electronics

MDPI

Article

Real-Time 3D Object Detection and Classification in Autonomous Driving Environment Using 3D LiDAR and Camera Sensors

K. S. Arikumar [1], A. Deepak Kumar [2], Thippa Reddy Gadekallu [3,4], Sahaya Beni Prathiba [5,*] and K. Tamilarasi [5]

[1] School of Computer Science and Engineering, VIT-AP University, Vijayawada 522237, India
[2] Department of Computer Science and Engineering, St. Joseph's Institute of Technology, Chennai 600119, India
[3] School of Information Technology and Engineering, Vellore Institute of Technology, Tamil Nadu 632014, India
[4] Department of Electrical and Computer Engineering, Lebanese American University, Byblos P.O. Box 13-5053, Lebanon
[5] School of Computer Science and Engineering, Vellore Institute of Technology, Chennai 600127, India
* Correspondence: prathiba.sbb@vit.ac.in

Abstract: The rapid development of Autonomous Vehicles (AVs) increases the requirement for the accurate prediction of objects in the vicinity to guarantee safer journeys. For effectively predicting objects, sensors such as Three-Dimensional Light Detection and Ranging (3D LiDAR) and cameras can be used. The 3D LiDAR sensor captures the 3D shape of the object and produces point cloud data that describes the geometrical structure of the object. The LiDAR-only detectors may be subject to false detection or even non-detection over objects located at high distances. The camera sensor captures RGB images with sufficient attributes that describe the distinct identification of the object. The high-resolution images produced by the camera sensor benefit the precise classification of the objects. However, hindrances such as the absence of depth information from the images, unstructured point clouds, and cross modalities affect assertion and boil down the environmental perception. To this end, this paper proposes an object detection mechanism that fuses the data received from the camera sensor and the 3D LiDAR sensor (OD-C3DL). The 3D LiDAR sensor obtains point clouds of the object such as distance, position, and geometric shape. The OD-C3DL employs Convolutional Neural Networks (CNN) for further processing point clouds obtained from the 3D LiDAR sensor and the camera sensor to recognize the objects effectively. The point cloud of the LiDAR is enhanced and fused with the image space on the Regions of Interest (ROI) for easy recognition of the objects. The evaluation results show that the OD-C3DL can provide an average of 89 real-time objects for a frame and reduces the extraction time by a recall rate of 94%. The average processing time is 65ms, which makes the OD-C3DL model incredibly suitable for the AVs perception. Furthermore, OD-C3DL provides mean accuracy for identifying automobiles and pedestrians at a moderate degree of difficulty is higher than that of the previous models at 79.13% and 88.76%.

Keywords: autonomous vehicular safety; 3D object detection; convolutional neural networks; 3D LiDAR sensor; camera sensor; fusing sensor data

Citation: Arikumar, K.S.; Deepak Kumar, A.; Gadekallu, T.R.; Prathiba, S.B.; Tamilarasi, K. Real-Time 3D Object Detection and Classification in Autonomous Driving Environment Using 3D LiDAR and Camera Sensors. *Electronics* **2022**, *11*, 4203. https://doi.org/10.3390/electronics11244203

Academic Editor: Donghyeon Cho

Received: 9 November 2022
Accepted: 8 December 2022
Published: 16 December 2022

1. Introduction

Autonomous Vehicles (AVs) safety and comfort of driving are significantly improving, howbeit decreasing the significance of common vehicles in the surroundings [1–3]. For developing such an AV, the sensor should keep track of the surrounding obstacles while driving, including their position, size, orientation, and classification of the object's circumstances [4–6]. The Light Detection and Ranging (LiDAR) sensor and camera sensors are used for this kind of environmental sensing [7–9]. To enhance object detection and recognize the three-dimensional (3D) shape of the object, 3D LiDAR was introduced [10,11].

Recently, the 3D LiDAR attracted researchers as it is a prominent sensor that helps AVs in their perception of the environment [12]. These sensors can recognize targets in the dark

and have a large field of view, precise information, and provide depth information [13,14]. 3D LiDAR holds significant benefits when compared with the camera sensors in object detection, including the ability to determine the position and shape of objects [15,16]. Still, the scattered 3D point clouds evolve into sparse distribution when the 3D LiDAR sensor [17–19] is distinct from the scan center, which turns out a concern in detecting certain objects in the classification process [20].

The camera sensors in the AVs act as vision sensors and produce hi-spectral images for classifying the images accurately. The classification techniques that are generally employed are rooted in extended intelligence algorithms such as deep learning [21,22]. In general, these methods use the design of an object to create boundary containers for recognizing the objects based on the environment [23,24]. However, camera sensors suffer from various lighting conditions levels and have insufficient knowledge of regions, directions, object shape, and structure, resulting in inaccurate object-area identifications [25,26].

For obtaining good accuracy in location and classification of obstacles in driving habitat, the feasible approach is to take the complementary data from the LiDAR sensor and camera. The 3D LiDAR sensor captures the 3D shape of the object and produces point cloud data that describes the geometrical structure of the object. The LiDAR-only detectors may be subject to false detection or even non-detection over objects located at a high distance. The sparse and chaotic distribution of the point clouds produced by LiDAR may lead the object detection methodology to lack in identifying smaller objects. The camera sensor captures RGB images with sufficient attributes that describe the distinct identification of the object. The high-resolution images produced by the camera sensor benefit the precise classification of the objects. When combining these two data, both the attributes and the geometrical information of the object helps in detecting real-time 3D objects in a more promising way. For this purpose, we propose an object detection mechanism that fuses the data received from the camera sensor and 3D LiDAR sensor (OD-C3DL). The mechanism fuses the point cloud and enhances the data for precise detection of the objects and we train the Convolutional Neural Networks (CNN) model with those data for accurate precision. Our main concern is to reduce the accidents between our vehicle and the pedestrians or the opposite vehicles around the driving environment.

The major contributions of the work are as follows

1. In OD-C3DL, the Point Cloud Augmentation (PCA) process estimates the depth information from the camera sensor data and coordinates the spatial information of the object, which enhances object identification to a greater extent.
2. The PCA process specifically uses the pre-trained Pyramid Stereo Matching Network (PSMNet), which exploits the global contextual information and extends the pixel-level features to region-level features with different scales of receptive fields to compute the disparity map.
3. The OD-C3DL applies VGG16 to implement ROI pooling next to the convolutional layers such as Conv3, Conv4, and Conv5, rather than just on the final convolutional layer, in which a feature tensor of fixed size is produced by each layer.
4. The OD-C3DL standardizes the attribute tensor utilizing L2 standardization and concatenates all the standardized attribute tensors, which ensures the detection system's reliability and scales the attribute associations arising out of several convolution layers to the equivalent size.
5. The OD-C3DL encompasses multiple task loss such as boundary container regression loss and classification loss functions for accomplishing object classification and boundary container regression at the training stage.

The rest of the paper is organized as follows: A survey of earlier similar research is presented in Section 2. A thorough explanation of the proposed OD-C3DL using data from a LiDAR sensor is provided in Section 3. The evaluation results and performance improvements of the OD-C3DL over the state-of-the-art methodologies are discussed in Section 4. In Section 5, the conclusions made from the observations are drawn.

2. Related Work

Typically, most object detection methodologies use cameras in 3D space to detect the object, which heavily relies on the anchor-based system. This process is expensive and causes delays in object detection in AV driving scenarios.

In [27], an anchor-free architecture is developed with 3D LiDAR object detection and proposed a dynamical fusion technique to act on images with point features through learning filters. Moreover, to investigate the overlapping region and for greater boundary container optimization, the Intersection over Union (IoU) loss is proposed. However, the model lacks in detecting small objects correctly such as pedestrians.

The authors in [28], tried to improve the object identifier of the Yolov5 model to perform object detection of tiny objects on the actual roads. The approach modifies the size of the output feature map and includes shallow high-resolution features so that the performance will be significantly improved. However, the approach does not concern the improvement of detection accuracy at night and under bad weather conditions. In [29], the authors tried to improve the performance and accuracy of the Yolov3 model by integrating self-attention and dilated convolution into the Yolov3 architecture. The approach is to reconstruct the loss function based on Floor IoU and focal loss that in turn improves the detection and the accuracy of Yolov3. By leveraging Vehicle-to-Infrastructure (V2I) communication, a framework is proposed for identifying the objects in the vehicular driving mode. However, the real-time performance of perception of the framework is poor [30]. A visibility enhancement scheme is proposed in [31], which has illumination enhancement, reflection component enhancement, and linear weighted fusion. The aim of the scheme is to improve AV driving under aggressive weather conditions such as sandstorms, heavy rain, or under heavy dust areas. The image restoration technique accompanied by the scheme achieves a great performance with detection accuracy while maintaining the tracking capability. However, the visibility enhancement scheme works better for recognizing vehicles rather than pedestrians.

The authors of [32], contribute the AV driving by intimating the selective attention mechanism of the human visual system. The authors used the Hidden Markov Model (HMM) for the lane-changing intention. However, the spatial context is not considered in the human visual system, which plays a major role in deciding the time at which the lane is to be changed. The authors in [33] aimed to maintain a safe distance with the pedal cyclists. With the help of the Received Signal Strength Indicator (RSSI) obtained from the Bluetooth device, the system identified the pervasive cyclists and improved the awareness of the situation to avoid collisions. It could be particularly useful in poor weather conditions where the visibility is low. However, the pose of the cyclist can only be predicted with the RSSI reference signals. The authors in [34] used exteroceptive sensors to detect the in-path objects and avoid collisions while riding on the hills or on some curved roads. The system used geo-referenced maps to identify the road geometrics and provide the desired velocity on those roads. The drawback of this approach mainly depends on the quality of the geo-referenced map. In [35], the Convolution-Transformer Network (CT-Net) is used as a unique deep learning technique to detect small objects. The attention-enhanced transformer block creates a multi-head self-attention system with enhanced features and improves the feature extraction ability of the model. In addition, a direct future fusion structure is presented to improve the detection accuracy of small objects and multi-scale objects.

The authors in [36] use millimeter-wave radar sensors for detecting the targets with zero-Doppler. These radars can be used for both long-range and short-range targets. The radars produce rage-angular azimuth radar images for long-range targets and 3D radar images for shorter-range targets. However, the shortcoming of this work is, it considered only the static targets. In [37], the authors proposed an off-the-shelf deep neural network architecture that is capable of detecting and recognizing the types of traffic signs and physical incidents. In [38], the authors aimed at detecting pedestrians using Field Programmable Gate Array (FGPA) during AV driving with the normalization-based validity index. Then, the Manhattan distance is calculated between the target histogram-

oriented gradient features and pedestrian histogram features. In [39], the monocular camera is used to detect the 3D object (M3D) by utilizing the deceptive depth and orientation representation using deep learning methodology. In the 3D space, the key points are detected from the object's center point. However, the M3D methodology has a limitation in detecting pedestrians and cyclists.

Thus, the existing system utilizes the merits of different sensors individually. However, the fusion of two different sensors for accurate object detection still has a place in the research. Moreover, the existing system lags in detecting the driving scenario objects such as cyclists, pedestrians, and cars during rainy and fog conditions. Hence, this paper fuses the 3D LiDAR sensor and camera sensor and proposes an effective object detection methodology named OD-C3DL. The 3D point clouds obtained from the 3D LiDAR sensor are used to identify the object region and its starting spot. The employed CNN extricates the quantified attributes from the object region and recognizes the corresponding object.

3. Proposed Work

The proposed OD-C3DL fuses the 3D point cloud data obtained from the 3D LiDAR sensor and image data obtained from the camera sensor. The overview of the proposed OD-C3DL is represented in Figure 1. The 3D point cloud data and the image data are considered as the input to the OD-C3DL. These data are fed into the PCA process, which is followed by the extraction and removal of floor points. The OD-C3DL then creates the outline of the object region and uses CNN for feature extraction and object classification to generate the desired accurate object identification. The OD-C3DL is comprised of two major components

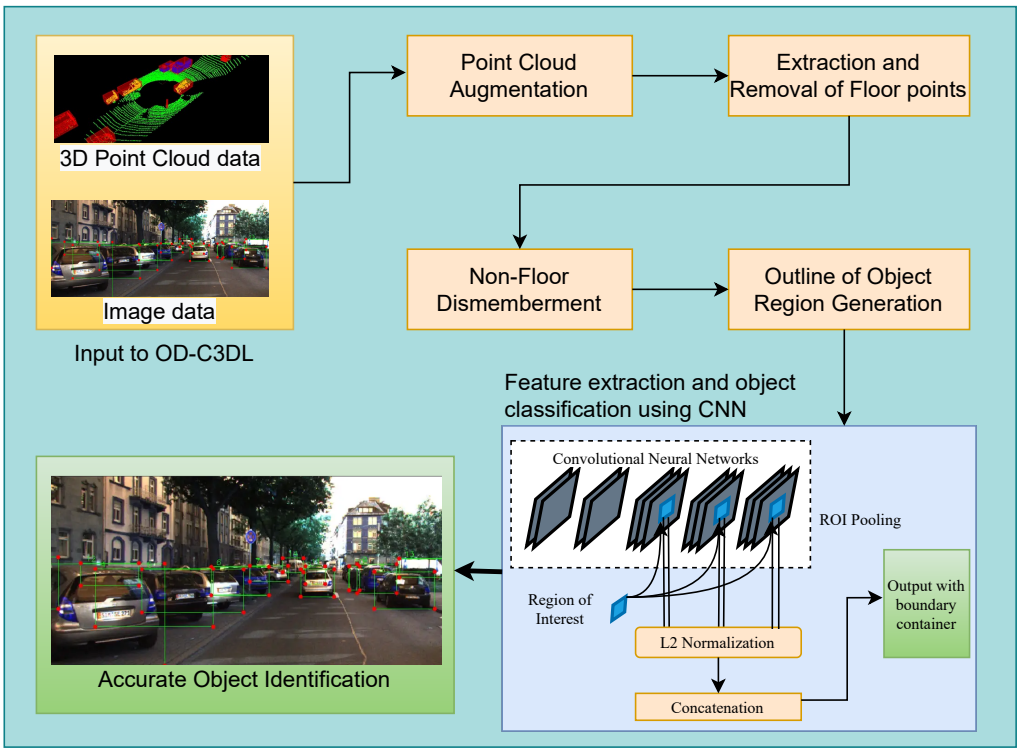

Figure 1. Overview of the proposed object detection methodology from the camera sensor and 3D LiDAR sensor data.

1. Augmentation of point clouds
2. Object region identification
3. Feature extraction and object classification using CNN

3.1. Point Cloud Augmentation

The PCA process in OD-C3DL enhances object identification to a greater extent. The PCA process estimates the depth information that is obtained from the camera sensor and coordinates the spatial information of the object. Thus, the PCA process tried to create a representation of the pseudo-object region from the images captured by the camera sensor to improve the features of the raw 3D LiDAR data.

The PCA process specifically uses PSMNet, which requires two input pictures to compute the disparity map of 375 × 1242. The PSMNet exploits the global contextual information and extends the pixel-level features to region-level features with different scales of receptive fields; the resultant combined global and local feature clues are used to form the cost volume for reliable disparity estimation. Thus, the PSMNet reduces the computational complexity. The following formulas are used to obtain each pixel's 3D coordinates from the left camera coordinate system,

$$p = \frac{a * (k - ek)}{M(k, v)} \tag{1}$$

$$q = \frac{r * (v - ev)}{fv} \tag{2}$$

$$r = \frac{fh * a}{M(k, v)} \tag{3}$$

where (p, q, r) stands for the 3D coordinate value that relates to each pixel (k, v) in the image plane. The terms ek, ev represent the location of the pixel, $M(k, v)$ display the disparity map formed by PSMNet, fh, fv denotes the vertical and horizontal focal length, and a is the horizontal balance between the pair of images.

The PCA then discards unusual height and reduces unnecessary noise interference from the pseudo point by setting the reluctance for each point to 1.0. The produced pseudo object region is represented by the notation $(p_i,\ q_i,\ r_i)$, where $i = (1, ...,\ P)$, where P represents the number of effective pseudo points (100 k∼400 k).

The generated dense pseudo object region is first sub-sampled in accordance with the calibration matrix, and then it is concatenated point-wise as $S = (p, q, r) \in \mathcal{R}^{N \times 3}$ with the LiDAR point clouds $O = (x,\ y,\ z) \in \mathcal{R}^{N \times 3}$. By taking into consideration the variations in axes permutation such as $S \oplus O$ and $O \oplus S$, a dual $N \times 6$ (here N represents the number of points clouds) point vectors are fed as input into the separate fully-connected structure to record global responses S_W and O_W in the high-dimensional feature space, respectively. The PCA process exploits the most important feature information by concatenating and compressing two 256-dimensional presentations into a single vector. The stimulator probability σ is used as a measuring variable to assess the feature channel's differentiability. In order to acquire the enhanced point output after segment-wise progress, then re-weight both point features according to the product operation. The entire procedure may be expressed numerically as,

$$S_W = W_2^S\ (W_1^s(S \oplus O)) \tag{4}$$

$$O_W = W_2^O\ (W_1^O(S \oplus O)) \tag{5}$$

$$D_E = \sigma\ S_W \oplus (1 - \sigma\ )\ O_W \tag{6}$$

where σ represents the softmax function, W_1^s and W_1^O represent the weight variables for fully connected layers, $\oplus$ represents segment-wise progression, and D_E signifies the outcome of enhanced point.

Thus, the pseudo point clouds give image semantics for the raw 3D LiDAR data enhancement feature. More precisely, the PCA process may adaptively re-weight the importance of various point channels, producing feature representations that are more robust and discriminative.

3.2. Object Region Identification (ORI) with 3D LiDAR Data

The proposed OD-C3DL uses the pseudo point clouds generated by the PCA process and the raw data obtained from the 3D LiDAR for Object Region Identification (ORI). The ORI module comprises three significant steps

1. Extraction and removal of floor points
2. Non-floor dismemberment
3. Outline of object region generation

3.2.1. Extraction and Removal of Floor Points

The floor points from the point cloud must be removed before object clustering to develop the object region outline effectively. The distance of nearby rings is far more susceptible than perpendicular displacement to the slope of the land.

The floor points in ORI are determined by the radius space between adjacent rings. To prevent inconsistent modifications in the range difference of the adjacent rings received through 3D LIDAR at unique sites, ORI employs the ratio of the differences of the actual measurement to the estimated measurement. Additionally, the distance between adjacent rings is not necessarily to be at a fixed position, it can be varied depending on the state of the street. The fact that the point clouds are represented by Cartesian coordinates is one of the great challenging issues. These point clouds require time-consuming operations such as searching and indexing. As an alternative, we code a construction of a multiple passage deep matrix N from a sparse point cloud P.

$$\mu(a_{i,j}) = (G_z, G_I, G_\alpha) \tag{7}$$

$$G_\alpha = \sqrt{G_x^2 + G_y^2} \tag{8}$$

where G_z, G_I, G_α stands for a point's altitude, intensity, and depth values, respectively.

It is assumed that, on a perfectly flat horizontal plane, the altitude of the LiDAR setup and the floor points, and every laser joint pitch angle are been sensed easily. As a result, it is possible to compute the expected intensity difference between the adjacent beams. With increasing surface elevation, the distinction in this range becomes less obvious.

Assume that $a_{i+1,j}$ indicates a specific field in the matrix μ, and $G_\alpha^{i+1,j}$ indicates the average intensity. The envisioned depth distinction of adjoining cells $(a_{i,j})$ and $(a_{i+1,j})$ in the equal column of matrix N is represented by the notation $\mathrm{F_d}(a_{i,j},\ a_{i+1,j})$. A concentric circle will be formed by the LIDAR points of the nearby scan lines at the plane converting $\mathrm{F_d}(a_{i,j},\ a_{i+1,j})$ as a steady whose cost depends on the set peak of LiDAR as well as the pitch angle of the adjacent it i and $i+1$th scan lines in the vertical route. $\mathrm{N_d}(a_{i,j},\ a_{i+1,j})$ represents the actual measured difference in depth between $a_{i,j}$ and $a_{i+1,j}$.

The parameters involved in the process of extraction and removal of floor points, where F_d is the anticipated range dissimilarity and N_d is the actual range dissimilarity among two 3D LiDARs, are represented in Figure 2. To calculate the apparent variation between the measured and estimated intensity fluctuations, the depths of the 3D LiDAR points in the fields $a_{i+1,j}$ of the matrix are shortened by the objects, which causes a quick drop in the intensity distance of the two neighboring fields $a_{i,j}$ and $a_{i+1,j}$. In order to establish whether the field points $a_{i+1,j}$ are floor points or hurdle points, this model can analyze the values of $\mathrm{F_d}(a_{i,j},\ a_{i+1,j})$ and $\mathrm{N_d}(a_{i,j},\ a_{i+1,j})$. The LiDAR point cloud is almost evenly spread over the floor, the greater the distance between adjoint bodies and the LiDAR base leads to the better $\mathrm{F_d}(a_{i,j},\ a_{i+1,j})$ values. This span fluctuates depending on the role since the exact difference between $\mathrm{F_d}(a_{i,j},\ a_{i+1,j})$ and $\mathrm{N_d}(a_{i,j},\ a_{i+1,j})$ lies between $[0,\ \mathrm{F_d}(a_{i,j},\ a_{i+1,j})]$. It is also challenging to determine an appropriate threshold for classifying the LiDAR factor cloud; however, the proportionate range of $\mathrm{F_d}(a_{i,j},\ a_{i+1,j})$ and $\mathrm{N_d}(a_{i,j},\ a_{i+1,j})$ at any function is often [0,1]. In order to prevent fluctuations within the intensity differences of adjoint 3D LiDAR laser lines in unique places, we implement a proportional technique. Therefore, using Equation (9), it is possible to compute the fundamental fact of the field $a_{i+1,j}$.

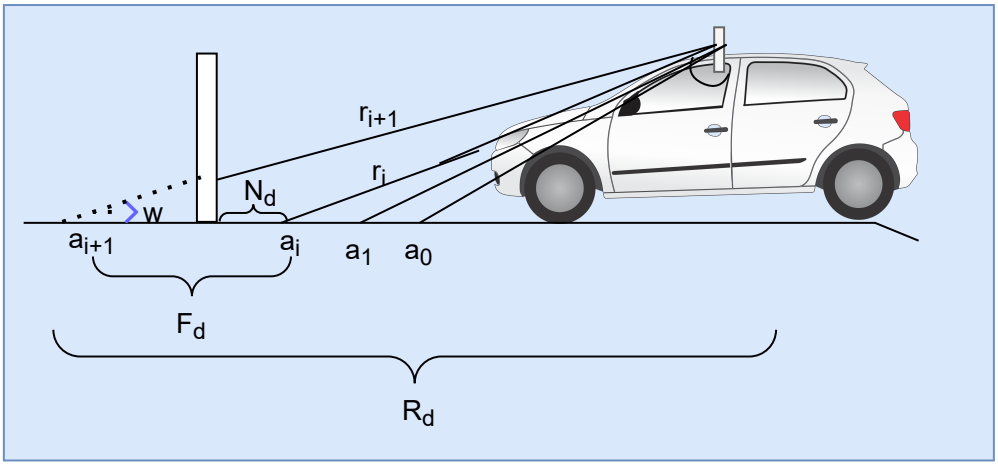

Figure 2. Parameters involved in the process of extraction and removal of floor points, where F_d is the anticipated range dissimilarity and N_d is the actual range dissimilarity among two 3D LiDARs.

$$G\left(a_{i+1,j}\right) = \frac{\mathrm{N_d}\left(a_{i,j},\ a_{i+1,j}\right)}{\mathrm{F_d}\left(a_{i,j},\ a_{i+1,j}\right)} \tag{9}$$

such that,

$$\mathrm{N_d}\left(a_{i,j},\ a_{i+1,j}\right) = G_{\alpha}^{i+1,j} - G_{\alpha}^{i,j} \tag{10}$$

In order to make the OD-C3DL technique relevant to varied road environments, the expanded line of the LiDAR axis is not always considered to be at the right angle to the floor level when estimating the predicted depth distance of adjoining scanning lines. In this case, the LiDAR extension line and the floor's angle are referred to as varying parameters that change with the AV's tilt gradient. According to the AV's pitch angle, a variable ρ is defined to represent the angle between the floor surface and the LiDAR axis adjunct line. Thus, $\mathrm{F_d}\left(a_{i,j},\ a_{i+1,j}\right)$ can be estimated as,

$$\frac{\mathrm{F_d}\left(a_{i,j},\ a_{i+1,j}\right)}{\sin\Delta\tau} = \frac{l}{Sin\ \dot{\omega}} \tag{11}$$

where $\dot{\omega}$ serves as the degree bounded by the $i+1$th scan line and the floor surface, which can be calculated using (12)

$$\dot{\omega} = \pi - \tau_{i+1} - \rho \tag{12}$$

where τ_{i+1} represents the $i+1$th scan line's vertical pitch angle. The gradient bounded by the floor area and the LiDAR axis adjunct line (ρ) can be computed as,

$$\frac{b_i}{\sin\rho} = \frac{\mathcal{B}_d}{\sin\tau_i} \tag{13}$$

$$B_d^2 = l^2 + b_i^2 - 2lb_i\cos\tau_i \tag{14}$$

where b_i serves as the radical distance of the points in the field $a_{i,j}$. From the above equations, the value of $\mathrm{F_d}\left(a_{i,j},\ a_{i+1,j}\right)$ can be estimated as,

$$\mathrm{F_d}\left(a_{i,j},\ a_{i+1,j}\right) = \frac{b_i\sin\Delta\tau}{\sin\left[arc\,\sin\left(\frac{l\sin\tau_i}{\sqrt[2]{l^2 + b_i^2 - 2lb_i\cos\tau_i}}\right) - \tau_i + 1\right]}$$

Using the aforementioned method, the OD-C3DL model linearly extracts all the floor field values from the matrix, and then we apply the following equations to convert each floor field into a point cloud.

$$G_x = \frac{Gz}{\sin(\Delta\tau \times \varepsilon)} \times \cos(\Delta\tau \times i) \times \cos(\Delta\tau \times j) \tag{15}$$

$$G_y = \frac{Gz}{\sin(\Delta\tau \times \varepsilon)} \times \cos(\Delta\tau \times i) \times \cos(\Delta\tau \times j) \tag{16}$$

$$Gz = Gz \tag{17}$$

3.2.2. Non-Floor Dismemberment

The remaining point clouds need to be further segmented after the floor points have been changed. The traits of 3D LiDAR points include sparsity, disorder, and non-uniformity. The likelihood of over-fit and under-fit partitioning of non-floor points will grow due to the inadequate and the variance in the points.

The Non-Floor Dismemberment (NFD) in OD-C3DL depends on the azimuth sequence and the distance factors. The NFD initially divides the non-clustered points and applies a threshold with minimal azimuth variations to cluster the non-clustered non-floor points. The clustered non-floor points are then dismembered by using the robust threshold mechanism. The process of dismembering the non-floor points is given in Algorithm 1. At the beginning of the algorithm, the initialized points are grouped as the main cluster. Since the 3D LiDAR scans and provides the information in azimuth sequence, the northern degree of the LiDAR point striking a similar object is continually disseminated. Whenever the dissimilarity in azimuth among the two locations is smaller than the limit, then it is considered that they may have originated from identical objects. Thus, the NFD algorithm initially groups the non-floor points into the cluster group $CL = cl_1, cl_2, \ldots, cl_n$ in accordance with the azimuth alterations between factors. We then determine the exact difference of azimuth $\Delta\varphi(G_i, G_j)$ and compare it to the alternate point G_j inside the point set cl_k for a factor $g_i \in G(i > 1)$ that is not allocated to any alternate clusters. If the variation is less than the similarity θ_ς, we can attach point G_i to the cluster cl_k as both the points are originated from the single northern region as similar the cluster cn. In all other circumstances, a new cluster cl_k may be constructed and factor G_i added to it. Based on the difference in azimuth between the points after each point, entire points persisted into clusters.

The Euler distance in bounded with two points (G_i, G_j) in the cluster $\varpi(G_i, G_j)$ is first evaluated. The distance is then compared with the threshold value d_{th}, if it is lesser then the point G_i will be included in the *Te_cl* cluster. If the value is greater, *Te_cl* will be combined with the non-floor cluster $\mathbb{R}$. Then, the points in the *te_cl* will be removed and G_i will be included in *te_cl* to enable the dismembering of the new object. Thus, each cluster in the set $CL = cl_1, cl_2, \ldots, cl_n$ will be dismembered into individual object.

Algorithm 1 Cluster-based non-floor point dismemberment Algorithm.

Input: Non-floor points ($G_{non-floor}$) floor partitioning methodology, azimuth limit difference ($t\Theta_{similarity}$)
Output: Decisive set of cluster non-floor points (R)

Initialize a Cluster based on angular azimuth $(cl) \leftarrow \varnothing$
Initialize a Cluster based on the list of angular azimuths $(CL) \leftarrow \varnothing$
Initialize a Temporary non-floor point cluster $(te\_cl) \leftarrow \varnothing$
for G_i in G **do**
  if G_i is the first non-floor point **then**
    $CL \leftarrow cl \cup G_i$
  else
    for G_j in c **do**
      $\Theta \leftarrow \Delta\varphi(G_i, G_j)$
      if $\theta < \theta_\varsigma$ **then**
        $cl \leftarrow cl \cup G_i$
      else
        $CL \leftarrow \{cl\} \cup CL$
        $cl \leftarrow \varnothing$
        $cl \leftarrow cl \cup \{G_i\}$
      end if
    end for
  end if
end for
for cl in CL **do**
  for G_i in cl **do**
    for G_j in cl **do**
      $D \leftarrow \varpi(G_i, G_j)$
      $d_{th}(G_i) \leftarrow \varpi_{th}(G_i)$
      if $d < d_{th}(G_i)$ **then**
        $Te\_cl \leftarrow te\_cl \cup \{G_i\}$
      else
        $\mathbb{R} \leftarrow \{te\_cl\} \cup \mathbb{R}$
        $te\_cl \leftarrow \varnothing$
        $te\_cl \leftarrow te\_cl \cup \{G_i\}$
        Eliminate G_i from cl
      end if
    end for
  end for
end for

3.3. Feature Extraction and Object Classification Using CNN

The OD-C3DL uses CNN for extracting the features and classifying the objects from the boundary containers generated from the ORI module. The main objective is to find the objects that were caught under difficult circumstances with wildly different object sizes. Although earlier regional CNN models, such as Fast-Regions with CNN features (Fast-RCNN) [40], did not need the boundary containers to possess definite dimensions, it is still challenging to robustly identify miniature objects using these models. This is mostly because these models only display a Region of Interest (ROI) grouping in the final object map. The final layer of convolutional components, however, has relatively insufficient information about the object after several pooling and convolution procedures for the aspirant region of the small object. In these conditions, even when an object is present in the candidate locations, it is challenging to locate and recognize it based on this attribute. Thus, OD-C3DL employs the CNN model not to apply the ROI pooling solely over the

ending convolutional feature association. Instead, the ROI pooling process is carried out in each layer after the region proposal is projected onto several feature map layers.

To implement ROI pooling next to the Conv3, Conv4, and Conv5 layers rather than just on the final convolutional layer, OD-C3DL applies VGG16 [40], in which a feature tensor of fixed size is produced by each layer. To ensure the detection system's reliability and to scale the attribute associations arising out of several convolution layers to the equivalent size, the OD-C3DL standardizes the attribute tensor utilizing L2 standardization and concatenates all the standardized attribute tensors. The workflow of the CNN architecture in the proposed OD-C3DL is given in Figure 3. The input image is given to the CNN, which has various convolutional layers. The 3D point cloud generates an object proposal region, which then acts as the ROI. The ROI is fed into the convolutional layers 3, 4, and 5 as ROI pooling. The pooled ROIs then undergo L2 normalization, which is concatenated to find the output with the boundary container. Every pixel of the feature maps is subjected to standardization, and each feature map is handled separately. This standardization process is described as follows:

$$\eta = \frac{\eta}{\|\eta\|_2} \tag{18}$$

$$\|\eta\|_2 = \sqrt{\sum_{i=1}^{d} |\eta_i^2|} \tag{19}$$

where η denotes the real attributes and $_\eta$ denotes the normalized attributes.

$$z_i = \eta_i \lambda_i \tag{20}$$

where z_i indicates the value of the rescaled attribute. The backpropagation principle states that the measuring element λ_i is addressed as,

$$\frac{dh}{d\eta} = \frac{\lambda \times dh}{dz} \tag{21}$$

$$\frac{dh}{d\eta} = \lambda \frac{dh}{d\eta} \left(\frac{I}{||\eta||_2} - \frac{\eta\eta}{||\eta||_2^3} \right) \tag{22}$$

$$\frac{dh}{d\lambda_i} = \sum_{z_i} \frac{dh}{zi} \lambda_i \tag{23}$$

where $z\left[z_1, z_2, \ldots, z_j\right]^T$.

We utilize 1×1 convolution to minimize the linked feature dimensions and maintain the ROI pooling feature map's original size. The two fully connected linked layers are then given the final feature tensor for use in object placement and recognition.

In order to accomplish object classification and boundary container regression in favor of ROI at the training stage, the OD-C3DL encompasses the multiple task loss (boundary container regression loss and classification loss) functions. Two components make up the network model's output. A vector with $M + 1$ dimensions represents the likelihood dissemination of the classification to which the image sample is associated, where $q = \{q_1, q_2, \ldots, q_m\}$. The predicted location of the bounding container for each of the M object classes is represented by a vector with four parameterized coordinates, denoted as $c = \left\{c_{clx}, c_{cly}, c_w, c_l\right\}$ in one of the other outputs. The notations c_{clx}, c_{cly}, c_w, and c_l signify the two coordinates, the width, and the height, respectively, of the predicted bounding container center. Thus, the proposed OD-C3DL identifies the objects effectively in the AV environment.

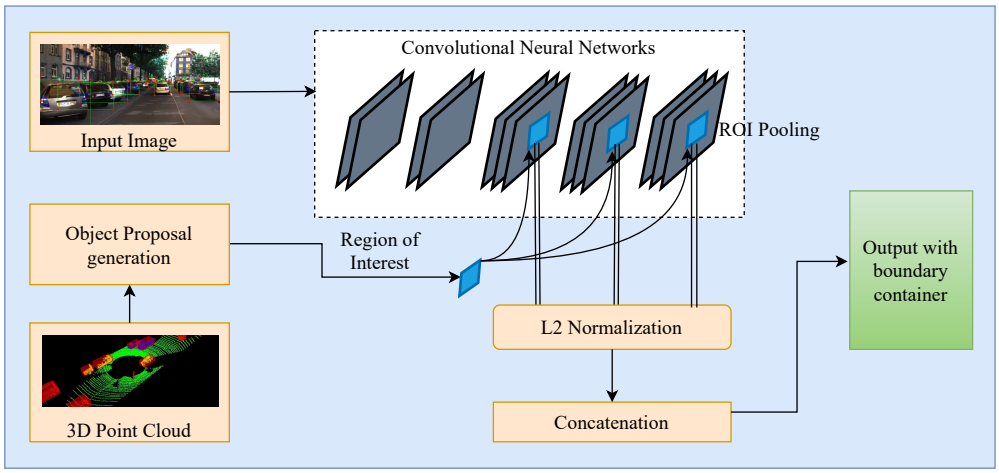

Figure 3. Workflow of the convolutional neural network architecture in the proposed OD-C3DL.

4. Evaluation of the Proposed Work

For effectively identifying the objects, the experiments were performed using the 2012 KITTI Object identification benchmark [41]. The evaluations are then used to assess the effectiveness of the OD-C3DL. The dataset encompasses coincident images from the camera sensor and 3D LiDAR images taken by the AV. The camera image is cropped and modified pixel by pixel. In particular, 3D LiDAR images are acquired by the HDL-64E LiDAR sensor with 68 scan lines capable of 400 scans. When the sensor rotates at a repetition of 11 Hz, it can create more than 10^6 points at each second. The dataset gives 7484 images for learning and 7528 images for evaluation. As the label was not disclosed in the test set, the learning data was split into a subset to train a model (75%) and a subset to test a model (25%). The learning data contains nine distinct classes with 51,962 labels: "AV", "Pedestrian", "Cyclist", "Van", "Truck", "Sitting Person", "Road Surface". "Public bus", "Other", "Never care" and showed different AV circumstances. We categorized the object samples in the KITTI dataset into ternary categories, namely low, medium, and hard levels of difficulty, depending on the dimension of the 2D boundary container on the image environment and in the obstruction circumstances. The object detection achieved from the trained CNN model from the KITTI benchmark dataset is represented in Figure 4.

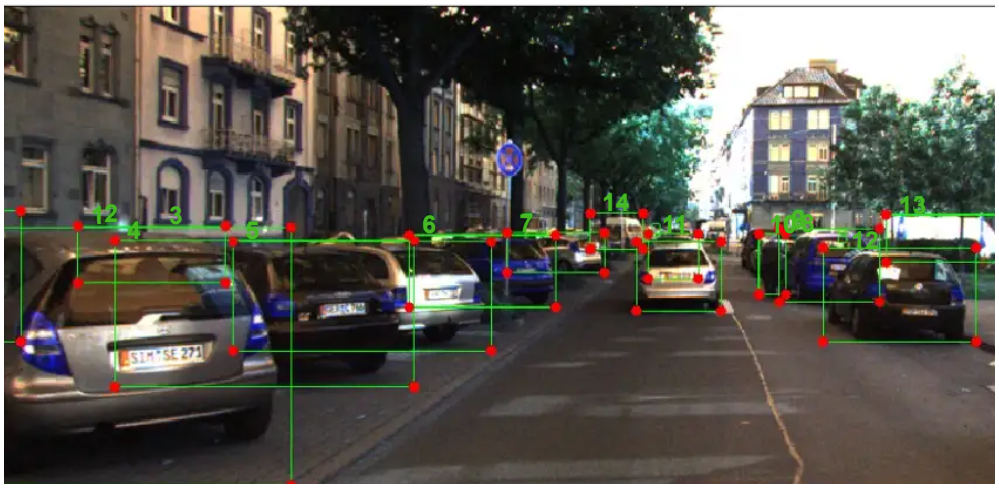

Figure 4. Object detection from the trained CNN model from the KITTI benchmark dataset. The object detection in this figure is represented using the boundary boxes that occupy the various objects such as parking vehicles, moving vehicles.

4.1. Performance Analysis of Proposed Methodology

4.1.1. Analysis of Precision Recall Curve

To evaluate the average accuracy and test time, we employed Fast-RCNN [42] that learns the image features and assesses the performance of OD-C3DL in ORI. The precision–recall curve obtained in the evaluation is displayed in Figures 5 and 6. The network was optimized for object identification using the KITTI dataset as a training and validation subset. Three categories such as AVs, persons, and environment were considered in the training stage. We used IoU as the object detection criterion and followed KITTI's evaluation. If the detection's boundary container in the image space overlapped the floor truth by at least 50%, it was considered valid. For an average calculation accuracy, we used the PASCAL VOC [43] evaluation toolkit.

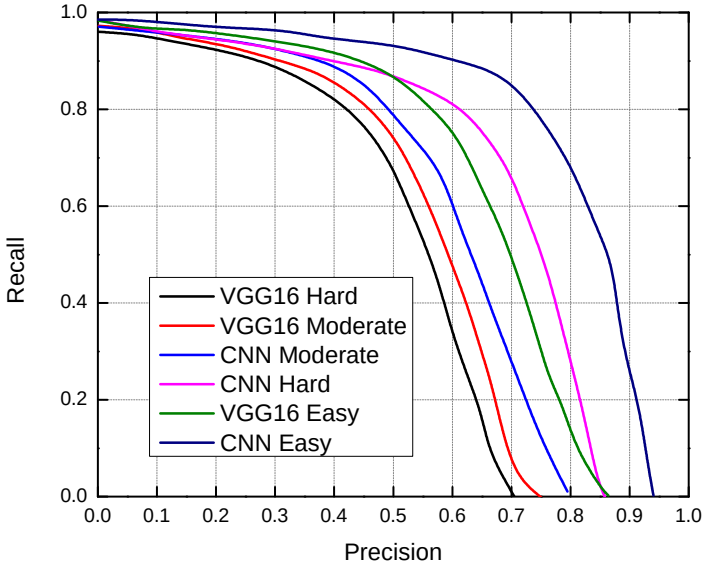

Figure 5. Three object classes with three varying degrees of complexity were examined using our CNN model, the VGG16 model, and its Precision–Recall Curve.

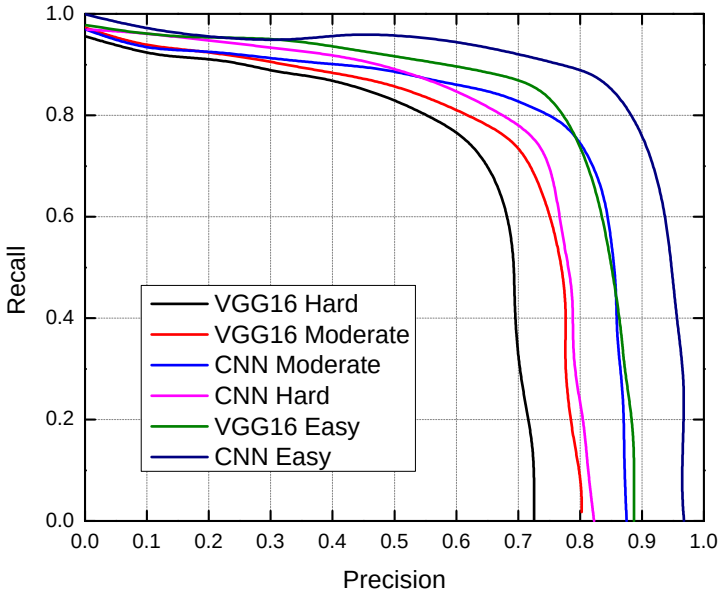

Figure 6. Three object classes with three varying degrees of complexity were examined using our CNN model, the VGG16 model, and its Precision–Recall Curve.

4.1.2. Analysis of Accuracy Gains

We evaluated a total of 7488 images from the KITTI validation and training datasets to estimate the duration of OD-C3DL. We observe that the mean duration is relatively 64.79 ms, which indicates that the frame rate of OD-C3DL is higher than that of LiDAR. This demonstrates that OD-C3DL can be applied for faster and online frames. On the KITTI validation dataset, we then contrasted the performance of the OD-C3DL method with a few state-of-the-art approaches, including Yolov5, Yolov4, Yolov3, CT-NET, and M3D. Table 1 lists the average accuracy results of the difficulty levels of easy, hard, and medium as well as the duration of each technique. We outperformed the majority of existing object identification techniques with average accuracy values of 88.96% and 77.18% in a medium level for AVs and persons, with a mean running time of approximately 65 ms. This is hard evidence that the OD-C3DL strategy has performed better than state-of-the-art approaches in terms of results. The proposed OD-C3DL approach may still perform accurate detection while gathering the distance of the target information, despite the fact that there are some strong obstacles and small objects in the image.

Table 1. The enhanced Three-Dimension detection capabilities offered by several LiDAR signals and feature combination techniques on the KITTI dataset. The accuracy gains are indicated via brackets.

Components				**Car AP(%)**		
3D LiDAR	**Camera**	**Fusion**	**PCA**	**Easy**	**Moderate**	**Hard**
0	0	0	0	87.07	77.82	72.28
1	0	1	0	87.39(+0.39)	78.57(+1.20)	73.10(+0.95)
1	0	0	1	87.56(+0.52)	78.94(+1.23)	73.48(+1.33)
0	1	1	0	87.47(+0.42)	78.89(+0.99)	73.47(+1.22)
0	1	0	1	87.79(+0.59)	79.54(+1.42)	73.84(+1.67)

4.1.3. Analysis of Elapsed Epochs vs. Computed Loss for the CNN Model

Figure 7 shows the validation loss and training loss of CNN in the correspondence of computed loss values vs elapsed epochs. From the figure, we can observe that the validation loss and training loss are decreasing exponentially, which concludes that the CNN has capable of identifying the objects effectively.

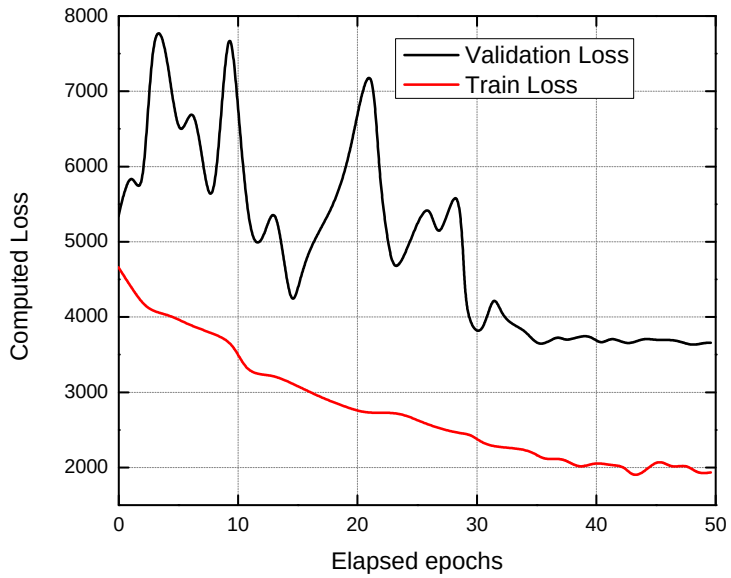

Figure 7. Graph representing the Elapsed epochs Vs Computed loss for the CNN model used In Object detection.

4.1.4. Analysis of Object Detection under Various States

Table 2 demonstrates the 3D LiDAR and PCA combo achieves an extraordinary increase in performance by 0.59%, 1.42%, and 1.67% in easy, moderate, and hard levels. Using 3D LiDAR with PCA, the performance is slightly lower than OD-C3DL, which means that the semantics of a channel in an image would be complementary to the 3D LiDAR geometric information, and more advantageous than 3D object detection is the pseudo signal. Additionally, the combination of the camera sensor with 3D LiDAR results with straightforward feature chaining is increased slightly and illustrates how significance weighting mechanisms or PCAs might be useful for improving detection accuracy and learning to differentiate more features.

Table 2. Each component of the suggested method's performance gain is listed between brackets. 1 means "enabled", while 0 means "disabled".

Components			**Car AP (%)**		
PCA with ORI	**CNN**	**IoU Head**	**Easy**	**Moderate**	**Hard**
0	0	0	87.07	77.84	72.28
1	0	0	87.69(+0.57)	79.25(+1.42)	73.92(+1.56)
1	1	0	89.21(+2.09)	81.78(+3.94)	77.34(+4.85)
1	1	1	90.43(+3.33)	83.27(+5.41)	78.93(+6.73)

4.2. Comparative Analysis of Proposed Methodology

The performance of the ORI phase on the 2D callback of accurate interpretation was first evaluated. The proposed item was projected onto the 2D picture plane using the calibration file, and the out-of-image detection was ignored. The IoU metric is employed in this instance to compare the results obtained from the object area on three different levels of difficulty.

4.2.1. Comparative Analysis of Recall vs. IoU

In the evaluation, the OD-C3DL's popularity results are compared to those of many existing approaches, such as the Yolov5 [28], Yolov4 [44], Yolov3 [29], CT-Net [35], and M3D [39]. When producing various object regions, we compared the recall rates of all methods and displayed the results in Figure 8. The recall rate was plotted using the proposal for a region of 1000 objects, which is the function of the IoU threshold. As seen, the OD-C3DL solution offered a recall rate of greater than 95% across the board for IoUs. The primary cause of this outcome is that all fundamental approaches produce proposals for object areas from the 2D visual space, where it is challenging to identify the faraway object regions because they frequently overlap. The object depth feature, however, allows the object area to be differentiated in the 3D point cloud created by 3D LiDAR. Further, the area where the outline is proposed based on visual information may provide the width only a rough boundary container position. Therefore, the recognition value drops promptly with increasing overlap, but the 3D LiDAR captures the pose and the laser scan effectively uses the spatial coordinates of the object to determine the shape of the object that was detected. It has a clear advantage over the camera in that it holds point clouds.

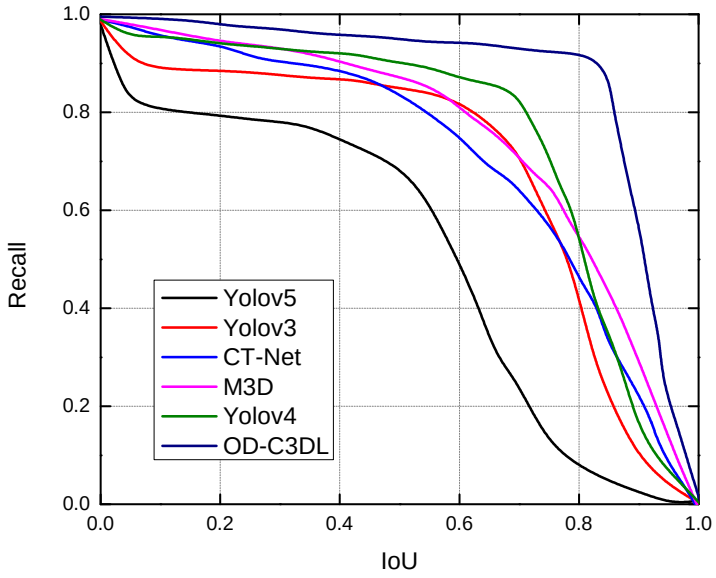

Figure 8. Comparative analysis of OD-C3DL with existing mechanisms for measuring the recall by varying the IoU.

4.2.2. Comparative Analysis of Average-Precision

Table 3 shows the Average-Precision (AP) calculated using earlier LiDAR- and image-based techniques on the KITTI dataset. Our object-region design OD-C3DL produced 88 non-duplicate designs on an average per frame which is less than the designs produced by existing methods as 2010 frames. We acquired an estimated average accuracy of 89% for the car category using OD-C3DL which is greater than the achieved value using the majority of state-of-the-art methods because it produced fewer errors and a faster recall speed. At the same time, we greatly decreased computation time while outperforming existing approaches in every intermediate-level category, with 90.8%, 72.7% for cars, and 73.7% for pedestrians. This demonstrates strongly that object-level object areas may be precisely extracted from a 3D LiDAR point cloud. Using the produced region design as input and the real VGG16 model as the default, we used CNN to assess the OD-C3DL efficiency. The experiment used the KITTI dataset to train the proposed CNN model, and the object categories were background, AVs, and pedestrians.

Table 3. On the KITTI dataset, Average-Precision (AP) (percent) was calculated using earlier LiDAR- and image-based techniques. LiDAR and camera sensors are indicated by the letters Li and Ca, respectively.

		Cars			**Pedestrians**			
Approach	**Sensor**	**Easy**	**Moderate**	**Hard**	**Easy**	**Moderate**	**Hard**	**Run-Time (ms)**
Yolov5	Li	88.01	78.16	77.97	-	-	-	167
Yolov3	Li	92.93	83.15	78.66	41.18	35.42	33.96	64
CT-Net	Li	95.87	91.15	81.10	86.67	71.17	66.14	472
M3D	Li+Ca	95.84	95.18	85.49	89.86	80.10	75.09	171
OD-C3DL	Li+Ca	96.74	89.75	86.49	90.12	80.13	75.66	65

4.2.3. Comparative Analysis of Sensitivity and Average Precision

With each learning session, this approach is adaptively modified in accordance with the slope of the loss function to quicken convergence. With batch sizes of 16 and a momentum coefficient of 0.9, we utilized the Nesterian Accelerated Gradient (NAG) optimizer to calibrate the CNN model, leaving the parameters of the first two sets of convolutional layers untouched and adjusting the other layers with a maximum iteration count of 200,000. We evaluated the item detection capabilities of models and the fundamental methodology

on KITTI datasets after training with the industry-standard accuracy of the recall curve. We used the evaluation tool PASCAL VOC AP calculation kit and applied it to the KITTI grading benchmark. The average accuracy value is the area under the accuracy of the recall curve. A comparison of the accuracy of recall curves demonstrates that OD-C3DL consistently outperformed the existing approaches despite the increasing difficulty for each degree of complexity with three object categories. The sensitivity vs false positive per frame for OD-C3DL with the existing techniques is plotted in Figure 9. This result shows that by layering more convolutional elements, information loss can be reduced. The results also demonstrate that the integration of several convolution layers reduces the dropout layer that nullifies the data and small objects can be identified more efficiently. The average precision calculated against the threshold of IoU of the CNN model is plotted in Figure 10.

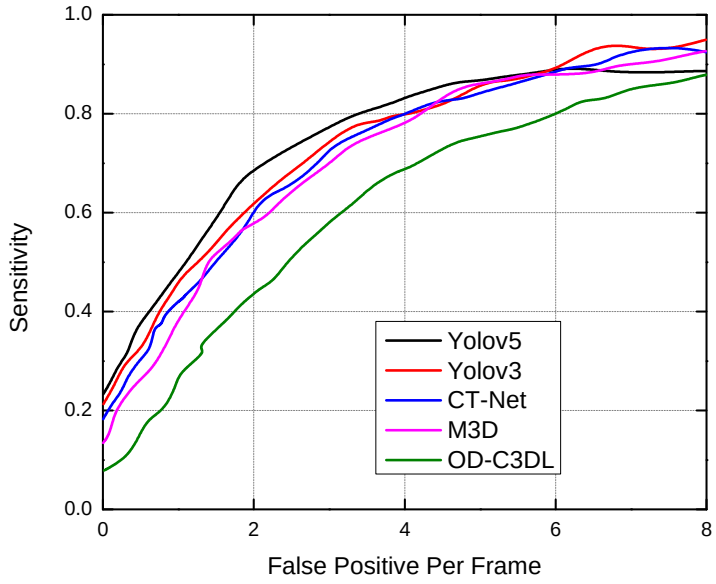

Figure 9. Sensitivity vs. False Positive Per Frame for multiple models.

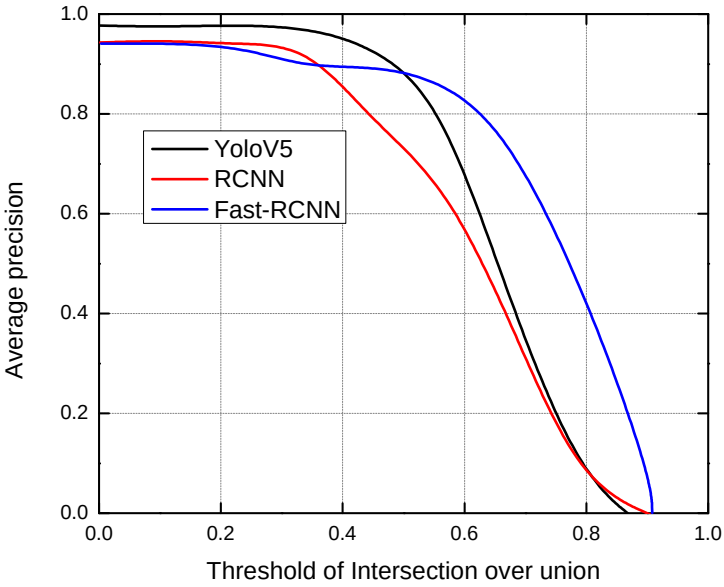

Figure 10. Average precision vs. Threshold of Intersection over union for our CNN model.

5. Conclusions

In this paper, using the complementary nature of the 3D LiDAR and camera sensor data, we have suggested a unique and potent object identification method OD-C3DL that successfully identifies various objects that are present in the surroundings of the AV. OD-C3DL outperformed most of the existing approaches in terms of accuracy, and reachability with 88.76% and 79.13%, respectively, for quite challenging detection of AVs and people nearby. OD-C3DL, which refers to quick detection, had an average run time for a frame of roughly 65ms. We achieved a better result when we implemented it online, and it is a strong rival to other well-known models.

However, our proposed OD-C3DL lags in the prediction accuracy of cyclists with various feature ambiguities. To address this problem in the future, we can include the Probabilistic Neural Network (PNN) in the OD-C3DL approach for extracting the features effective from each modality. In addition, image segmentation can be performed via a clustering approach.

Author Contributions: Conceptualization, K.S.A. and S.B.P.; methodology, S.B.P. and K.T.; validation, K.T.; data curation, K.T.; writing—original draft preparation, K.S.A. and A.D.K.; writing—review and editing, T.R.G. and S.B.P.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: KITTI is a publicly available dataset. KITTI consists hours of traffic scenarios recorded with a variety of sensor modalities,grayscale stereo cameras, and a 3D laser scanner. The KITTI dataset is obtained from https://paperswithcode.com/dataset/kitti and accessed on 9 September 2022.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AVs	Autonomous Vehicles
LiDAR	Light Detection and Ranging
3D LiDAR	Three-Dimensional Light Detection and Ranging
ROI	Regions of Interest
IoU	Intersection over Union
CNN	Convolutional Neural Networks
V2I	Vehicle-to-Infrastructure
HMM	Hidden Markov model
RSSI	Received signal strength indicator
CT-Net	Convolution Transformer Network
FGPA	Field Programmable Gate Array
M3D	Monocular camera is used to detect the 3D object
PCA	Point Cloud Augmentation
PSMNet	Pyramid Stereo Matching Network
ORI	Object Region Identification
NFD	Non-Floor Dismemberment
NAG	Nesterian Accelerated Gradient
Fast-RCNN	Fast-Regions with CNN
AP	Average-Precision
PNN	Probabilistic Neural Network

References

1. Lee, S.; Lee, D.; Choi, P.; Park, D. Accuracy–power controllable LiDAR sensor system with 3D object recognition for autonomous vehicle. *Sensors* **2020**, *20*, 5706. [CrossRef] [PubMed]
2. Francies, M.L.; Ata, M.M.; Mohamed, M.A. A robust multiclass 3D object recognition based on modern YOLO deep learning algorithms. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e6517. [CrossRef]

3. Gupta, B.B.; Gaurav, A.; Marín, E.C.; Alhalabi, W. Novel Graph-Based Machine Learning Technique to Secure Smart Vehicles in Intelligent Transportation Systems. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–9. [CrossRef]
4. Prathiba, S.B.; Raja, G.; Anbalagan, S.; Dev, K.; Gurumoorthy, S.; Sankaran, A.P. Federated Learning Empowered Computation Offloading and Resource Management in 6G-V2X. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3234–3243. [CrossRef]
5. Zhang, X.; Xia, X.; Liu, S.; Cao, Y.; Li, J.; Guo, W. An Integrated Framework on Autonomous-EV Charging and Autonomous Valet Parking (AVP) Management System. *IEEE Trans. Transp. Electrif.* **2022**, *8*, 2836–2852. [CrossRef]
6. Deb, S.; Carruth, D.W.; Hudson, C.R. How communicating features can help pedestrian safety in the presence of self-driving vehicles: Virtual reality experiment. *IEEE Trans. Hum.-Mach. Syst.* **2020**, *50*, 176–186. [CrossRef]
7. Zhao, L.; Xu, S.; Liu, L.; Ming, D.; Tao, W. SVASeg: Sparse Voxel-Based Attention for 3D LiDAR Point Cloud Semantic Segmentation. *Remote Sens.* **2022**, *14*, 4471. [CrossRef]
8. Prathiba, S.B.; Raja, G.; Anbalagan, S.; Arikumar, K.S.; Gurumoorthy, S.; Dev, K. A Hybrid Deep Sensor Anomaly Detection for Autonomous Vehicles in 6G-V2X Environment. *IEEE Trans. Netw. Sci. Eng.* **2022**, 1–10. [CrossRef]
9. Zhao, C.; Fu, C.; Dolan, J.M.; Wang, J. L-shape fitting-based vehicle pose estimation and tracking using 3D-LiDAR. *IEEE Trans. Intell. Veh.* **2021**, *6*, 787–798. [CrossRef]
10. Song, W.; Li, D.; Sun, S.; Zhang, L.; Xin, Y.; Sung, Y.; Choi, R. 2D&3DHNet for 3D object classification in LiDAR point cloud. *Remote Sens.* **2022**, *14*, 3146.
11. Iftikhar, S.; Asim, M.; Zhang, Z.; El-Latif, A.A.A. Advance generalization technique through 3D CNN to overcome the false positives pedestrian in autonomous vehicles. *Telecommun. Syst.* **2022**, *80*, 545–557. [CrossRef]
12. Dai, D.; Wang, J.; Chen, Z.; Zhao, H. Image guidance based 3D vehicle detection in traffic scene. *Neurocomputing* **2021**, *428*, 1–11. [CrossRef]
13. Prathiba, S.B.; Raja, G.; Dev, K.; Kumar, N.; Guizani, M. A Hybrid Deep Reinforcement Learning For Autonomous Vehicles Smart-Platooning. *IEEE Trans. Veh. Technol.* **2021**, *70*, 13340–13350. [CrossRef]
14. Fernandes, D.; Afonso, T.; Girão, P.; Gonzalez, D.; Silva, A.; Névoa, R.; Novais, P.; Monteiro, J.; Melo-Pinto, P. Real-Time 3D Object Detection and SLAM Fusion in a Low-Cost LiDAR Test Vehicle Setup. *Sensors* **2021**, *21*, 8381. [CrossRef] [PubMed]
15. Ye, X.; Shu, M.; Li, H.; Shi, Y.; Li, Y.; Wang, G.; Tan, X.; Ding, E. Rope3D: The Roadside Perception Dataset for Autonomous Driving and Monocular 3D Object Detection Task. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 21341–21350.
16. Nebiker, S.; Meyer, J.; Blaser, S.; Ammann, M.; Rhyner, S. Outdoor mobile mapping and AI-based 3D object detection with low-cost RGB-D cameras: The use case of on-street parking statistics. *Remote Sens.* **2021**, *13*, 3099. [CrossRef]
17. Wang, G.; Wu, J.; Xu, T.; Tian, B. 3D vehicle detection with RSU LiDAR for autonomous mine. *IEEE Trans. Veh. Technol.* **2021**, *70*, 344–355. [CrossRef]
18. Prathiba, S.B.; Raja, G.; Bashir, A.K.; AlZubi, A.A.; Gupta, B. SDN-Assisted Safety Message Dissemination Framework for Vehicular Critical Energy Infrastructure. *IEEE Trans. Ind. Inform.* **2022**, *18*, 3510–3518. [CrossRef]
19. Zhang, X.; Li, Z.; Gong, Y.; Jin, D.; Li, J.; Wang, L.; Zhu, Y.; Liu, H. OpenMPD: An Open Multimodal Perception Dataset for Autonomous Driving. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2437–2447. [CrossRef]
20. Sengan, S.; Kotecha, K.; Vairavasundaram, I.; Velayutham, P.; Varadarajan, V.; Ravi, L.; Vairavasundaram, S. Real-Time Automatic Investigation of Indian Roadway Animals by 3D Reconstruction Detection Using Deep Learning for R-3D-YOLOV3 Image Classification and Filtering. *Electronics* **2021**, *10*, 3079. [CrossRef]
21. Rangesh, A.; Trivedi, M.M. No blind spots: Full-surround multi-object tracking for autonomous vehicles using cameras and lidars. *IEEE Trans. Intell. Veh.* **2019**, *4*, 588–599. [CrossRef]
22. Prathiba, S.B.; Raja, G.; Anbalagan, S.; Gurumoorthy, S.; Kumar, N.; Guizani, M. Cybertwin-Driven Federated Learning Based Personalized Service Provision for 6G-V2X. *IEEE Trans. Veh. Technol.* **2022**, *71*, 4632–4641. [CrossRef]
23. Li, Z.; Du, Y.; Zhu, M.; Zhou, S.; Zhang, L. A survey of 3D object detection algorithms for intelligent vehicles development. *Artif. Life Robot.* **2021**, *27*, 115–122. [CrossRef] [PubMed]
24. Choi, J.D.; Kim, M.Y. A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection. *ICT Express* 2022, *in press*. [CrossRef]
25. Li, G.; Lin, S.; Li, S.; Qu, X. Learning Automated Driving in Complex Intersection Scenarios Based on Camera Sensors: A Deep Reinforcement Learning Approach. *IEEE Sens. J.* **2022**, *22*, 4687–4696. [CrossRef]
26. Hartley, R.; Kamgar-Parsi, B.; Narber, C. Using Roads for Autonomous Air Vehicle Guidance. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3840–3849. [CrossRef]
27. Hata, A.Y.; Osorio, F.S.; Wolf, D.F. Robust curb detection and vehicle localization in urban environments. In Proceedings of the 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, USA, 8–11 June 2014; pp. 1257–1262.
28. Xiao, B.; Guo, J.; He, Z. Real-Time Object Detection Algorithm of Autonomous Vehicles Based on Improved YOLOv5s. In Proceedings of the 2021 5th CAA International Conference on Vehicular Control and Intelligence (CVCI), Tianjin, China, 29–31 October 2021; pp. 1–6.
29. Tian, D.; Lin, C.; Zhou, J.; Duan, X.; Cao, Y.; Zhao, D.; Cao, D. Sa-yolov3: An efficient and accurate object detector using self-attention mechanism for autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 4099–4110. [CrossRef]
30. Duan, X.; Jiang, H.; Tian, D.; Zou, T.; Zhou, J.; Cao, Y. V2I based environment perception for autonomous vehicles at intersections. *China Commun.* **2021**, *18*, 1–12. [CrossRef]

31. Hassaballah, M.; Kenk, M.A.; Muhammad, K.; Minaee, S. Vehicle detection and tracking in adverse weather using a deep learning framework. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 4230–4242. [CrossRef]
32. Xia, Y.; Qu, Z.; Sun, Z.; Li, Z. A human-like model to understand surrounding vehicles' lane changing intentions for autonomous driving. *IEEE Trans. Veh. Technol.* **2021**, *70*, 4178–4189. [CrossRef]
33. Barnett, J.; Gizinski, N.; Mondragón-Parra, E.; Siegel, J.; Morris, D.; Gates, T.; Kassens-Noor, E.; Savolainen, P. Automated vehicles sharing the road: Surveying detection and localization of pedalcyclists. *IEEE Trans. Intell. Veh.* **2020**, *6*, 649–664. [CrossRef]
34. Waqas, M.; Ioannou, P. Automatic Vehicle Following Under Safety, Comfort, and Road Geometry Constraints. *IEEE Trans. Intell. Veh.* **2022**. [CrossRef]
35. Ye, T.; Zhang, J.; Li, Y.; Zhang, X.; Zhao, Z.; Li, Z. CT-Net: An Efficient Network for Low-Altitude Object Detection Based on Convolution and Transformer. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–12. [CrossRef]
36. Cai, X.; Giallorenzo, M.; Sarabandi, K. Machine learning-based target classification for MMW radar in autonomous driving. *IEEE Trans. Intell. Veh.* **2021**, *6*, 678–689. [CrossRef]
37. Levering, A.; Tomko, M.; Tuia, D.; Khoshelham, K. Detecting unsigned physical road incidents from driver-view images. *IEEE Trans. Intell. Veh.* **2020**, *6*, 24–33. [CrossRef]
38. Li, T.; Ma, Y.; Shen, H.; Endoh, T. FPGA implementation of real-time pedestrian detection using normalization-based validation of adaptive features clustering. *IEEE Trans. Veh. Technol.* **2020**, *69*, 9330–9341. [CrossRef]
39. Haq, M.; Ruan, S.J.; Shao, M.E.; ulHaq, Q.; Liang, P.J.; Gao, D.Q. One Stage Monocular 3D Object Detection Utilizing Discrete Depth and Orientation Representation. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21630–21640. [CrossRef]
40. Liang, X.; Yu, X.; Chen, C.; Jin, Y.; Huang, J. Automatic Classification of Pavement Distress Using 3D Ground-Penetrating Radar and Deep Convolutional Neural Network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22269–22277. [CrossRef]
41. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
42. Zhang, Y.; Zhang, Z.; Fu, K.; Luo, X. Adaptive Defect Detection for 3-D Printed Lattice Structures Based on Improved Faster R-CNN. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–9. [CrossRef]
43. Kj, J.; Rajasegaran, J.; Khan, S.; Khan, F.S.; N Balasubramanian, V. Incremental Object Detection via Meta-Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 9209–9216. [CrossRef]
44. Wang, R.; Wang, Z.; Xu, Z.; Wang, C.; Li, Q.; Zhang, Y.; Li, H. A Real-Time Object Detector for Autonomous Vehicles Based on YOLOv4. *Comput. Intell. Neurosci.* **2021**, *2021*, 9218137. [CrossRef] [PubMed]