Radar Dynamic Object and Camera Detection Fusion

Junnan Shimizu, Aaron Kingery, Tiare Delgado, Dezhen Song, Jason O'Kane

Abstract—**Radar and camera sensors when implemented independently have strengths of their own but noticeable limitations. Radar sensors can measure the position and radial velocity of points on objects, but cannot accurately classify them. On the other hand, camera sensor data can be used to accurately classify objects, but cannot be used to accurately determine their position. Through radar-camera fusion, the strengths of each sensor can fill the limitations of the other, allowing for a comprehensive imaging of the environment and its dynamic objects and their corresponding attributes. In this paper, we explore this sensor fusion by processing the radar data through a grid-based DBSCAN clustering algorithm and processing the camera data through the YOLOv4 real-time object detection model to determine the objects' position, radial velocity, and class. Then, we aim to produce a real-time stream of information of dynamic objects and their attributes that is intended for an autonomous driving decision making context, but can be applied in others as well.**

I. Introduction

To make informed decisions that yield positive results, having the proper information to your disposal is necessary. For example, in the context of autonomous vehicles, collecting the position of surrounding vehicles and objects is necessary for safe and accurate decision-making. This collection and processing of data varies based on the context — some sensors and data processing algorithms are more valuable in certain contexts over others. For this project, we utilize radar and camera sensors to collect data about the environment and process it to detect dynamic objects and determine their attributes such as their class (i.e. cars, motorcycles, pedestrians), radial velocity, and position in hopes of providing valuable information later in the autonomous decision-making process. Radar and camera sensors were chosen specifically because radar sensors can determine the positions and radial velocity of objects while camera sensor data can be used to classify objects. Through radar-camera fusion, the strengths of each sensor can fill the limitations of the other, allowing for a comprehensive imaging of the environment and its objects. Through the clustering of radar points and the classification of objects using an object-detection machine learning model of the camera data, we hope to associate dynamic objects found in the radar data and image detections found in the camera data, and then output a package of information about the environment and the objects within in. The contributions of this paper are found in the processing of data within the autonomous decision-making process.

National Science Foundation

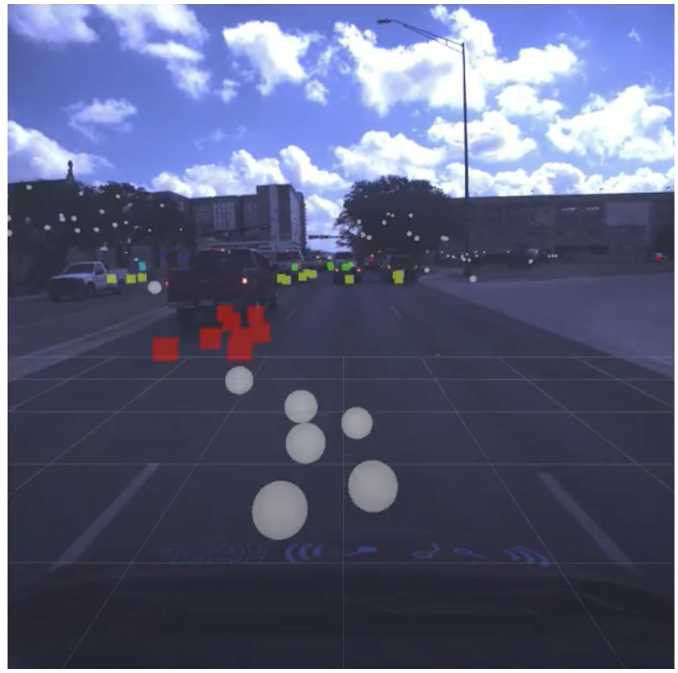

Fig. 1. The visualization of processed and clustered radar data overlaid with camera data. Static points (grey points) and dynamic points (colored points) are seen and colors correlate with their calculated cluster.

II. Related Work

Radar technology improves substantially every year, and its development is only accelerated with the rapid growth of the autonomous and semi-autonomous vehicles industry [1] [13]. This growth has made radar systems smaller and cheaper, making it a very appealing option [10]. Mass market radars (MMRs) and high end radars (HERs) are in consideration for this project. Most modern vehicles use MMRs as it is cost effective, however HERs find benefits mainly in its ability to combine Dopper or micro-Doppler measurements with spatial resolution, as well as a higher resolution and greater dynamic range [1]. Initially, radar implementation in cars was used for crash avoidance, but quickly transitioned to functions such as: adaptive cruise control (ACC) [3], automatic emergency break (AEB), blind spot detection (BSD), and lane change assist (LCA) [13] with the aid of object detection and classification [5]. Radar has also been implemented in electronic equipment for the military (radar guidance for missiles, gunfire control, tracking of low-altitude targets) [15] as well as weather forecasting, environmental monitoring, and astronomical, atmospheric physics, ionospheric research, [7]

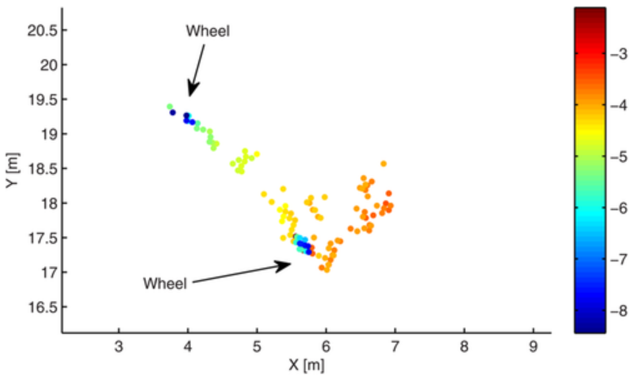

Fig. 2. Image of a passenger car, cropped from a high-resolution radar image. The color indicates the radial velocity in m/s. Image pulled from [1]

intelligent robots, gesture recognition, biological sign recognition, [15], surveillance, among others [6]. It is important to note that with this increase of vehicles with some having up to 12 radar sensors, interference between radar sensors is more likely causing moments of unreliable sensor measurement accuracy and sensor blindness [13]; however otherwise, it is advantageous compared to other imaging technologies because of its ability to function properly in low light and severe weather conditions [13] [12] [7]. Because of the reflection of radar off the road surface, vehicles will appear as laterally and longitudinally extended objects and the bottom side of vehicles can be seen which makes determining the class of the vehicle (cars, trucks, motorcycles, etc.) relatively easy compared to other imaging systems [5]. With the development of high resolution radar, a point-source assumption is not possible because an object will have more than one reflection, and therefore a clustering algorithm is required to track and determine the class of an object [3]. The most common clustering algorithm utilizes k-means which requires a predetermined number of clusters which is not valuable in an automotive scenario. On the contrary, DBSCAN is a density-based clustering method. The algorithm is given a threshold for the density of points within a certain degree, and the number of points must exceed for an object to be detected [5]. For this project, YOLOv3 or YOLOv4 will be utilized for clustering and tracking of objects [11] which avoids the issues experienced in [3].

The two main options to interpret the incoming radar data include synthetic aperture radar (SAR) and grid mapping. Within grid maps are amplitude and occupancy grid maps which are both useful for determining obstacles; however, occupancy grid maps provide a distinction between free space, obstacles, and unknown areas [13], so for this project, it is the most appropriate as it is highly transferable to an automotive context.

Another advantage that radar has over other imaging technologies is its ability to measure radial velocity via Doppler measurement. Examining this sample data of the radial velocity and its spatial data in Figure 2, the radial velocity of the points gradually increase from left to right with two exceptions. Given these unique clusters in the data, it allows for easier object identification, in this case, identifying the wheels of a vehicle [1]. Determining the full yaw rate, longitudinal, and lateral speed is possible as seen in [4] potentially allowing for more accurate radar object detection as well.

In [13], by utilizing an analyze-before-measure approach, the receiver filters out the data depending on the operational parameters such as transmit direction or center frequency [13]. Within this project, data will be filtered using YOLOv4 to find the regions of interest (ROI) [11] [7] which will reduce the amount of data to process and therefore reduce processing times for dynamic object detection. By utilizing the YOLO algorithm, model classification performance saw great improvements. It is important to note that deep learning models with limited training data sets like radar samples will result in many over shooting scenarios and that is why there is great value in running a deep learning algorithm on camera data [7]. YOLO has also seen applications in similar veins such as the detection of marine ships. In [8], they fused LiraNet and YOLO and found strong performance benefits [8] and we hope to find those same benefits in radar-YOLO fusion.

In [4], to detect moving objects and estimate their velocity, the project compared the optical flow of two consecutive frames. For points with no known depth, two or more consecutive frames are fused using Kalman filters [4]. For this project, we will use radar's own radial velocity measurements. If time persists, actual object velocity will be calculated using processes seen in [4].

However, radar has its limitations. Although it has relatively high distance resolution [7] [12], it has low direction resolution and lower resolution in terms of orientation (azimuth/elevation) [15] [10] which can make detecting the objects' vertical position difficult [12]. Also, millimeter wave radar (MMW) is affected by electromagnetic waves and the detection is sparse so physical characteristics like sharp angles or curves are not easy, and in an automotive context, this means no line detection or easily detecting pedestrians [15]. On the contrary, a camera is able to fill the gaps that are left with radar as a camera provides high spatial resolution [12] with similar characteristics to human visual perception [14], but has poor depth perception. A fusion of these sensors fill each others shortcomings. For radar and camera fusion, the calibration of their locations is necessary [12]. For this project, a manual calibration of the radar and camera sensors will be established, but for future iterations, a dynamic calibration for the sensors will be implemented. In [12], their proposed method for calibration is to estimate the homography or the origin of data sets given by the radar and the camera planes. Many research projects for 3D object detection utilize the fusion of LiDAR and camera sensors [9]; however, LiDAR and camera still fail in high noise environments such as poor weather or fog [14]. In [9], they utilize CenterNet, a 3D object detection deep network with the use of a singular camera, and later fuse it with radar data called CenterFusion [9]. For this research, object detection will solely be done with the camera data and distance and velocity measurements will be done with the radar data. Another alternative for object detection, object size, and radial distance is by utilizing a multiple-camera system outlined in. This system fused with radar can obtain a fairly accurate 3D space. One thing to note from [10] is that a fixed y-coordinate can cause messy data and a rejection algorithm may be necessary for data filtration. Furthermore, when using multiple sensors, an efficient fusion rule is important to maintain the quality of the data from both sources [10]. In [2], they utilize a middle-fusion approach meaning that each set of sensor data is independently processed and merged later. From my understanding, [2] utilizes a similar process to [11], however [2] does not cluster the radar points and treats every radar point an an independent object detection. With all this, we hope to create a stream of valuable information in an autonomous vehicle decision-making process.

III. Problem Statement

The capturing of data is a crucial step in the autonomous decision-making process and sensors of all kinds can be used, each having specific strengths and weaknesses depending on the context. However, the processing of the data into usable information is just as important as the collecting of data — the processing step converts the data into a format that can be actually be read and interpreted later in the decision making process.

For this project, we consider a vehicle equipped with camera and radar sensors collecting data within a dynamically changing environment with multiple dynamic objects within the sensors field of view and range. In sample data collections, the car was driven around streets in College Station, Texas. The radar data is collected in the format: x, y, z, radar cross section (RCS), signal-to-noise ratio (SNR), and radial velocity. The x, y, and, z are the positional coordinates of each point of measurement (however, the z value is set to a constant value depending on its position on the test vehicle), the RCS is the measure of how detectable an object is by radar — a larger RCS measurement denotes that an object is more easily detectable versus a small value measurement, SNR is the ratio of the power of the signal and the power of the background noise, and the radial velocity is the change of position of a point over time relative to the position of the sensor. The camera data is a collection of pixels in a grid, each pixel having a different value to determine its color — simply collecting color information of the environment

With each stream of data, we process it to determine the dynamic objects and their corresponding position, radial velocity and class, and return a package of information for each frame that includes a list of dynamic objects and their attributes. Ideally, with a successfully running program, we hope to accurately detect dynamic objects and their class, position, and radial velocity, and this can be measured by comparing object detection and classification with human perception, and position and radial velocity with real-world measurements. With this information, we hope to streamline the data collection and data processing steps in an autonomous decision making model.

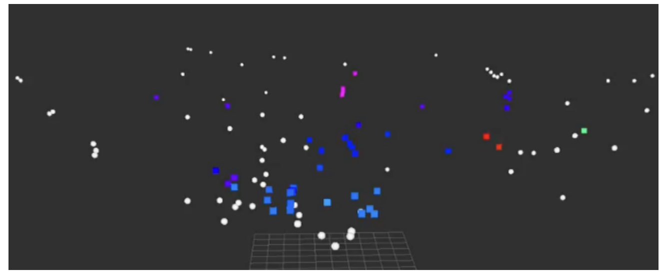

Fig. 3. Example radar data displayed in a point cloud. Square, colorful points represent dynamic objects, and the color temperature represents radial velocity.

IV. Algorithm Description

Because radar data is simply a collection of points, in order to process it to identify objects, we must utilize a clustering algorithm to associate objects with the points that are measuring it. However, there is a non-equidistant sampling density for radar data because the radio waves of the radar are sourced from a single point, therefore a clustering algorithm that takes this into account as well as the clutter and noise that comes with radar data must be used. For this project, we will use a grid-based density-based clustering algorithm seen in [5]. For this to function properly, the position of radar data formatted in a Cartesian coordinate system (x, y, z), must be converted to range distance/direction (r) and azimuth distance/direction (θ) using the formula: $(r, \theta) = (x^2 + y^2, atan2(y, x))$. In the grid-based DBScan implementation we used, the Cartesian points must be converted to gate and beam units. In order to do so from r and θ, the two following values need to be set: r resolution (rr) and θ resolution (θr). Then the following equation can be used: $Gate, Beam = (r/rr, \theta/\theta r)$. For most of our testing, rr and θr were set to 120 and 0.05 respectively. These values were chosen because they yielded positive results, although it is important to note other values were tested and yielded similar results as well. Continuing, each cluster of points will represent an object within the frame. In order to find the position and radial velocity of the singular object, we will average out the x, y, z, and radial velocities values found among all the points in an object cluster deeming that cluster to be an object of average position and radial velocity. For visualization purposes, we created a ROS2 node that takes the radar data as an input and runs the points through the clustering algorithm. Once every dynamic point within the frame is clustered, we add that cluster value as an attribute to each point and create a new point cloud that is fed into a tf2-ros node, allowing for the point cloud to be projected onto the camera image.

For the camera sensor data, first we had to feed the camera data through an image-proc node to eliminate the natural image distortion in the camera data and convert image data to color. Then, we used YOLOv4, a real-time object detection system, to determine the class of the objects in each frame. YOLOv4 is a strong contender for this project because it is a real-time object detection software that can be trained for our specific needs, though we did not train the model ourselves for

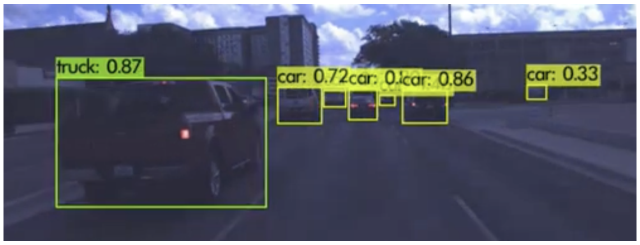

Fig. 4. YOLOv4 working with the collected camera data

this project (yet). YOLO uses a neural network that is trained to detect certain objects, and it will divide an image or frame into regions and predicted bounding boxes for objects seen in Figure 4.

The next step is to associate the radar dynamic objects with the image detections. To do this, we will overlay the radar points with the bounding boxes of the object detections, making sure to align the radar points at the same angle and calibration at which the camera is obtaining data. For points found in the overlap of bounding boxes, we will associate those points with their respective clusters.

After associating the dynamic objects, we will create a ROS2 node that publishes this information for every frame of data in a simple package of information that includes an object identifier, class, position, and radial velocity that hopefully will streamline the steps prior to the autonomous decision-making algorithm.

V. Results

For the radar collection and processing, the grid-based DBScan clustering algorithm failed to process the data as fast as it was coming in, which creates the tendency to skip frames of data; however, even with some frame skips, the clustering was in real time and works fast enough for our general purposes in this project, though for future work this should be ironed out. Furthermore, the clustering is not very reliable in more complex scenarios. When faced with environments with a large number of dynamic objects in frame, the clustering algorithm tended to cluster objects close to each other as one large object as seen in Figure 5 — the green cluster in the center is actually made up of three cars, for example; however, it is important to note that in simpler scenarios, the clustering is fairly accurate.

For the camera collection and processing, YOLOv4 can detect objects in real-time with relatively high accuracy. There are a few instances of incorrect classifications in our testing, however consistent classification errors are rare. One issue that we ran into was a hardware limitation — for real-time object detection, YOLO requires a graphics card with CUDA cores. We were able to obtain a computer with the required hardware, but there may be other ways to bypass this problem such as reducing the density of data collected by the camera or retraining the model to detect fewer objects and simplifying the model, however both were not explored in this project. This may be important if there are hardware limitations for the

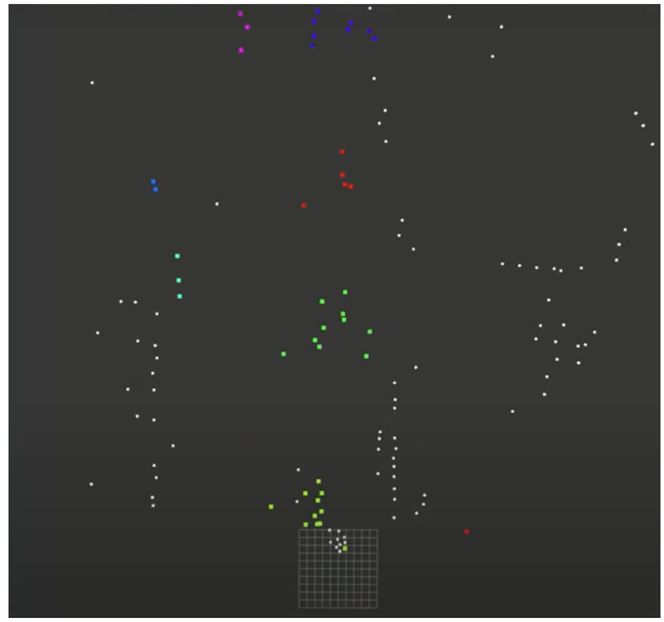

Fig. 5. Top-down view of radar point cloud run through the clustering algorithm. Each color corresponds to a cluster and white points are static points.

autonomous vehicle we are using.The YOLO model seems to work strongly in this context by providing a real-time stream of object detections with minimal error which is important in a high-risk context such as autonomous driving.

Unfortunately, the association of radar dynamic objects and image detections was not able to be implemented with time constraints. This also means that the publishing of the packages of information has yet to be completed, however these are the next steps in this research.

VI. Conclusion

The collection and processing of data is an important step in the decision-making process, and depending on the context, certain sensors and data processing algorithms are more valuable than others. For this project, we believed a radar and camera sensor fusion would yield the best results through clustering and object-detection, and therefore obtain the data and process it into a format that is of great value later in the autonomous decision-making process. Unfortunately, there are still a couple limitations in accuracy with our implementation. For example, the algorithm used for the clustering of radar data is not as efficient as we need for real-time streams of data — the lag seen in our testing may be a problem for later stages of the decision-making process, although we cannot say for sure yet. Furthermore, the algorithm tends to cluster separate objects that are side by side as one large object, and we saw that it is difficult for low resolution objects in the distance to be clustered correctly. As for the object detection, YOLOv4 with its provided dataset is an expensive process and requires a graphics card with CUDA cores for real-time object detection which limits the hardware that can be used; however, this can be combated by training the model specific for our context which can yield more accurate results as well as reduce the processing power required by the model.

Finishing the intended end of this research project — the implementation of the radar dynamic object and image detection association and the publishing of this information — is the next step for this research. Continuing beyond this project can include different clustering algorithms for radar involving the use of radial velocity measurements rather than just x, y, z positions to reduce the over-clustering errors we seen in more complex environments, in addition to more efficient clustering algorithms that eliminate the lag seen in this project. Also, the training of YOLO with targeted data sets can be another step for yielding more accurate results and at the same time reducing the processing power required by the YOLO object-detection model. Lastly, collecting more dynamic object attributes such as object velocity, position overtime, object color, object dimension, just to name a few. With further implementation, we hope to completely streamline the data collection and processing steps, providing a comprehensive set of information that is valuable for the autonomous decision-making model, hopefully resulting in accurate and safe autonomous decision-making.

References

[1] Stefan Brisken, Florian Ruf, and Felix Höhne. Recent evolution of automotive imaging radar and its information content. *IET Radar, Sonar & Navigation*, 12(10):1078–1081, 2018.

[2] Siyang Han, Xiao Wang, Linhai Xu, Hongbin Sun, and Nanning Zheng. Frontal object perception for intelligent vehicles based on radar and camera fusion. In *2016 35th Chinese Control Conference (CCC)*, pages 4003–4008, 2016.

[3] Dominik Kellner, M. Barjenbruch, K. Dietmayer, Jens Klappstein, and Juergen Dickmann. Instantaneous lateral velocity estimation of a vehicle using doppler radar. pages 877–884, 01 2013.

[4] Dominik Kellner, Michael Barjenbruch, Jens Klappstein, Juergen Dickmann, and Klaus Dietmayer. Instantaneous full-motion estimation of arbitrary objects using dual doppler radar. pages 324–329, 06 2014.

[5] Dominik Kellner, Jens Klappstein, and Klaus Dietmayer. Grid-based dbscan for clustering extended objects in radar data. In *2012 IEEE Intelligent Vehicles Symposium*, pages 365–370, 2012.

[6] Du Yong Kim and Moongu Jeon. Data fusion of radar and image measurements for multi-object tracking via kalman filtering. *Information Sciences*, 278:641–652, 2014.

[7] Tong Lin, Xin Chen, Xiao Tang, Ling He, Song He, and Qiaolin Hu. Radar spectral maps classification based on deep learning. In *Proceedings of the 2020 International Conference on Computer Communication and Information Systems*, CCCIS 2020, page 29–33, New York, NY, USA, 2020. Association for Computing Machinery.

[8] Zhou Long, Wei Suyuan, Cui Zhongma, Fang Jiaqi, Yang Xiaoting, and Ding Wei. Lira-yolo: A lightweight model for ship detection in radar images. *Journal of Systems Engineering and Electronics*, 31(5):950–956, 2020.

[9] Ramin Nabati and Hairong Qi. CenterFusion: Center-based radar and camera fusion for 3d object detection. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, jan 2021.

[10] Felix Nobis, Maximilian Geisslinger, Markus Weber, Johannes Betz, and Markus Lienkamp. A deep learning-based radar and camera sensor fusion architecture for object detection. In *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–7, 2019.

[11] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement, 2018.

[12] S. Sugimoto, H. Tateda, H. Takahashi, and M. Okutomi. Obstacle detection using millimeter-wave radar and its visualization on image sequence. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 342–345 Vol.3, 2004.

[13] Christian Waldschmidt, Juergen Hasch, and Wolfgang Menzel. Automotive radar — from first efforts to future systems. *IEEE Journal of Microwaves*, 1(1):135–148, 2021.

[14] Zizhang Wu, Guilian Chen, Yuanzhu Gan, Lei Wang, and Jian Pu. Mvfusion: Multi-view 3d object detection with semantic-aligned radar and camera fusion, 2023.

[15] Yong Zhou, Yanyan Dong, Fujin Hou, and Jianqing Wu. Review on millimeter-wave radar and camera fusion technology. *Sustainability*, 14(9), 2022.