LiDAR-Camera Fusion in Self-Driving Cars

Chinmay Khamesera
*c*khamesra@wpi.edu

Anujay Sharma
*a*sharma@wpi.edu

Rutwik Bonde
*r*rbonde@wpi.edu

Atharva Mahindrakar
*a*amahindrakar@wpi.edu

***Abstract*—In this paper, we focus on Low level fusion and Mid level fusion of camera and LiDAR data. In the low level fusion we use the YOLO Detection for object detection and visualize 3D LiDAR points in Open3D for depth estimation. After performing these two tasks we then project the 3D LiDAR points on the image plane. Further, we try to fuse the 2D bounding box obtained from LiDAR points with YOLO Detections. Whereas in the Mid Level fusion, we fuse the Yolo detection with a 3D bounding box obtained from the LiDAR points data on an image, we get the 3D Bounding Boxes from LiDAR. These 3D Bounding Boxes are converted to 2D Bounding Boxes. Finally, the LiDAR 2D Bounding Boxes are fused with YOLO 2D Bounding Boxes using intersection over Union.**

***Index Terms*—Sensor Fusion, 3D LiDAR points, Calibration, YOLO Detections, 2D/3D Bounding Box, Hungarian Algorithm, KITTI Dataset**

I. Introduction

The method of combining data from RADAR, LiDAR, camera, and ultrasonic sensors to evaluate ambient conditions for detection confidence is known as sensor fusion. Each sensor cannot function independently and provide all the data required for an autonomous car to operate in the safest possible manner. Autonomous driving systems can profit from the strengths of each individual sensor while balancing out the aggregate advantages of using a variety of them. Sensor fusion data is processed by autonomous cars using preprogrammed algorithms. This enables autonomous cars to make decisions and choose the appropriate course of action. Machine learning techniques are used by autonomous driving systems to continuously comprehend their environment, make wise choices, and anticipate potential changes that can affect a path.

In this study, we execute tasks to understand the environment. YOLO detection and 3D LiDAR points are some of these jobs. YOLO stands for the phrase "You Only Look Once." This algorithm instantly recognizes and locates different things in images. Convolutional neural networks are used by the YOLO method to recognize items instantly. The approach just needs one forward propagation through a neural network to detect objects, as the name would imply. This indicates that a single algorithm run is used to perform prediction throughout the full image. Multiple class probabilities and bounding boxes are simultaneously predicted using CNN.

A collection of multiple dots dispersed across a 3D space called a 3D LiDAR points cloud is created when data points are gathered using sensors like lidar. Each dot is produced by sensors that emit light and time how long it takes for the light to be reflected back into the sensor. To create a complete image, the gathered dots are compounded. The fundamental training data for driverless technology is 3D point cloud picture labeled data. Annotating a 3D point cloud is thought to be the most effective method for precise lidar sensor recognition. Using 3D boxing, 3D point cloud image annotation identifies the target object in a 3D image acquired by lidar sensors. Vehicles, people, trees, traffic signs, and other objects are among the targets.

After YOLO detection and 3D LiDAR points, we aim to fuse the LiDAR points with YOLO detection by projecting the 3D LiDAR points onto the image plane. Additionally, LiDAR provides 3D Bounding Boxes. They are transformed into 2D bounding boxes from their original 3D form. Finally, intersection over union is utilized to combine the YOLO 2D bounding boxes with the LiDAR 2D bounding boxes.

II. Literature Review

Autonomous vehicles map and detect the environment around them using a variety of sensors, including LiDAR, radar, cameras, and ultrasonic sensors[1]. The authors of the article [2] presented RoadPlotDATMO, which uses numerous LiDAR sensors to detect moving objects. However, the capacity to classify objects placed limitations on numerous LiDAR sensor-based object recognition and tracking. Limitations on the detection and tracking of numerous things using LiDAR sensors are imposed by the ability to classify items. Because it can depict a scene in 3D, the stereo-vision-based method among more recent approaches appears to be more appropriate for general object detection [3]. However, real-time performance is a crucial concern.The collaborative fusion of the laser scanner and camera technique described in [4] is also restricted to vehicle recognition, making it challenging to identify stationary objects like pedestrians, motorcyclists, and other road users.

LiDAR and camera fusion techniques have been introduced as an alternative, taking real-time performance, object identification, classification capability, and accurate distance calculation into consideration. In order to speed up processing, the fusion technique creates a relationship

between the 3D points from LiDAR and the object identified by a camera[5]. This is achievable because sensor fusion improves robustness and detection accuracy while making up for the shortcomings of the individual sensors.Based on various levels of data fusion, LiDAR and camera fusion algorithms have been introduced for object detection, taking into account real-time performance, object detection, classification capabilities, and accurate distance calculation. The fusion technique connects 3D points from a LiDAR and an object that a camera has detected in order to expedite processing.

To integrate camera and LiDAR data at the low level, intrinsic calibration between sensors is a vital first step in the fusion process. Therefore, when calculating geometrical properties such as the position and orientation of each sensor, it is vital to take into account the relative positions and orientations of the other sensors. Therefore, the calibration for fusing sensor data is done by finding the correlation between 3D points and 2D image pixels. For the integration of heterogeneous sensor data, it is essential to identify the features from each sensor and the geometric relationship between them [6].

The exact distance estimation and object identification in an autonomous vehicle's path are the main topics of this research. The accuracy of sensor calibration is crucial, especially when sensor data fusion is used to determine the depth of moving objects. The majority of calibration techniques use external objects such a trihedral rig [7–10], circles, board patterns [16–17], checkerboards, and others [11–14] to verify the two sensors' correlation. Numerous scholars have recently suggested automatic calibrating techniques [15]. These techniques locate the intended object since each sensor detects the object's center automatically and circles it on a plane. The plane is extracted using the random sample consensus (RANSAC) approach, and the external parameters for data fusion are precisely refined using the iterative closest point (ICP) algorithm based on nonlinear optimization [12,15].

Using LiDAR and video sensors, the approach records the numerous physical characteristics of the environment in order to understand its surroundings. In order to detect the sensor displacement using the 3D marker, the LiDAR and camera sensors must first be calibrated, as seen in Figure 1. The LiDAR point cloud is then mapped onto the camera image using the calibration parameter and the intrinsic parameter of the camera. Using pixel-to-point matching of the sensor data and an assessment of the alignment score of the matched data, the accuracy and utility of the suggested method are assessed.

A. The Calibration Method of the Camera and LiDAR

The calibration of LiDAR and the camera is done in two steps back-to-back in order to find the intrinsic and extrinsic calibration parameters. The camera intrinsic parameters are initially determined using the well-known checkerboard approach for camera calibration before acquiring the LiDAR

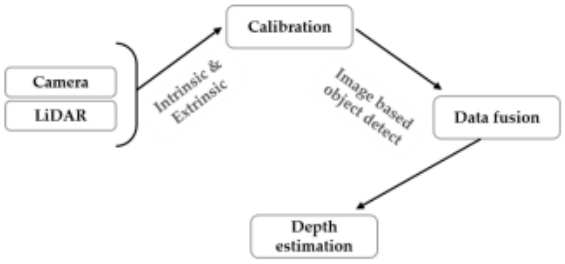

Fig. 1. Process flow of the Algorithm

and camera extrinsic parameters. The camera's image data is represented using a 2D coordinate system (U, V), and the 3D point cloud produced using the LiDAR sensors' raw measurements is displayed using a 3D coordinate system (X, Y, Z). The main objective of the camera and LiDAR calibration is to calculate the projective transformation matrix, which will project the 3D LiDAR points (X, Y, and Z) onto the 2D image (U, V). Equation (1) is the formula for the projective transformation matrix.

$$M_{projective} = C_{intrinsic} * M_{extrinsic} \tag{1}$$

$$c_i = \begin{pmatrix} u \\ v \\ i \end{pmatrix} = M_{projective} * P_i \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2}$$

An initial raw measurement from the sensor is in the form of parameters in a standard polar coordinate format, where geometric distance and horizontal and vertical angle are represented as polar coordinates with respect to the sensor coordinate system, and points are converted from the polar coordinate to the Cartesian coordinate. The calibration parameters are used to put 3D points onto the image once the LiDAR data has been transformed into Cartesian coordinates. where p2D carries the values for the camera intensity and the LiDAR point indices, and ui and vi represent the 2D coordinates.

The camera's intrinsic calibration uses the checkerboard marker to estimate the internal parameters of the device, such as the focal length, skew distortion, and picture center. The camera matrix, distortion coefficients, and camera projection matrix are all included in the calibration findings, which are crucial steps in calibrating the LiDAR and camera for data fusion. The extrinsic parameter, or the six degrees of freedom (6DOF) relative transformation between LiDAR and the camera for data fusion, must be determined after the intrinsic camera parameter estimation.

The calibration parameters are estimated using the four circles that were found in the image. Equation (1) can be rewritten as follows, taking into account rotational invariance.

$$M_{projective} = \begin{bmatrix} f & 0 & u_o & 0 \\ 0 & f & v_o & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & t_z \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

These translation vectors are calculated using a distance calculation based on the circle radius that the marker's sensor has detected. To improve the accuracy of the calibration parameters, rotation parameters are estimated after translation parameters. We used the typical least-square best-fitting rigid body transformation to estimate the rotation.

$$p_{2D} * \begin{pmatrix} U \\ V \\ 1 \end{pmatrix} = \left| r_{11} \right| \quad (4)$$

B. Fusion of LiDAR and Camera Data

The LiDAR data points are projected onto the camera image once the intrinsic and extrinsic parameters have been estimated. The calibration parameters are used to put 3D points onto the image once the LiDAR data has been transformed into Cartesian coordinates.

$$c_i * \begin{pmatrix} U \\ V \\ 1 \end{pmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} * p_i * \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (5)$$

III. Problem Description

Although vision-based techniques are more reliable and accurate in detecting objects, they fall short in accurately calculating the object's distance.In contrast, LiDAR-based approaches are extremely reliable and accurate at determining an item's distance, but they are constrained by the capacity to classify the object. There are a few methods based on the stereo vision camera as an alternative because of its capacity to see the surroundings in 3D.

Therefore, the sensor-fusion-based strategy seems to be the best for handling object recognition at a distance from the vehicle's current position while taking into account the capability and limitations of the camera and LiDAR sensor. Additionally, based on the various data levels employed for fusion, sensor fusion systems are widely divided into three primary categories: low-level fusion and mid-level fusion. Low-level fusion uses the physical position of the sensors to fuse the raw measurements from the sensors, while mid-level fusion uses a series of pre-processing steps to extract a specific feature from the raw data. Low-level fusion looks to be more ideal for autonomous vehicles due to its real-time performance and the more exact fusion of data, despite the fact that each fusion approach exhibits well-known advantages and disadvantages when compared to one another. Therefore, we start with low-level fusion and then attempt to implement mid-level fusion and high-level fusion to compare the actual results.

IV. Overview of Sensors

Camera:
Cameras are one of the most widely used technologies for observing the environment. A camera produces crisp images of the surrounding by detecting lights emitted from the surroundings on a photosensitive surface (image plane) using a camera lens (placed in front of the sensor). Cameras are generally affordable, and when used in conjunction with appropriate software, they can identify both moving and stationary impediments within their field of vision, as well as produce high-resolution photographs of the surroundings.

LiDAR:
LiDAR, or light detection and ranging, was first developed in the 1960s and has since been widely employed in the mapping of aeronautical and aerospace terrain. The first commercial LiDAR's with 2000 to 25,000 pulses per second (PPS) for topographic mapping applications were manufactured and deployed in the mid-1990s by laser scanner manufacturers. LiDAR is a distant sensing technique that works on the principle of producing infrared or laser light pulses that reflect off of target objects. The equipment detects these reflections, and the time between emission and reception of the light pulse allows for distance estimate. LiDAR sensors produce data in the form of a series of points, also known as point cloud data (PCD), in 1D, 2D, and 3D areas, as well as object intensity information. The PCD comprises the x, y, and z coordinates as well as the intensity information of the obstacles inside the scene or surrounds for 3D LiDAR sensors.

V. YOLO Detection

A. Introduction

Regions in the image are used by all of the earlier object detection techniques to pinpoint the object's location inside the image. The network does not view the entire picture. Instead, there is a significant likelihood that the object will be present in some areas of the photograph. Techniques for detecting objects form the basis of artificial intelligence.
You Only Look Once (YOLO) is a popular algorithm that has gone viral. YOLO's object detection technique is well known. The compact size and quick calculation speed of the model form the basis of the YOLO target identification technique. The organization of YOLO is simple. Through the neural network, it can output the position and category of the bounding box immediately. Because all YOLO needs to do to obtain the final detection result is upload the image to the network, its speed is quick. 24 convolution layers precede two fully coupled layers in the original YOLO design. Non-maxima suppression is the practice of selecting the bounding boxes with the highest Intersection Over Union (IOU) with the ground truth out of several bounding boxes predicted by YOLO for each grid cell.

B. YOLO Architecture

The head, neck, and backbone are the three main parts that make up the YOLO Model. Convolutional layers make up the backbone of the network, which is used to identify and process an image's main elements. Since detection requires finer features than classification, the backbone is initially trained on a classification dataset, such as ImageNet, and is often trained at a lower resolution than the final detection model. Predictions on probabilities and bounding box coordinates are made by the neck using the features from the convolution layers in the backbone with fully connected layers. The head is the network's final output layer and can be switched out for other layers that have the same input form for transfer learning. The original YOLO research paper's head is tensor of form,

$$S \bigotimes S \bigotimes (C + B * 5) \tag{6}$$

measuring 7 x 7 x 30 with a split size S of 7, 20 classes C, and 2 projected bounding boxes B. Together, these three model components extract important visual properties from the image, classify them, and then bind them.

VI. Low-Level Sensor Fusion

The goal of low-level sensor fusion is to combine the unprocessed data obtained from many sensors. For instance, we combine pixels from cameras and point clouds from LiDARs. Raw sensor data is processed using low level sensor fusion, also known as sensor level abstraction. The data can be aggregated at this level if the same physical attribute is measured by many sensors. Data from sensors that measure several properties is integrated at a higher level.

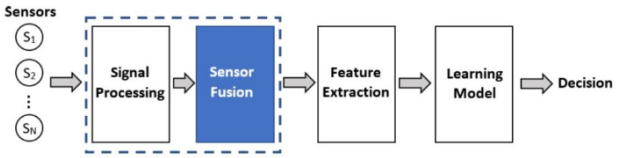

Fig. 2. Low Level Sensor Fusion

In this study, we first construct YOLO detections at the basic level of fusion, then we visualize 3D LiDAR points in Open3D, then we project 3D LiDAR points into the picture plane, and finally we fuse 3D LiDAR points with YOLO detections.

VII. Mid-Level Sensor Fusion

The goal of mid-level sensor fusion is to combine things that were individually recognized using sensor data. Mid-level sensor fusion, sometimes referred to as feature level abstraction, takes the features from several independent sensors and produces distinct feature vector representations.

In this paper, in the mid-level fusion, we implement YOLO detections, followed by the projection of LiDAR points on image, and then generating 3D bounding boxes

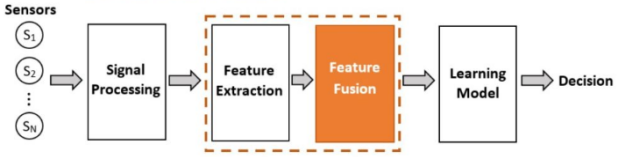

Fig. 3. Mid Level Sensor Fusion

from LiDAR. The 3D bounding box is converted to 2D bounding boxes, and then finally the LiDAR 2D bounding box is fused with YOLO 2D bounding box using intersection over the union.

VIII. Results

The algorithm was successfully able to fuse pointcloud data with image data. The implementation for YOLO object detection as shown in Figure 4 was able to detect objects in real-time and draw bounding boxes around them. These bounding boxes were successfully able to determine the object type in real time. Next, the Lidar data-points were

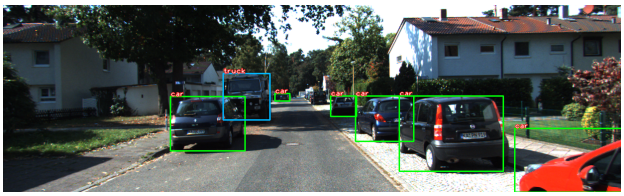

Fig. 4. YOLO

processed to form mapped onto to the image plane can be done using the transformation as given in Eq. 2 can be shown in Figure 5. Once a the 3d points are mapped they were removed from outside the bounding boxes created by YOLO.

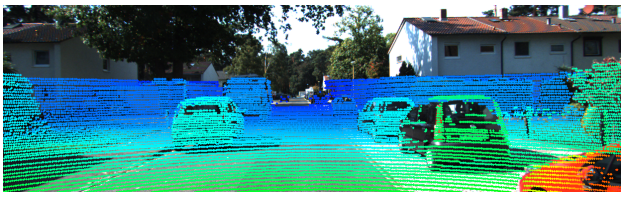

Fig. 5. LIDAR Points

The points inside the bounding boxes can be seen in Figure 6. These points only give the 3D coordinates of points inside the boxes. Some of these points can be removed by applying a set threshold for minimum and maximum distance.

The new 3D metric created in Figure 6 can be used to generate successful 3D bounding boxes for the objects by fusing depth and image data. This has been successfully demostrated in Figure 7.

Once the 3D boxes are successfully generated in the image space, they are used to generate 2D bounding boxes

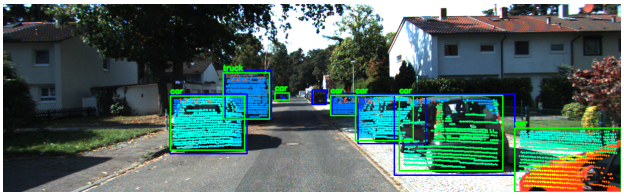

Fig. 6. Lidar Points inside the bounding box

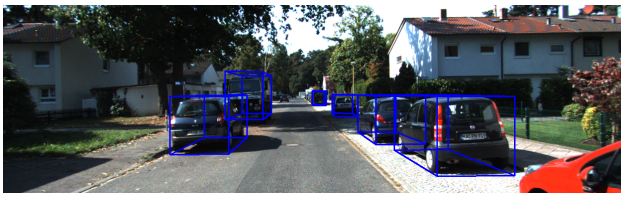

Fig. 7. 3D LIDAR

as shown in Figure 9. This is the final box for obstacle avoidance from the resulting sensor fusion. The algorithm was

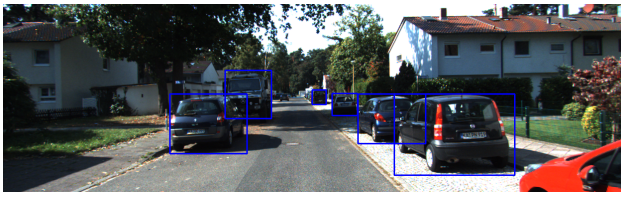

Fig. 8. 2D Bounding Box from 3D box

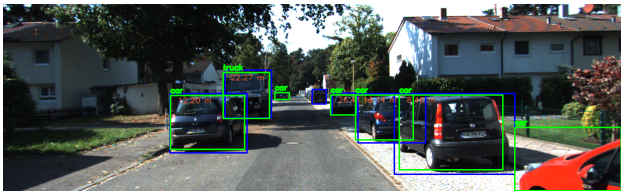

Fig. 9. YOLO detection with LiDAR depth estimation

successfully able to get fusion from the Lidar and Camera sensor, with bounding boxes given from the Intersection over Union.

IX. Discussion

The paper demonstrates that a more reliable and accurate way of detecting objects is possible given the fusion method. This approach takes care of the classification aspect using the cameras and the distance metric with LiDars. This approach is also easy to implement since the calibration takes care for processing the point cloud by reprojection on the camera.

This approach even though simpler, does not process pointclouds. This could be limiting in cases where pointcloud processing might be a requirement for safety critical standards. This method is highly restrained due to 2D image processing for two reasons. The first one being limited by the Field of View per camera lens, which means that the pointcloud is only reprojecting on the camera image and away all the other useful data. A full utilisation would require multiple FOV's from different camera modules. The second being the bounding box generation from trained detection algorithm from camera images is only limited to the data set. It would not be able to fuse distance metrics when the object in the image in unknown.

Some of these limiting challenges can be addressed by also processing the pointcloud for object detection [19]. This can be computationally expensive but would give better results for safer self driving [18].

X. Conclusion and Future Work

In this paper, with the aid of a monocular 2D picture and a LiDAR point cloud data, we introduced the sensor fusion-based 3D object recognition technique in this study. We also showed how the suggested low-level and mid-level sensor fusion networks performed on the gathered dataset. This improves accuracy when used with LIDAR data and the YOLO, a unified object detection model. To decrease the number of detection errors and overcome the drawbacks of individual sensors functioning independently, different calibration metrics and data streams from diverse sensing modalities are used. Additionally, sensor fusion aids in the creation of a consistent model that can properly sense the world under a variety of environmental situations.

For safe and dependable scene perception, further work must be done to enhance item detection performance in all conceivable settings, including harsh weather. It is essential to create trustworthy object identification algorithms that can tell barriers apart from their surroundings.

References

[1] Li, Q.; Chen, L.; Li, M.; Shaw, S.L.; Nüchter, A. A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios. IEEE Trans. Veh. Technol. 2014, 63, 540–555.

[2] Na, K.-I.; Byun, J.; Roh, M.; Seo, B.-S. RoadPlot-DATMO: Moving object tracking and track fusion system using multiple sensors. In Proceedings of the International Conference on Connected Vehicles and Expo (ICCVE), Shenzhen, China, 19–23 October 2015; pp. 142–143.

[3] Mathias, P.; Raphael, L.; Dominique, G.; Alain, L.; Didier, A. Proposition of Generic Validation Criteria Using Stereo-vision for On-Road Obstacle Detection. IJRA Int. J. Robot. Autom. 2014, 29, 65–87, ACTA Press; pp. 1925–7090.

[4] Dominique, G.; Aurelien, C.; Rachid, B. Vehicle Detection and Tracking by Collaborative Fusion between Laser Scanner and Camera. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013.

[5] Zhao, G.; Xiao, X.; Yuan, J.; Ng, G.W. Fusion of 3D-LIDAR and camera data for scene parsing. J. Vis. Commun. Image Represent. 2014, 25, 165–183.

[6] Jessica, V.B.; Marie, O.; Dominique, G.; Homayoun,N. Autonomous Vehicle Perception: The Technology of Today and Tomorrow. Transp. Res. C Emerg. Technol. 2018, 89, 384–406.

[7] Maxime, D.; Aurelien, P.; Martial, S.; Guy Le, B. Moving Object Detection in Real-Time Using Stereo from a Mobile Platform. Unmanned Syst. 2015, 3, 253–266.

[8] Raphael, L.; Cyril, R.; Dominique, G.; Didier, A. Cooperative Fusion for Multi-Obstacles Detection with Use of Stereovision and Laser Scanner. Auton. Robot. 2005, 19, 117–140

[9] Douillard, B.; Fox, D.; Ramos, F.; Durrant-Whyte, H. Classification and semantic mapping of urban environments. Int. J. Robot. Res. 2011, 30, 5–32.
[10] Li, Q.; Chen, L.; Li, M.; Shaw, S.L.; Nüchter, A. A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios. IEEE Trans. Veh. Technol. 2014, 63, 540–555.
[11] Na, K.-I.; Byun, J.; Roh, M.; Seo, B.-S. RoadPlot-DATMO: Moving object tracking and track fusion system using multiple sensors. In Proceedings of the International Conference on Connected Vehicles and Expo (ICCVE), Shenzhen, China, 19–23 October 2015; pp. 142–143
[12] Mathias, P.; Raphael, L.; Dominique, G.; Alain, L.; Didier, A. Proposition of Generic Validation Criteria Using Stereo-vision for On-Road Obstacle Detection. IJRA Int. J. Robot. Autom. 2014, 29, 65–87, doi:10.2316/Journal.206.2014.1.206-3765. ACTA Press; pp. 1925–7090.
[13] Zhao, G.; Xiao, X.; Yuan, J.; Ng, G.W. Fusion of 3D-LIDAR and camera data for scene parsing. J. Vis.Commun. Image Represent. 2014, 25, 165–183.
[14] Li, J. Fusion of Lidar 3D Points Cloud with 2D Digital Camera Image; Oakland University: Rochester, MI, USA, 2015
[15] Li, J.; He, X.; Li, J. 2D LiDAR and camera fusion in 3D modeling of indoor environment. In Proceedings of the 2015 National Aerospace and Electronics Conference (NAECON), Dayton, OH, USA, 15–19 June 2015; pp. 379–383.
[16] Mastin, A.; Kepner, J.; Fisher, J. Automatic registration of LiDAR and optical images of urban scenes. In Proceedings of the Computer Vision and Pattern Recognition, Miami, FL, USA, 22–24 June 2009; pp. 2639– 2646
[17] Park, Y. Calibration between color camera and 3D LiDAR instruments with a polygonal planar board. Sensors 2014, 14, 5333–5353.
[18] Siheng Chen, Baoan Liu, Chen Feng, Carlos Vallespi-Gonzalez, Carl Wellington. 3D Point Cloud Processing and Learning for Autonomous Driving, arXiv, 2020
[19] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu and M. Bennamoun, "Deep Learning for 3D Point Clouds: A Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 12, pp. 4338-4364, 1 Dec. 2021, doi: 10.1109/TPAMI.2020.3005434.