A Real-Time Driver Assistance System Using Object Detection and Tracking

Jamuna S. Murthy(✉), Sanjeeva S. Chitlapalli, U. N. Anirudha, and Varsha Subramanya

Department of ISE, B N M Institute of Technology, Bangalore, India
jamunamurthy.s@gmail.com

Abstract. ADAS (Advanced Driver Assistance System) has become a vital part of the driving experience. In recent years, there have been several advancements in ADAS technology such as parking assistance and lane detection. The proposed work presents a real-time Driver assistance framework by implementing the state-of-the-art object detection algorithm YOLOv4. This paper provides a comparison between and other state-of-the-art object detectors. Comparison is done based on mean average precision (mAP) and frames per second (FPS) on three different datasets and one standard dataset. YOLOv4 proves to be faster and more accurate than the other object detection algorithms in the comparison. This framework is used to build an application which helps users make better decisions on the road. This application consists of a simple user interface that displays alerts and warnings.

Keywords: Object detection · ADAS · LIDAR sensor · CNN · YOLOv4

1 Introduction

According to WHO, approximately 1.3 million people die each year due to road traffic crashes. With a rise in accidents and with the increase in the number of vehicles, ADAS (Advanced Driver Assistance System) has become a vital part of the driving experience. Prior warnings seconds before an incident can help the driver handle the situation in a better manner. ADAS has emerged as an extremely vital tool with respect to safety in the automobile industry. Notable automotive giants have stepped in to integrate ADAS into their models. Existing ADAS technologies operate on sensors such as LIDAR for the object detection module.

Realizing the importance of timing information, the proposed frame was introduced. Our proposed framework is applied to build an interactive application. The proposed framework assists the user by notifying them with unique alerts and warnings. The proposed framework aims towards providing Alerts and warnings a few seconds prior based on the Real-time data. In order to build such a system, YOLOv4 was implemented by analyzing multiple Object Detection Systems based on their speed and accuracy. Our proposed framework includes a system that will be able to assist drivers in compromising situations by giving a heads up with significant speed and accuracy.

M. Singh et al. (Eds.): ICACDS 2022, CCIS 1614, pp. 150–159, 2022.
https://doi.org/10.1007/978-3-031-12641-3_13

2 Literature Review

Numerous researches are done on different aspects of ADAS and Autonomous vehicles. The Simultaneous Localization And Mapping (SLAM) techniques along with the Detection And Tracking Of Moving Objects (DATMO) techniques were used in autonomous vehicles to solve the problem of object detection by Wang C C et al. (2003), by using a laser scanner [1]. The different observed shapes on each laser scan made it difficult to identify the object. Wang C C et al. (2005) provided a night-vision-based driver assistance system by using their proposed system that performs vehicle recognition as well as lane detection [2]. ADAS also includes Driver Monitoring Systems. Driver Monitoring System (DMS) helps in keeping track of various facial features of the driver like eyelid and mouth movement. One such system was proposed by Shaily S et al. (2021) [3].

Lane detection, being a basic problem in ADAS, has many challenges such as for instance-level discrimination and detection of lane lines with complex topologies. Liu, L et al. (2021) proposed CondLaneNet. The CondLaneNet framework first determines the lane instances and thereafter generates the line shape for every instance dynamically is predicted. A conditional lane detection strategy was introduced by them based on row-wise formulation and conditional convolution to solve instance-level discrimination and in order to tackle detection of lane lines having complex topologies, including fork lines and dense lines, they designed the Recurrent Instance Module (RIM) [4].

In the field of autonomous vehicles, Manoharan S (2019) proposed a better safety algorithm for artificial intelligence [5]. There is a lot of research done in Object Detection since it plays a crucial role in many of the technologies. To get a better understanding of state-of-the-art object detection techniques and models, cloud-based. Liu L et al. (2016) conducted a survey of most of the research that provides a clear picture of these techniques. The main goal of this survey was to recognize the impact of deep learning techniques in the field of object detection that has led to many groundbreaking achievements. This survey covers many features of object detection ranging from detection frameworks to evaluation metrics [6, 7].

For many region-based detectors, like Fast R-CNN [8], a costly per-region subnetwork is applied several times. In order to address this, Dai J et al. (2016) introduced R-FCN by proposing location-sensitive score maps to address a dilemma between translation-invariance in image classification and translation-variance in object detection [9]. One of the major challenges of object detection was to detect and localize multiple objects across a large spectrum of scales and locations, due to which the pyramidal feature representations were introduced. In this, an image is represented with multiscale feature layers. Feature Pyramid Network (FPN), one such model to generate pyramidal feature representations for object detection, presents no difficulty and as well as effective but may not be the optimal architecture design. For image classification in a vast search space, the Neural Architecture Search (NAS) algorithm demonstrates favorable results on the productive discovery of outstanding architectures. Hence, inspired by the modularized architecture proposed by Zoph et al. (2018), Ghiasi G et al. (2019) proposed the search space of scalable architecture that generates pyramidal representations. They proposed an architecture, called NAS-FPN, that provides a lot of flexibility in building object detection architecture and is adaptable to a variety of backbone models, on a wide range of accuracy and speed tradeoffs [10].

Various detection systems repurpose classifiers by taking a classifier for an object and then evaluating it again at multiple locations scales and locations in a test image. For example, R-CNN uses region proposal methods to first produce bounding boxes that are likely to appear in an image and then, on these suggested boxes, run a classifier. These intricate pipelines were slow and hard to optimize. Hence Redmon, J. et al. (2016) proposed You Only Look Once (YOLO), an algorithm that is a single convolutional network simultaneously that predicts multiple bounding boxes and class probabilities for those boxes. Unlike R-CNN and other similar algorithms, YOLO is found to be extremely fast, sees the entire image during training and testing hence making fewer background errors. When trained on natural images and tested on the artwork, YOLO outperforms other algorithms by a wide margin. However, YOLO was shown to fall short of state-of-the-art detection systems in terms of accuracy, and it struggled to precisely localize some objects [11]. Redmon, J. et al. (2017), by focusing mainly on improving recall and localization while maintaining classification accuracy, proposed YOLOv2. It was then found that detection methods are constrained to a small set of objects, hence they as well proposed a joint training algorithm that allows one to train object detectors on both detection and classification data, using which they trained the YOLO9000 algorithm which was built by modifying YOLOv2 [12].

The majority of the accurate CNN-based object detectors required high GPU power and training in order to achieve their optimal accuracy. High GPU power is essential for achieving accuracy and speed in real-time since it is vital in a car collision or obstacle warning model. Bochkovskiy A et al. (2020) proposed a modified version of the state-of-the-art object detection models, YOLOv4, with significant improvement in the speed and accuracy of the models. An impressive aspect of this model is that it can operate in real-time on a conventional GPU and training as well requires only a single GPU. Hence using conventional GPUs such as 1080Ti or 2080 Ti one can train an accurate and extremely fast object detector [13]. Since YOLOv4 outperforms other frameworks, our proposed framework is based on it.

Contributions:

- The proposed framework helps in providing an additional safety layer for the Driver (User) and the passengers of the vehicles.
- The Extraction and Detection modules present in the proposed framework enable the software to outperform its competitors.
- Due to the addition of the visualization module the user can easily interact with the framework. This is one of the key features of the framework.

Based on the previous work done in the field of ADAS and object detection as reviewed the proposed framework was formulated.

3 Proposed Work

The input video is processed as frames, each of which acts as input to the object recognition and detection algorithm (YOLOv4). Each frame is processed along three stages in the algorithm namely-Backbone, neck, and head as presented in Fig. 1.

- Backbone: CSPDarknet53,
- Neck: Concatenated Path Aggregation Networks with Spatial Pyramid Pooling (SPP) additional module
- Head: YOLOv3

Concerning the framework, object detection can be categorized into 3 major modules. These 3 modules are

1. Extraction (Backbone and Neck)
2. Detection (Head)
3. Visualization

3.1 Extraction

The backbone and neck take images (each of the frames) as input to extract the feature maps using CSPDarknet53 and SPP, PANet path-aggregation. Darknet53 comprises 53 Convolutional layers. For detection tasks, 53 layers stacked on to the original architecture of 53 layers give us 106 layers of architecture.

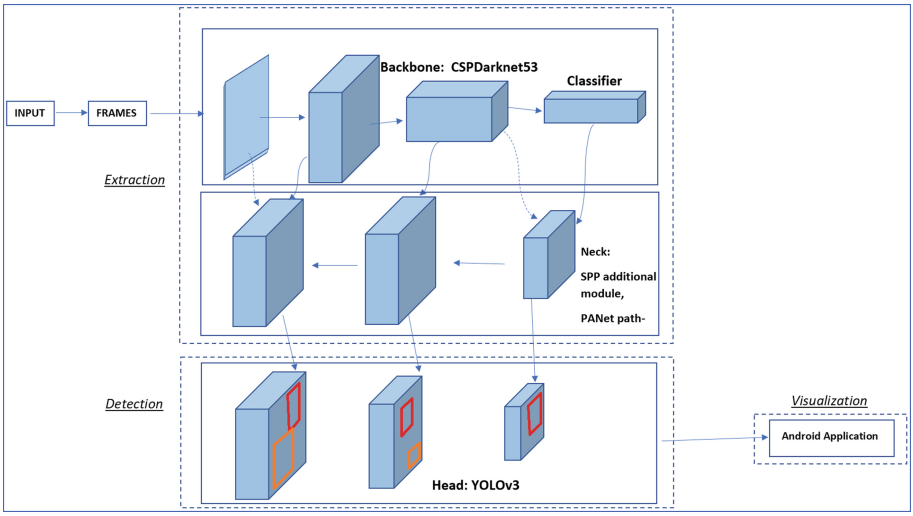


Fig. 1. Proposed object detection framework for ADAS using deep learning

Step 1: Input: the video input is processed frame by frame.

Step 2: CSPDarknet53 - Cross-Stage-Partial-connections are Concerning used to eliminate duplicate gradient information that occurs while using conventional DenseNet [14].

- In CSPDenseNet the base layer is divided into 2 parts; here part A and part B.
- One part will go into the original Dense Block and is processed accordingly; here part B is processed in the Dense block.
- The other part will directly skip to the transition stage.

As a result of this, there is no duplicate gradient information; it also reduces a lot of computations. As shown in Fig. 2.

Step 3: Additional Layers are added between the backbone and the head using the neck. To aggregate the information, the YOLOv4 algorithm applies a modified Path aggregation network [15] with a modified spatial attention module and a modified SPP (Spatial Pyramid Pooling) [16]. Concatenated Path Aggregation Networks [15] with Spatial Pyramid Pooling (SPP) additional modules [16] is used to increase the accuracy of the detector.

3.2 Detection

Each frame processed in the backbone and neck is then transferred to the head which involves the YOLOv3 algorithm which works using the following techniques:

Step 1: Residual **blocks** - initially, the input frame is divided into grids. Each grid cell is responsible for detecting the objects present in its cell.

Step 2: **Bounding box regression** - The YOLO algorithm runs such that bounding boxes and confidence scores are predicted around every object present in that particular grid.

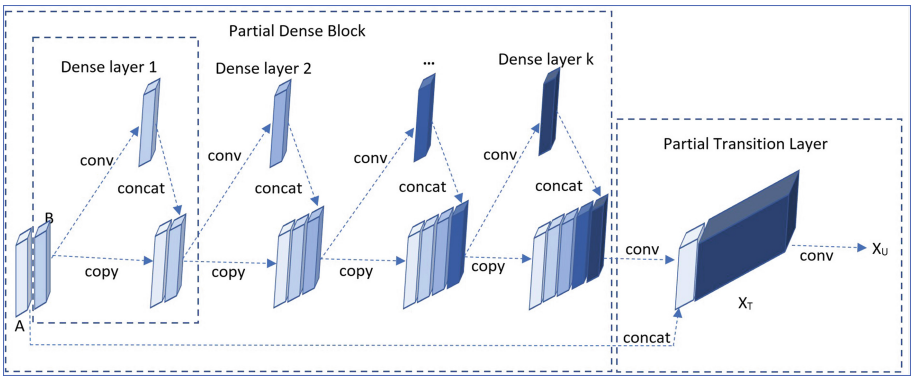
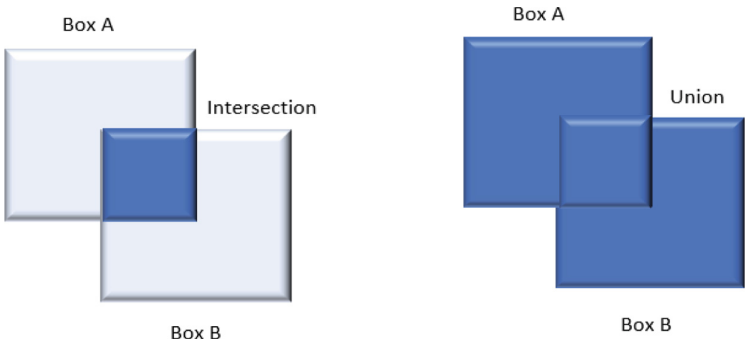


Fig. 2. CSPDenseNet

Every bounding box consists of these attributes: width (bw), height (bh), bounding box center (x, y) and confidence score (c). The confidence score represents how confident, accurate the algorithm is of a particular object in that bounding box. Together with these attributes, YOLO uses a single bounding box regression to predict the probability of an object appearing in the bounding box. Figure 1 shows the YOLOv4 algorithm being run in real-time on a webcam. The algorithm detected objects in the frames by indicating the classes they belong to and the confidence scores representing how sure it is of the objects.

Step 3: Intersection over Union (IoU) - If there is no object in a grid cell, the confidence score is 0; otherwise, the confidence score should be equal to the intersection over union (IoU) between the ground truth and predicted box. Here, the ground truth boxes are manually pre-defined by the user, hence greater IoU means greater confidence score, which means higher accuracy of prediction by the algorithm. This expressed in the equation below (Fig. 3). Filtration of those boxes with no objects is done based on the probability of objects in that box. Non-max suppression processes eliminate the unwanted bounding boxes and the box with the highest probability or confidence score will remain [17, 18].



$$IoU = \frac{Area\ of\ (Box\ A \cap Box\ B)}{Area\ of\ (Box\ A \cup Box\ B)}$$

Fig. 3. Equation for intersection-over-union (IoU)

IoU calculation is used to measure the overlap between two proposals.

Step 4: Final detection - The algorithm detects the object and class probabilities.

3.3 Visualization

The final module of the proposed system involves an android based application. The application inputs a real-time video stream from the device camera runs an object detection algorithm on it and notifies the user under any case of any condition that requires to be brought to the user's attention and needs to be acknowledged.

4 Evaluation Results

With the aim of creating a CNN for real-time operation on a conventional GPU, YOLOv4 was introduced. In the process of doing so, various training improvement methods on the accuracy of the classifier on the ImageNet dataset were tested and their influence was noted along with the accuracy of the detector on the MS COCO dataset.

While comparing YOLOv4 with other state-of-the-art object detectors, it was found that YOLOv4 improved YOLOv3's AP by 10% and FPS by 12% and within comparable performance, YOLOv4 ran twice as fast as EfficientDet. It was found that the classifiers accuracy was enhanced by proposing features such as CutMix and Mosaic data augmentation, Class label smoothing, and Mish activation. YOLOv4 was chosen in our proposed framework since it has a higher accuracy and speed compared to other state-of-the-art object detectors that have real-time operations on a conventional GPU.

In order to evaluate the proposed framework, 3 different types of datasets were used. The 3 datasets were created by using Google Open Images Dataset and custom dataset obtained by labeling the images were gathered. 8% of the collected data consisted of blurry images and images with low visibility. The 3 different datasets were categorized as explained in Table 1 below.

Table 1. Datasets

Datasets	Category	Number of images
Dataset 1	Rural roads	50
Dataset 2	Urban roads	75
Dataset 3	Highways	50

The mAP of a few state-of-the-art object Detectors such as YOLOv3 [19, 20], Faster-RCNN [21, 22], EfficientDet was compared using these datasets. The results of this comparison are represented in Fig. 4 (Fig. 4). With respect to mAP, it is clearly seen that YOLOv4 outperforms its competitors by a significant margin.

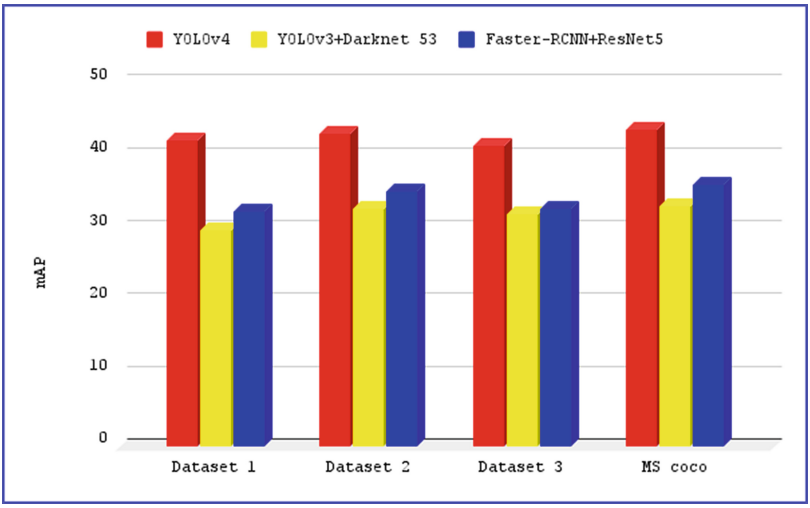


Fig. 4. Comparison of state-of-the-art object detectors on the custom dataset and standard MS coco dataset

The graph in Fig. 5 (Fig. 5) represents a comparative analysis of YOLOv4 with other state-of-the-art object detection algorithms regarding average precision (Y-axis) and frames per second (X-axis). It can be inferred that indeed YOLOv4 algorithm outperforms others in real-time detection. It achieves an average precision between 38 and 43, and frames per second between 65 and 124. The YOLOv3 algorithm, on the other hand, obtains an average precision (AP) of 31 to 33 and frames per second (FPS) of 73 to 120.

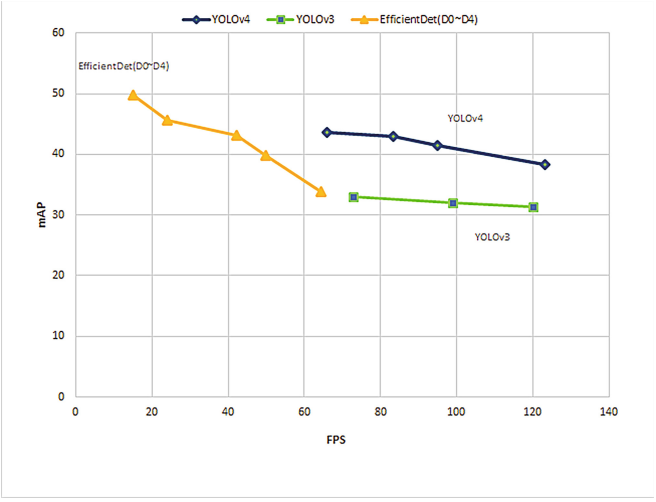


Fig. 5. Comparison of YOLO and EfficientDet(D0~-D4) on Standard MS-coco dataset

5 Conclusion

The proposed framework is intended to provide real-time Object Detection with optimal speed and accuracy to assist the driver. This framework is achieved by implementing the state-of-the-art YOLOv4 algorithm. The whole framework is implemented in the form of three major modules namely Extraction, Detection, and Visualization.

Extraction is the first module, and is used to get the feature map of the provided Input. The Detection module identifies and localizes the object present in the Input. The last module is used to provide an interface that comprises alerts and warnings.

The proposed framework is applied to build the android application which assists the user by notifying them of significant events that require the user to analyze and decide based on it. The proposed application, proposed framework relies majorly on a camera. With the help of some sophisticated cameras, this system can operate under challenging weather conditions.

Hence, in the future, this system can be integrated with other sensors, such as LIDAR [23], to enhance speed and accuracy. The visualization can be improved by integrating the proposed framework with other driver assistance technologies, such as Google Maps and voice assistant. In the future, with the help of a cloud-based approach [24], the processes can be recorded and analyzed. The cloud-based approach also helps in increasing the accessibility of the application. Raspberry pi can also be used in order to have a smooth flow in the processes and increased efficiency. In the future, the proposed framework can be integrated with the Electronic Control Unit (ECU) [25] present inside the vehicles.

References

1. Wang, C.C., Thorpe, C., Thrun, S., Hebert, M., DurrantWhyte, H.: Simultaneous *localization, mapping and moving object tracking. Int. J. Robot. Res. **26**(9), 889–916 (2007)
2. Wang, C.C., Huang, S.S., Fu, L.C.: Driver assistance system for lane detection and vehicle recognition with night vision. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3530–3535. IEEE, August 2005
3. Shaily, S., Krishnan, S., Natarajan, S., Periyasamy, S.: Smart driver monitoring system. Multimedia Tools Appl. **80**(17), 25633–25648 (2021). https://doi.org/10.1007/s11042-021-10877-1
4. Liu, L., Chen, X., Zhu, S., Tan, P.: CondLaneNet: a top-to-down lane detection framework based on conditional convolution. arXiv preprint arXiv:2105.05003 (2021)
5. Manoharan, S.: An improved safety algorithm for artificial intelligence enabled processors in self driving cars. J. Artif. Intell. **1**(02), 95–104 (2019)
6. Liu, L., et al.: Deep learning for generic object detection: a survey. Int. J. Comput. Vision **128**(2), 261–318 (2020)
7. Zou, Z., Shi, Z., Guo, Y., Ye, J.: Object detection in 20 years: a survey. arXiv preprint arXiv:1905.05055 (2019)
8. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
9. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems, pp. 379–387 (2016)
10. Ghiasi, G., Lin, T.Y., Le, Q.V.: NAS-FPN: learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7036–7045 (2019)

11. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
12. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
13. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: YOLOv4: optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
14. Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W., Yeh, I.H.: CSPNet: a new backbone that can enhance learning capability of CNN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 390–391 (2020)
15. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8759–8768 (2018)
16. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
17. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: a metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 658–666 (2019)
18. Neubeck, A., Van Gool, L.: Efficient non-maximum suppression. In: 18th International Conference on Pattern Recognition (ICPR 2006), vol. 3, pp. 850–855. IEEE, August 2006
19. Redmon, J., Farhadi, A.: YOLOv3: anincremental improvement. University of Washington (2018)
20. Lee, Y.H., Kim, Y.: Comparison of CNN and YOLO for object detection. J. Semicond. Disp. Technol. **19**(1), 85–92 (2020)
21. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. Adv. Neural. Inf. Process. Syst. **28**, 91–99 (2015)
22. Shine, L., Edison, A., Jiji, C.V.: A comparative study of faster R-CNN models for anomaly detection in 2019 AI city challenge. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 306–314 (2019)
23. Beltrán, J., Guindel, C., Moreno, F.M., Cruzado, D., Garcia, F., De La Escalera, A.: BirdNet: a 3D object detection framework from LiDAR information. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 3517–3523. IEEE, November 2018
24. Cabanes, Q., Senouci, B.: Objects detection and recognition in smart vehicle applications: point cloud based approach. In: 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN), pp. 287–289. IEEE, July 2017
25. Talavera, E., Díaz-Álvarez, A., Naranjo, J.E., Olaverri-Monreal, C.: Autonomous vehicles technological trends. Electronics **10**(10), 1207 (2021)