Parking Area Data Collection and Scenario Extraction for the Purpose of Automatic Parking ADAS Function

Lei Wang¹, Bo Yu¹ and Chao Chen¹

¹ Automotive Data of China (Tianjin) Co., Ltd., Tianjin, 300393, China

Abstract. Automatic Parking is the most significant ADAS function that could achieving mass production in the field of unmanned driving. For the purpose of automatic parking function, we're fundamentally facing the corresponding data collection and data analysis. In view of three classical parking scenarios, this paper is concentrating on three contents including the sensor hardware construction, the perception algorithm development, and scenario data analysis. For the hardware construction, we focus on the sensor installation, time calibration, and space unification. Based on the front camera and RADAR data fusion, we put great emphasis on the multi-target matching and fusion algorithm. And for the purpose of the scenario extraction and big data analysis work, we are developing automatic big data processing Python code, realizing TB level original data labeled, and data completely processed in short time. The whole technology process and tool chain in this program, have extensively high level scientific research and commercial value.

1. Introduction

Automatic parking suitable for massive production is an utmost significant commercial project in the industry of driverless automobile today. The accumulation of tests and scenarios based on data collection in parking lots is one of the important tasks for achieving the mass production of automatic parking. Autonomous driving is relatively demanding in terms of sensors, controllers and actuators from both functional and safety perspectives. Skipping the stages of data collection and the accumulation of scenarios (especially corner scenarios and dangerous scenarios) will cause unpredictable serious consequences.

As for automatic parking in the parking lot, the data we collect mainly includes three types of typical parking scenes, including the scenarios of underground parking lots in office buildings, the scenarios of ground parking lots in industrial parks, and the scenarios of parking lots in public places. The original data is taken from both the simple scenarios (no passenger, low-speed, and light traffic) and the complex scenarios (many passengers, high-speed, and heavy traffic). The classification and definition of the scenarios can be found in Section 4.3.

Figure 1. Ground parking lot in public places and underground parking lot in office buildings.

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
This paper illustrates in detail the processes of the construction of the hardware platform system, the compilation of algorithm for data processing, target matching, and fusion, data acquisition, and data analysis. The entire technical tool chain covering the hardware platform construction, software algorithm development, big data processing and analysis for any types of data to be collected (automatic driving data, advanced assisted driving data), any conditions and scenarios (underground parking lots, highways), and any automatic driving functions (automatic parking, intelligent logistics) is of great scientific and commercial values.

2. Construction of acquisition hardware platform

The acquisition scheme regarding the sensor hardware platform adopted in this paper is mainly based on the Camera camera sensor system and RADAR (RADAR) detection system. The advantage of this combination platform system is that it is not only capable of recognizing surrounding obstacles and detecting pedestrians and lane lines by using the camera system but also capable of detecting distance, azimuth, and speed of the obstacles at a distance with high-precision by using a RADAR:

| Output Information          | Camera | RADAR  | Combination |
|-----------------------------|--------|--------|-------------|
| Distance                    | Average| Excellent| Excellent   |
| Azimuth                     | Average| Excellent| Excellent   |
| Speed \((vertical)\)        | Poor   | Excellent| Excellent   |
| Lane line recognition       | Excellent| Poor   | Excellent   |
| Vehicle detection           | Average| Excellent| Excellent   |
| Pedestrian detection        | Excellent| Poor   | Excellent   |
| Obstacle recognition        | Excellent| Poor   | Excellent   |
| Anti-weather interference   | Poor   | Excellent| Excellent   |
| Anti-light interference     | Poor   | Excellent| Excellent   |

Simultaneously, the automatic parking data acquisition platform combing the camera and RADAR has a relatively lower cost, making it a overall hardware solution targeting the level of massive production. At present, the LiDAR sensors necessary for autonomous driving platforms often cost hundreds of thousands of dollars; therefore they are not suitable to be on the hardware configuration list at the data collection stage.

| Mainstream sensors for data acquisition | Price range (unit: 10,000 RMB) |
|----------------------------------------|---------------------------------|
| Forward RADAR                          | 3                               |
| Short Range RADAR(SRR);                | 3                               |
| Ethernet-Interface camera              | 1                               |
| High-Channel LiDAR                     | 20~100                          |
| Low-Channel LiDAR                      | 40~60                           |
| Mobileye 630                           | 5                               |
| MEMS INS                               | 5                               |
| Binocular Camera                       | 5~8                             |
| Fiber Level INS                        | 20~40                           |

As shown in Figure 2, the main structure of the sensor platform includes the following four subsystems.
Table 3. Functions list of the four subsystems

| Subsystem                        | Function                                                                 |
|----------------------------------|--------------------------------------------------------------------------|
| Forward RADAR system             | For detecting the distance, speed, and azimuth of the targets in front of ego vehicle |
| Forward camera recognition system| For detecting and recognizing targets in the front, and for conducting sensor fusion processing with RADAR target results |
| Blind zone monitoring camera system | For detecting and recognizing the target at side and conducting vehicle forward and backward blind spot monitoring |
| Surround view camera system       | For close range detection of the vehicle using surround view camera system |

The layout of the four subsystems is shown in Figure 2.

Figure 2. Installation of sensors of data acquisition platform.

2.1. The fusion system of forward camera and RADAR

2.1.1. Number and type of sensors: one ARS408 RADAR; one ethernet-interface camera

2.1.2. Overall detection layout of the forward fusion system:

Figure 3. Detection range of the system combining forward Camera and RADAR.

2.1.3. Installation position of the forward fusion system:

Table 4. Positions of forward fusion system installed in vehicles

| Forward fusion system         | Installation position                   |
|-------------------------------|----------------------------------------|
| Forward 77GHz millimeter wave | Center of front grille                  |
| Forward recognition camera    | Inside of front windshield             |

Figure 4. Installation position of ARS408 RADAR.
2.2. Surround view camera monitoring system

2.2.1. Number and type of sensors. 4 fisheye cameras with a horizontal view angle range close to 180°

2.2.2. Overall layout of the surround view camera:

2.2.3. Detection range of the surround view camera:

2.2.4. Installation position of the surround view camera system:

| Sensor name            | Installation position                                |
|------------------------|------------------------------------------------------|
| Forward fisheye camera | Center of front grille                               |
| Backward fisheye camera| Right above the license plate                        |
| Side fisheye camera    | Embedded under the left and right rearview mirrors   |

2.3. Side blind spot monitoring camera system

2.3.1. Number and type of sensors: 5 ethernet-interface cameras

2.3.2. Overall layout of blind spot monitoring cameras

2.3.3. Detection range of blind spot monitoring camera
2.3.4. Installation position of blind spot monitoring camera system

Table 6. Installation positions of blind spot monitoring system

| Sensor name                   | Installation position        |
|-------------------------------|-----------------------------|
| Front view BSD camera         | Left and right rearview mirror |
| Rear view BSD camera          | Left and right rearview mirror |
| Backward surveillance camera  | Right above the license plate |

Figure 10. Installation position of the camera at the left side of the vehicle.

3. Space and time calibration of hardware sensor platform

After setting the position of the sensors, it is necessary to calibrate the hardware system before data acquisition. This task mainly includes space synchronization calibration and time synchronization calibration.

3.1. Space calibration of acquisition platform sensors

Space synchronization refers to putting the data measured by all sensors into the same coordinate of the vehicle (the ground projection of the center point of the rear axis is usually selected as the reference coordinate). After converting the independent results obtained by different sensors targeting the same object, we can obtain the coordinated results. This process involves the translation and rotation transformation between various coordinate systems. Sensor coordinates are generally divided into two categories: 1) local coordinate system, including image coordinate system, radar coordinate system, odometer coordinate system, vehicle coordinate system, etc.; 2) global coordinate system, including world coordinate system, WGS-84 latitude and longitude coordinate system, UTM coordinate system, which are used to describe the absolute position of the vehicle in world coordinates (GPS) or to characterize the position and status of the vehicle on a HD map.

The space coordinate systems involved in the acquisition platform cover the six categories, namely, pixel coordinate system, image plane coordinate system,; camera coordinate system, vehicle coordinate system, RADAR coordinate system, and global coordinate system.

Take the space calibration of the raw data acquired by the camera sensor in the vehicle coordinate system as an example, \((u, v)\) is the pixel coordinate, \(Zc\) is the depth of field, and \((X, Y, Z)\) is the vehicle coordinate. Their conversion relationship is shown the following formula:

\[
Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1/\frac{dx}{dv} & 0 & u_0 \\ 0 & 1/\frac{dy}{dv} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \]

The translation and rotation of the coordinates from the non-camera sensor coordinate system to the vehicle coordinate system follow the formula:
\[
\begin{bmatrix}
X_c \\
Y_c \\
Z_c
\end{bmatrix} =
\begin{bmatrix}
R & t \\
0^T & 1
\end{bmatrix}
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

\( \mathbf{R} \) is a 3*3 space rotation matrix; \( \mathbf{t} \) is a 1*3 Space translation matrix.

For the image sensor, the calibration plate is adopted to obtain internal parameters such as focus and focal length, which are used for radial tangential distortion correction. In the 3*3 calibration matrix, there are 4 internal parameters and 5 distortion parameters.

Figure 11. Calibration method using camera calibration board.

The information calibrated by other sensors are as follows:

| Type of calibration sensor | Parameters of calibration sensor |
|-----------------------------|----------------------------------|
| RADAR system | Calibration of course angle in three directions (forward and backward radar) |
| Forward recognition and blind spot monitoring camera system | Calibration of pitch angle and roll angle (side short range radar) |
| Four calibration boards are used to calibrate the camera system | The calibration board calibrates the camera parameters to obtain internal parameters such as focus and focal length, which are used to correct radial tangential distortion. |
| Surround view camera system | Calibration board calibrates the camera system to obtain internal parameters such as focus and focal length, which are used to correct radial tangential distortion. |

Correct space calibration between sensors and ego vehicle, with position measurement between sensors and the target, can help us to obtain the positional relationship of the vehicle relative to the current target. Figure 12 shows a calculation demonstration: the position of the target to the RADAR is (1.0, 2.0, 0.0); after translation and rotation transformation, the position of the target to the vehicle is (1.1, 2.0, 0.2).

Figure 12. Space calibration of RADAR.

### 3.2. Time calibration of the acquisition platform sensor

The unification of time stamps is also an important technical point of the multi-sensor data acquisition system. Different types of sensor data acquisition systems perceive the environment independently. Generally speaking, the refresh rates among sensors are different, and the collected sensor data is not at the same time stamp. Therefore, it is necessary to accurately calibrate the space coordinate system and synchronize the time of the various sensors.
Table 8. Refresh rate of mainstream perception sensors.

| Sensor type       | Refresh rate |
|-------------------|--------------|
| Camera            | 30Hz         |
| RADAR             | 20Hz         |
| Mobileye          | 60~70ms      |
| High-Channel LiDAR| 10Hz         |
| Low-Channel LiDAR | 25Hz         |

There are basically three ways for timing synchronization:

- Different sensors have their own independent clocks. In the process of data fusion, the low-frequency sensor is used as the reference to match the data frame of the high-frequency sensor for time fusion;
- The time stamp of GPS is used as the standard of timestamp for other sensors. For the timing of other low frequency sensors, if the timed sensor and the GPS sampling point are not at the same time point, then it is necessary to perform interpolation and extrapolation on the non-synchronized time points;
- For multiple-camera sensor systems, the same switch can be used to synchronize the time points of cameras’ exposure and sampling.

4. Matching and fusion of forward RADAR and camera

The algorithms used for multi-target dynamic matching mainly include bidding algorithm, Hungarian algorithm, GNN algorithm and JPDA algorithm. This paper selects GNN (Global Nearest Neighbor Algorithm) for target matching. Consider target matching from both simple and complex scenarios.

4.1. For multi-target objects with a long relative distance, if there is no overlapping area in the pixel coordinate system (as shown in Figure 13), then we can track each bounding box. When a RADAR projection value falls into a rectangular box, we can match this projection value with this bounding box.

4.2. For obstacle-dense areas, there will be some overlap between different bounding boxes. Multiple projection points of RADAR will fall into the same bounding box. We use GNN algorithm to calculate the allocation probability of RADAR projection points that fall into the same rectangular frame; then we take the maximum allocation probability (or the minimum value of the Mahalanobis distance) as the matching result. After that, those unmatched RADAR pixel values are deleted.
After matching, we perform fusion output on the distance and speed of the RADAR and the camera using a non-linear Kalman filter.

The problems to be solved during multi-sensor matching and fusion are summarized as follows:

Table 9. Problems to be solved using the matching and fusion algorithm.

| Problem                                                                 |
|-------------------------------------------------------------------------|
| For the same obstacle, if RADAR outputs two tracking trajectories,       |
| then there will be two fusion results.                                  |
| Diffraction effect of RADAR will detect the separated obstacles         |
| Too bright or too dark illumination will affect the obstacle detection   |
| efficiency of the camera                                                |
| Low pedestrian reflectivity will affect the RADAR detection efficiency   |
| If the obstacle is beyond the detection range of the camera, then fusion |
| shall take the RADAR results                                            |

5. Collection and output of raw data and scenario annotation

Based on the camera and RADAR fusion platform, we divide the collected data into two categories:

5.1. The original data of this vehicle

The original data of ego car contains a total of 38 fields. The main ones are summarized in the following table:

Table 10. CAN data of ego vehicle.

| Header field | Indication                      |
|--------------|---------------------------------|
| Frame ID     | Frame number                    |
| Vehicle Speed| Speed of this vehicle           |
| Steering Wheel| Rotation angle of the steering wheel |
| Brake        | Brake force                     |
| Engine Speed | Rotate speed of engine          |
| Gear         | Gear                             |
| Yaw Rate     | Yaw rate                        |
| CIPV         | ID of radial nearest target     |
| Longitude ACC| Longitudinal acceleration       |
|              |                                 |

5.2. Raw data of the target obstacle

The original data of the target contains 16 fields. The main ones are summarized in the following table:

Table 11. Raw data of the target car.

| Header field | Indication                      |
|--------------|---------------------------------|
| Frame ID     | Frame number                    |
| Object ID    | Target ID                       |
| Class        | Obstacle classification         |
| Width, Height| Obstacle height and width       |
5.3. Scene annotation
Based on the automatic parking scene, we categorize the parking areas into on-site area and off-site area, covering 8 phases such as driving into the parking lot, parking the car, and driving out of the parking lot. Each phase includes 18 major scenarios such as turning left and going straight. Each major scenario can be divided into 31 sub-scenarios. According to the original data of the video and that of the target vehicle described in Sections 4.1 and 4.2, we annotate the original data for various scenarios. The logical structure of scenario division is shown in Table 12:

| Region Division (2 categories) | Phase division (8 categories) | Scene division (18 categories) | Scene subdivision (31 categories) |
|-------------------------------|-------------------------------|--------------------------------|----------------------------------|

6. Big data analysis and extraction based on original and annotated data:
We need to extract and calculate 29 fields including ID information of important obstacles, TTC and THW based on the original and annotation data. However, manual extraction is inefficient in this respect, so we use python programming to process the original and annotation data, and to automatically process tens of thousands of original data tables in batches. We also perform automatic generation and parallel submission operations, so as to programmatically process the TB-level original data tables of three types of typical parking areas (underground parking lots in offices, ground parking lots in industrial parks, and ground parking lots in public places), with the time totaling 100 days. All data can be processed in a few hours. Python can make big data processing more efficient. The original data table is shown in Figure 15.

Python is more advantageous and efficient in big data processing. The code is shown in Figure 16:

7. Conclusion
This paper, focusing on automatic parking ADAS function, collects data from three types of typical parking areas, namely, underground parking lots in office buildings, ground parking lots in industrial parks, and ground parking lots in public places. Based on the collected raw data, this paper generates corresponding scenarios for data annotation, which is realized through big data processing and automatic batch generation of target field tables using Python. The whole process tool chain including the hardware installation and calibration of sensors (first step), algorithm development of the camera RADAR software (second step), coding of big data batch processing (third step), and generation of the form of required fields is of great scientific research and commercial value(last step). In addition, this paper, oriented at Shanghai Automotive Industry Corporation, can also provide direct reference for other data acquisition projects.
References

[1] telee. (2014) Golder's Data Acquisition Experience http://geekcar.com/archives/10033
[2] AI vanguard. (2017) Introduction to Hardware and Software Technology of Autopilot Vehicle. https://blog.csdn.net/jtnbcoc8n2i9/article/details/78898507
[3] Jin Li S, Cheng L (2016) Night Front Vehicle Detection Based on Millimeter Wave Radar and Machine Vision. Journal of Automotive Safety and Energy 7
[4] Ren Yan H. (2011) Research on Obstacle Information Recognition and Processing System for Unmanned Vehicles. Lanzhou University of Technology Lanzhou
[5] Chen An. (2017) Research on Automatic Driving System of Internet-connected Vehicles and Its Application in Parking Lot Scenes. Zhejiang University Hangzhou
[6] Zhengping Ji Danil Prokhorov. (2008) Radar-Vision Fusion for Object Classification.
[7] Luca Bombini, Pietro Cerri, Paolo Medici. (2006) Radar-vision fusion for vehicle detection
[8] Walther Wachenfeld Philipp Junietz, (2017) Safety Assurance for Highly Automated Driving – The PEGASUS Approach.
[9] Tom Voege, Bogdan Godziejewski, (2017) Connected and Automated Transport Studies and reports.