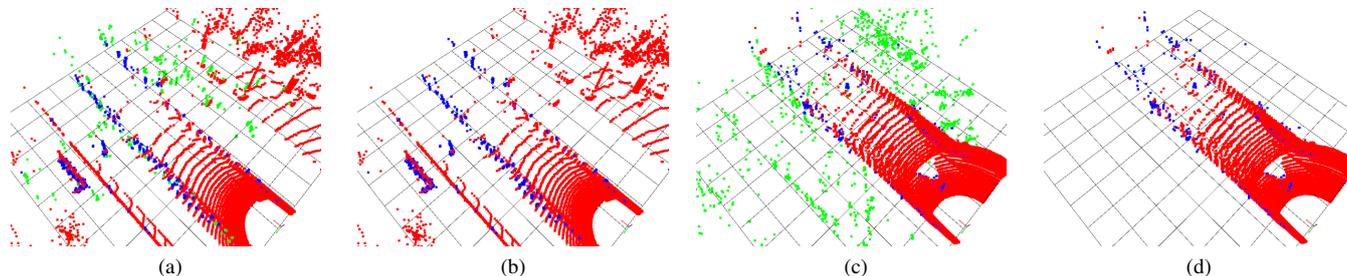


Fig. 5. The network output shows real targets (in blue) and ghost targets (in green), lidar data is shown for reference (in red). (b) and (d) show the point cloud after removing the detected ghost targets in (a) and (c) respectively.

REFERENCES

- [1] H. Winner, *Automotive RADAR*. Cham: Springer International Publishing, 2016, pp. 325–403.
- [2] C. Wolff, “Book 1 “radar basics”,” Dec 2009. [Online]. Available: <https://www.radartutorial.eu/druck/Book1.pdf>
- [3] F. Roos, M. Sadeghi, J. Bechter, N. Appenrodt, J. Dickmann, and C. Waldschmidt, “Ghost target identification by analysis of the doppler distribution in automotive scenarios,” in *2017 18th International Radar Symposium (IRS)*. IEEE, 2017, pp. 1–9.
- [4] P. Setlur, M. Amin, and F. Ahmad, “Multipath model and exploitation in through-the-wall and urban radar sensing,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 10, pp. 4021–4034, 2011.
- [5] P. Setlur, G. Alli, and L. Nuzzo, “Multipath exploitation in through-wall radar imaging via point spread functions,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4571–4586, 2013.
- [6] M. Leigsnering, F. Ahmad, M. Amin, and A. Zoubir, “Multipath exploitation in through-the-wall radar imaging using sparse reconstruction,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 2, pp. 920–939, 2014.
- [7] I.-h. Ryu, I. Won, and J. Kwon, “Detecting ghost targets using multilayer perceptron in multiple-target tracking,” *Symmetry*, vol. 10, no. 1, p. 16, 2018.
- [8] R. Prophet, J. Martinez, J.-C. F. Michel, R. Ebel, I. Weber, and M. Vossiek, “Instantaneous ghost detection identification in automotive scenarios,” in *2019 IEEE Radar Conference (RadarConf)*. IEEE, 2019, pp. 1–6.
- [9] J. M. Garcia, R. Prophet, J. C. F. Michel, R. Ebel, M. Vossiek, and I. Weber, “Identification of ghost moving detections in automotive scenarios with deep learning,” in *2019 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*. IEEE, 2019, pp. 1–4.
- [10] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.
- [11] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “Pointnet++: Deep hierarchical feature learning on point sets in a metric space,” in *Advances in neural information processing systems*, 2017, pp. 5099–5108.
- [12] Y. Zhou and O. Tuzel, “Voxelnet: End-to-end learning for point cloud based 3d object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4490–4499.
- [13] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, “Pointpillars: Fast encoders for object detection from point clouds,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 697–12 705.
- [14] O. Schumann, M. Hahn, J. Dickmann, and C. Wöhler, “Semantic segmentation on radar point clouds,” in *2018 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 2179–2186.
- [15] A. Danzer, T. Griebel, M. Bach, and K. Dietmayer, “2d car detection in radar data with pointnets,” in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 61–66.
- [16] N. Scheiner, F. Kraus, F. Wei, B. Phan, F. Mannan, N. Appenrodt, W. Ritter, J. Dickmann, K. Dietmayer, B. Sick *et al.*, “Seeing around street corners: Non-line-of-sight detection and tracking in-the-wild using doppler radar,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2068–2077.
- [17] M. Meyer and G. Kusch, “Automotive radar dataset for deep learning based 3d object detection,” in *2019 16th European Radar Conference (EuRAD)*. IEEE, 2019, pp. 129–132.
- [18] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, “nusscenes: A multimodal dataset for autonomous driving,” *arXiv preprint arXiv:1903.11027*, 2019.
- [19] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, “1 Year, 1000km: The Oxford RobotCar Dataset,” *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017. [Online]. Available: <http://dx.doi.org/10.1177/0278364916679498>
- [20] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, “The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset,” *arXiv preprint arXiv: 1909.01300*, 2019. [Online]. Available: <https://arxiv.org/pdf/1909.01300>
- [21] J. Ku, A. Harakeh, and S. L. Waslander, “In defense of classical image processing: Fast depth completion on the cpu,” in *2018 15th Conference on Computer and Robot Vision (CRV)*. IEEE, 2018, pp. 16–22.
- [22] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, “Class-balanced loss based on effective number of samples,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9268–9277.