ISSN: 2230-9926

Available online at http://www.journalijdr.com

IJDR

International Journal of Development Research
Vol. 14, Issue, 03, pp. 65263-65265, March, 2024
https://doi.org/10.37118/ijdr.27708.03.2024

RESEARCH ARTICLE

OPEN ACCESS

STREAMLINING DATA PROCESSING FOR SAFER AUTONOMOUS DRIVING: A SENSOR PIPELINE APPRAOCH

***Ashish Tiwari and Ankur Sarkar**

Capgemini Engineering, Bangalore, India

ARTICLE INFO

Article History:
Received 16th January, 2024
Received in revised form
19th January, 2024
Accepted 06th February, 2024
Published online 30th March, 2024

Key Words:
Sensor Data, Data Pipeline, Sensor Data Processing, Adas, Lidar, Camera.

****Corresponding author: Ashish Tiwari,***

ABSTRACT

Autonomous vehicles have become the crux of today's innovation and research. There are so many unidentified problems in the real-world scenario that the solutions can be endless. Many such companies who are marching on the way to define a perfect autonomous driving vehicle have also achieved level 3 or 4 of the SAE Driving Automation standard. Autonomous vehicles mainly rely on sensors which act as their senses to find their way and locate themselves in the environment and through traffic and various types of objects. The positioning of the sensors, their region and range of vision, their flexibility to work in different environmental conditions and finally how they are put together in the pipeline poses a critical and vital concern. This paper imparts an understanding about the sensor data and its pipeline in the spectrum of autonomous vehicle.

Citation: ***Ashish Tiwari and Ankur Sarkar. 2024.*** "Streamlining data processing for safer autonomous driving: A sensor pipeline appraoch", *International Journal of Development Research*, 14, (03), 65263-65265.

INTRODUCTION

In autonomous vehicles, sensors play a crucial role in perceiving the surrounding environment and gathering the necessary data for safe and efficient driving. The most important sensors used in autonomous vehicles are cameras, LiDAR (Light Detection and Ranging), ultrasonic sensors and radar. Each sensor provides unique capabilities and together they create a comprehensive perception system for the vehicle. There can be multiple radars, LiDARs, cameras and ultrasonic sensors. Also, the visual field of LiDAR, radar, and cameras complement each other by providing different types of information about the surrounding environment.

Camera: Cameras are optical sensors that capture images and video of the vehicle's surroundings. They are typically placed strategically around the vehicle to provide a wide field of view. Cameras heavily rely on visible light, making them susceptible to variations in lighting conditions. Poor lighting, glare, or shadows can impact their performance, potentially leading to difficulties in object detection or recognition. Adverse weather conditions like heavy rain, fog, or snow can impair the camera's visibility and degrade image quality, making it challenging to extract meaningful information from the captured images. Cameras produce a large amount of data that needs to be processed in real-time. Advanced computer vision algorithms and significant computational resources are required for efficient object detection, recognition, and scene understanding.

Lidar: LiDAR or Light Detection and Ranging sensors (360 degrees) emit laser pulses and measure the time it takes for the light to bounce back after hitting an object. This enables the vehicle to accurately perceive distances and shapes of objects, even in low light conditions. Its main advantages include accurately estimating the distance to objects, allowing for reliable depth perception. It generates dense point cloud data that represents the shape and structure of objects in the environment for tasks related to autonomous navigation and mapping, obstacle detection, obstacle avoidance, and path planning. But they have limitations from adverse weather conditions, such as heavy rain, fog, or snow. The laser beams can scatter or get absorbed by the weather elements, leading to reduced detection range or degraded data quality. While LiDAR can provide accurate measurements within its range, it has limitations in detecting distant objects. Long-range detection typically requires more powerful and expensive LiDAR sensors. LiDAR may struggle with accurate object detection and localization in scenarios where objects are partially or completely occluded, or when there are reflective surfaces that can cause multiple reflections or blooming effects in the data.

Radar: Radar sensors use radio waves to detect objects and determine their position and velocity. They emit radio frequency signals and analyze the reflected signals to calculate the distance, speed, and direction of objects in the vehicle's vicinity, which is important for predicting object trajectories and anticipating potential collisions. Radar complements LiDAR and cameras in several ways. Radar has a longer detection range compared to LiDAR and cameras, allowing it to detect objects that are far away or obstructed by other obstacles. This makes radar particularly useful for detecting vehicles on highways or in adverse weather conditions. It is less affected by environmental factors like rain, fog, or darkness compared to LiDAR and cameras. It can penetrate through adverse weather conditions and provide reliable object detection and tracking. Radar typically provides lower resolution compared to LiDAR and cameras. It can struggle to accurately distinguish between closely spaced objects or provide fine-grained details about their shape and texture. Radar is primarily focused on detecting and tracking objects based on their motion and radar signatures. It can struggle with classifying objects into specific categories, such as distinguishing between pedestrians and cyclists. It can be affected by environmental clutter, such as buildings, trees, and other reflective surfaces. This can lead to false positives or difficulties in accurately detecting and tracking objects in complex urban environments. Due to limited angular resolution, it may not accurately determine the precise location or orientation of objects, particularly in scenarios where objects are close together.

Ultrasonic sensors: An ultrasonic sensor is a device that uses sound waves with frequencies above the range of human hearing (typically above 20 kHz) to detect and measure distance, presence or other attributes of objects in its vicinity. It operates on the principle of sending out ultrasonic waves and measuring the time it takes for the waves to bounce back. They are commonly used in distance measurement, object detection, level sensing, Parking assistance and robotic navigation and obstacle avoidance. By combining the information from cameras, LiDAR, and radar, autonomous vehicles can create a robust perception system that improves reliability and safety. In the test system we are using one camera, one LiDAR and one Radar. The data from these sensors is processed by sophisticated algorithms, allowing the vehicle to understand its environment, detect obstacles, plan trajectories, and make real-time decisions. To overcome these limitations, we use sensor fusion techniques are used to combine the strengths of multiple sensors. By leveraging the complementary nature of radar, LiDAR, and cameras, autonomous vehicles can mitigate individual limitations and create a more robust perception system.

Experimental Setup

Vehicle Setup: The vehicle is a low-speed electric vehicle, which can go up to a speed of 20kmph but capped to 5kmph. It is integrated with drive-by-wire capability. The drive-by-wire refers to an electronic system that either augments or replaces traditional mechanical controls with electronic systems consisting of actuators and sensors to drive the vehicle allowing for seamless electronic control of vehicle's brake, throttle, steering and shifting to enable autonomous vehicle capability. The Compute platform provides the car with on-board computing capability to meet the high computing demands of an autonomous vehicle. All the on-board sensors, the CAN buses, the router, the monitor and IO devices are connected to the Compute platform. The framework used in the vehicle is ROS or Robot Operating System. ROS is an open-source framework that simplifies the development of robotic systems. It follows a modular and reusable approach, enabling developers to create and manage complex robot applications efficiently. ROS utilizes a publish-subscribe messaging system for communication between software components called nodes. This decoupled communication mechanism enhances flexibility and scalability in designing robotic systems. ROS packages, which organize software components, facilitate easy distribution, installation, and versioning. One of the key features of ROS is sensor and actuator abstraction, which simplifies the integration of various hardware components by providing standardized interfaces. Visualization and debugging tools like RViz aid in the development and analysis of robot behavior, while logging and introspection capabilities support debugging and performance analysis.

About Sensors: The vehicle is installed with the following sensors:

Velodyne VLP 16 Hi-Res LiDAR: It's a 16 channel laser based lidar, with a wide Horizontal FOV, but not that high vertical FOV. For further information visit, https://velodynelidar.com/products/puck-hi-res/

GMSL Cameras: The 4 GMSL cameras are responsible for taking images of the 4 sides of the vehicle. GMSL refers to Gigabit Multimedia Serial Link technology used in these cameras. These cameras can transmit large amounts of high-resolution digital video data, bidirectional control data and power over a single coaxial cable. Both these sensors are connected to the Compute platform

Radar: SmartMicro offers a family of automotive radar sensors called UMRR or Universal Medium Range Radar. In this case the project uses UMRR-11 type-132 4D/UHD radar for both long-range and medium range modes with narrow beam and wide bean respectively. For further information on the UMRR sensor specs visit, https://www.smartmicro.com/automotive-radar

Approach: **Sensor Data-Pipeline Architecture**

The */points_raw*, */radar_data* and */camera_data* are the ROS topics which sends in data. Data is passed to a preprocessing pipeline where radar data is filtered up to 5 meters. The Object detection and Decision-Making module takes these sensor data and waypoints as inputs. First the Lidar detection is performed, since it covers the largest area. If the lidar point cloud is found to be on the waypoint marker and the distance between the obstacle and the car is less than 15 meters and within the detection range of 1.5 meters, the car stops. If no object is detected by Lidar, the radar data is checked in an analogous manner, only the detection distance is 5 meters instead of 15. If the data is not detected by radar too, the camera detection is performed, which is performed by DriveNet, neural network able to classify among people, cycle, traffic light and car. The object must be inside the Region of Interest (ROI) for the car to stop. If none of the checks, the car continues.

Object Detection: This segment explains the working of individual object detection methods of the sensors.

Lidar based detection: Lidar Euclidean Cluster is a lidar based clustering technique which groups data points which are close to each other. This involves a preprocessing stage and a PointCloud clustering stage. The preprocessing stage involves:

Points closer than a certain distance is removed, and points above a certain height threshold are clipped. RANSAC algorithm is used to determine ground planes and remove any points that belong to the ground. Points are removed using "difference-of-normal" to remove any points that belong to smooth surface. Since the stack has already preprocessed the PointCloud, this package does not have to redo the filtering stage and can directly move on to the clustering stage. The preprocessed PointCloud is then clustered using Euclidean Cluster Extraction, the cluster tolerance is defined by 'clustering_distance' parameter. Resulting clusters are checked against other neighboring clusters, to determine if the clusters belong to the same group or different.

The intention of doing lidar clustering is to determine the places and sizes of objects and not to identify the objects. In a scenario where a person is standing beside a car, the PointCloud of the car and the person may be perceived as a single object, which exactly fulfils the motive of the algorithm.

Camera based detection: The camera data needs minimal preprocessing, where the size of the camera data is tuned according to the object detection model to be used. The project is able to use any type of object detection or recognition model like darknet, Yolo, VGG-16, inception-net, etc. from a plethora of various neural network architecture. These models work on 2D image data, with RGB information. The model that this project is using the DriveNet architecture and is able to classify only a handful of objects like pedestrian, cycle, car and traffic lights. Depth information cannot be estimated from these models. Given the lidar Euclidean cluster and image detection model, the bounding boxes of the results from these algorithms can be compared. If the 2 sensors are calibrated properly, the distance of the estimated class of object can be approximated.

Radar based detection: It uses a less complex detection algorithm since the radar does not give high resolution data as compared to Lidar and Camera. Though radars are able to estimate the velocity of an object and can be used for tracking, this project does not use that feature. The tracking feature is more robust in camera and lidar algorithms. This sensor is only used to detect objects that are out of Lidar FOV and go undetected by camera in certain environmental conditions.

Advantages

- All 3 detections are performed sequentially rather than in parallel. The sequential processes follow the order of Lidar, radar and camera detection. Since the detection is sequential, during the presence of an object the lidar detects the objects first, if not it is followed by radar detection and if not, then the camera detects the object. The reason for keeping this order is that lidar detects almost everything unless it is too close in which case the radar can detect and the camera might not get good enough detection during the dark. Also, lidar and radar will be able to detect objects from far, which is why the detection range is kept large.
- The sensors complement each other by sharing the entire environment among themselves. Lidar has a blind spot around the car with a radius of 1.8 meters and since the lidar is at a height of 1.7 meters, any object within that range is invisible to the lidar. The camera and radar fill in to cover that gap.
- Since the sensor data is fed into detection and recognition modules in their actual form without changing their datatypes, a plethora of such modules can independently process the data. The outputs of the modules can a combined into a decision-making system that can generate the best possible waypoints for the vehicle.
- This does not introduce extra cost of calibration complexity. The sensors can be changed and mounted differently whenever needed and can be added as needed.

Disadvantages

- The fault tolerant capability is no longer present in this scenario. If the lidar stops working, the vehicle might be able to stop only for objects that are detected and identified by the camera and the radar. Although without the lidar as the primary sensor for localization and mapping and object avoidance, the vehicle won't be able to determine its present location and won't move at all.

Observations: [Fig. 7] The above figure describes the lanes with blue lines and the dotted line in the middle represents the waypoints the car is supposed to travel. The LiDAR FOV is 360 degrees w.r.t the car, but the car only will take action if any object is on the waypoint. Therefore, the detection area for LiDAR is shown as a blue rectangle on the waypoint. As the direction of the waypoint changes, so will the detection region for LiDAR as in Fig 9. The triangle purple area and the dashed area represents the radar and camera detection area respectively. Though the radar has a larger FOV, due to preprocessing the region is viewed small. Multi-sensor data synchronization and failure detection [5] of the sensor impacts the ability of the vehicle to navigate through the environment. A camera usually produces 20-30 frames per second, LiDAR produces data at 20 Hz and Radar at almost 20 Hz. For applications like 3D detection, which requires camera and lidar data at same time, the synchronization is to be done beforehand. This issue becomes more challenging, considering the timestamp's accuracy of different sensors falls into different granularities. Though the RTC PPS module connected to CAN provides proper timestamps at a hardware level. Processing time of the individual sensors will slow down the frequency of detection which further delays the vehicle navigation modules.

In terms of failure detection, there is no universal standard or definition of sensor failure detection. There are no reliable and comprehensive studies on sensor failure detection, which is dangerous since most of the applications in self-driving rely on these sensors. Even when the sensors are working fine, the generated data may not reflect the actual scenario.

Our paper defines an approach, where

- The algorithm takes a pipeline-based approach to make decisions on the sensor data. This way, we do not have to solely rely on the timestamps of the data for proper synchronization.
- The vehicle keeps enough distance from the object and higher deceleration value such that the vehicle can come to a halt immediately after detection.
- If there is a delay on incoming sensor data, the process would continue to run given that the other sensors are working. When the sensor begins to function again the process takes the recent timestamps data from the sensor buffer.

CONCLUSION

Sensor data pipeline techniques have a profound effect in this sensor based real time projects. There is a plethora of algorithms that work in this sector, and there is not a one-works-for-all kind of algorithm. From the analysis, and their pros and cons there are few conclusions that can be made:

- Data pipelining doesn't require retraining any new image recognition model from scratch, rather we can use novel image and point cloud data-based recognition algorithm and pretrained models.
- Data pipelining reduces the cost of running multiple recognition processes running validation on data from multiple sensors to one process running validation on data from multiple sensors.
- Also having a multi-sensor mount architecture, the system would be fault tolerant and robust. Since, most of the blind spots will be covered, and will have a large portion of intersecting FOVs, even if one sensor is at fault, the other sensors object tracking, and detection will help the vehicle navigate safely.
- Data-pipeline allows all these sensors and their fused data to flow independently, as-well-as use the output of these detection tasks to on a decision-making algorithm to get the final action of the vehicle.

REFERENCES

Wang, Y. M. Y. Li, and J. B. Zheng, "A Camera Calibration Technique Based on OpenCV," The 3rd International Conference on Information Sciences and Interaction Sciences. doi:10.1109/icicis.2010.5534797.

https://github.com/autowarefoundation/autoware, Copyright 2021 The Autoware Foundation, The Autoware Foundation (https://www.autoware.org/), Copyright 2021 TIER IV, Inc.

Jelena Kocić, Nenad Jovičić, and VujoDrndarević "Sensors and Sensor Fusion in Autonomous Vehicles" 26th Telecommunications forum TELFOR 2018, doi: 10.1109/TELFOR.2018.8612054

Taraba, M., Adamec, J., Danko, M., &Drgona, P. (2018). "Utilization of modern sensors in autonomous vehicles." 2018 ELEKTRO. doi:10.1109/elektro.2018.8398279

Liangkai Liu, Sidi Lu, Ren Zhong, Baofu Wu, Yongtao Yao, Qingyang Zhang, Weisong Shi, "Computing Systems for Autonomous Driving: State-of-the-Art and Challenges", December 2020, IEEE Internet of Things Journal PP(99):1-1,DOI:10.1109/JIOT.2020.3043716