Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 3, No. 5, 2024

A Study on Obstacle Detection in Unmanned Driving Using an Improved Faster R-CNN Model

Emily Thompson[1], Eric Chen[2]

[1]Northern Arizona University, Flagstaff,USA

[2]Northern Arizona University, Flagstaff,USA

Correspondence should be addressed to Emily Thompson; emily23.thompson@nau.edu

Abstract: Intelligent vehicles, driven by advancements in computer technology, sensors, and artificial intelligence, are poised to revolutionize the transportation industry. These vehicles require robust systems for environmental perception and collision avoidance to ensure safety and efficiency. This study proposes an improved Faster-RCNN model, incorporating ResNet50 as the feature extraction network, aimed at enhancing obstacle detection accuracy in autonomous driving scenarios. Evaluated on the VOC2007 dataset, the model demonstrates a 12.15% improvement in average detection accuracy over traditional methods. The results indicate the model's superior performance in detecting various objects such as bicycles, buses, and pedestrians, underscoring its potential for broad application in intelligent vehicle systems.

Keywords: Unmanned driving; Obstacle detection; Faster-RCNN model.

1.Introduction

Intelligent vehicles represent a sophisticated system that integrates several critical functions, including environmental perception, path planning, multi-level vehicle management, and more. These functions are enabled by advancements in computer technology, cutting-edge sensors, information fusion, wireless communication, artificial intelligence, and automation, forming a highly advanced technological infrastructure. Current research in intelligent vehicles is predominantly focused on enhancing automotive safety and ride comfort, with an emphasis on refining the human-vehicle interface. In recent years, autonomous intelligent vehicles have emerged as a global research hotspot in the automotive manufacturing industry, driving industrial innovation and growth. As a result, many developed nations have prioritized the development of these technologies.

With the rapid advancement of modern high technologies, including digitization, informatization, and intelligence, every aspect of human society's production and daily life has been profoundly transformed. The day is fast approaching when we will see intelligent autonomous vehicles operating on roads, transitioning this cutting-edge technology from mere concept to reality. Various high-tech vehicles already demonstrated substantial progress in terms of performance, comfort, and safety. In intelligent autonomous vehicles, sensor devices are intricately linked to the surrounding environment, and tasked with collecting and organizing vast amounts of data. This data is then processed by highly intelligent computers, enabling swift control and operation of the vehicle systems. Consequently, the full potential of functions such as autonomous driving and intelligent control can be realized.

As socio-economic development accelerates, the transportation industry is flourishing, leading to a surge in the number of vehicles on the road. This rapid increase has exacerbated traffic congestion and led to frequent accidents, resulting in significant casualties and economic losses. To address these challenges, it is imperative to design a responsive, highly reliable, and cost-effective collision avoidance and warning system for vehicles. Ultrasonic collision avoidance is one of the most prevalent methods for distance measurement, particularly effective in short-range, low-speed collision prevention scenarios such as parking. It is also widely applied in vehicle reverse collision warning systems. Ultrasonic waves, as a unique form of sound waves, exhibit fundamental physical properties such as refraction, reflection, interference, diffraction, and scattering. Ultrasonic collision avoidance systems leverage these reflective properties to detect obstacles behind a vehicle during reverse maneuvers. The ultrasonic distance sensors alert the driver to the proximity and position of obstacles using indicator lights and a buzzer, thereby enhancing safety.

2. Optimization of Obstacle Detection Methods

In 2020, Chintakindi Balaram Murthy introduced an enhanced YOLOv3+ network designed to accurately detect small pedestrians in complex environments in real-time. This network incorporates K-means clustering before training to select the optimal K bounding boxes, ensuring more precise object localization. Additionally, the improved YOLOv3+ network integrates a reverse residual module, which significantly boosts feature extraction capabilities. The loss function is also refined to minimize errors in bounding box predictions. As a result, the network demonstrates superior robustness, achieving an Average Precision (AP) of 79.86%, outperforming existing networks in detection accuracy. However, a minor reduction in detection speed is observed, particularly when identifying smaller pedestrians.

Complementing this, the paper also presents a road obstacle detection method based on an improved Faster-RCNN model, which is trained and evaluated using the VOC2007 dataset. When compared to the EfficientNet network, the proposed method offers enhanced accuracy in detecting road obstacles, making it a promising solution for autonomous driving applications.

2.1.Overall Scheme for Obstacle Detection

Recent advancements in deep learning and neural networks have significantly contributed to the field of obstacle detection in autonomous driving. The development of sophisticated models, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), has enabled substantial progress in this domain.Jiang et al. [1] introduced a novel perspective on Recurrent Neural Networks (RNNs) by proposing the Carry-lookahead RNN model. This model offers a more efficient way to process sequential data, which is crucial for real-time obstacle detection in unmanned vehicles. The model's ability to handle sequential dependencies effectively makes it a valuable contribution to enhancing the temporal understanding necessary for autonomous driving scenarios.

Cao et al. [2] explored the Adaptive Receptive Field U-shaped Temporal Convolutional Network (TCN), which focuses on segmenting actions over time. The TCN's architecture, which adjusts its receptive field dynamically, is particularly relevant for processing temporal sequences in obstacle detection systems, where understanding the timing and sequence of events is critical for accurate prediction and reaction.Chen et al. [3] applied a Global-Local Attention Transformer to the classification of imbalanced data, specifically in leukocyte classification. Although this work is primarily in the medical domain, the methodology behind handling imbalanced datasets and employing attention mechanisms can be adapted to enhance the precision of obstacle detection in autonomous vehicles, where certain types of obstacles may be underrepresented in the training data.

Tao's work [4], [5] on blackbox attacks and adversarial defense through sequential query-based techniques and meta-learning respectively, provides crucial insights into the robustness of neural networks. These techniques can be adapted to improve the security and reliability of obstacle detection models in unmanned driving by defending against potential adversarial attacks that could compromise the safety of the vehicle.Xiao et al. [6] and Yan et al. [7] both contributed to the application of CNNs in medical image classification and survival prediction across cancer types. The CNN models developed in these studies, although applied to different domains, offer robust methodologies for image classification that are directly applicable to the visual tasks involved in obstacle detection in autonomous vehicles. Finally, Yao et al. [8] introduced NDC-Scene, a method for monocular 3D semantic scene completion. This work is highly relevant to obstacle detection in unmanned driving, as it deals with understanding and completing scenes in 3D space using monocular vision, a critical component of autonomous vehicle perception systems.

These studies collectively demonstrate the significant progress made in neural network architectures and their applications across various domains. By leveraging these advancements, the proposed improved Faster R-CNN model for obstacle detection in autonomous driving builds upon a solid foundation of deep learning research, aiming to enhance precision and robustness in real-world scenarios.

Initially, this paper involves resizing an image from its original dimensions P×Q to M×N. The resized M×N image is then processed through the Backbone network for feature extraction, resulting in a feature map. The selected Backbone network is based on ResNet50, which comprises two primary types of blocks: Conv Blocks and Identity Blocks. Conv Blocks, used to adjust the network's dimensions, handle different input and output sizes, thereby preventing uninterrupted concatenation. On the other hand, Identity Blocks, with matching input and output dimensions, allow for concatenation and are utilized to deepen the network's structure[9][10].

After the feature map is generated, it is passed to the Region Proposal Networks (RPN) layer. Here, the softmax activation function is employed to classify the generated anchors, while bounding box regression refines these anchors to produce accurate proposals. The Region of Interest (ROI) Pooling layer then operates with two inputs: the proposal layer and the feature map layer. This layer extracts relevant information, which is subsequently fed into a fully connected layer to classify the target category, as depicted in Figure 1.

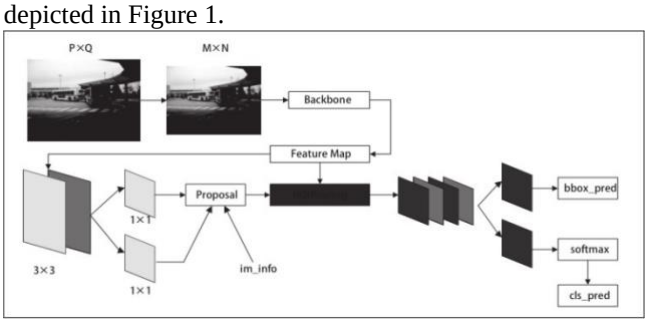

Figure 1. Overall Block Diagram

2.2.Model Structure

2.2.1.Backbone

In this paper, the Backbone section utilizes the ResNet50 network, where "50" refers to the total number of layers distributed across five stages: Stage 0, Stage 1, Stage 2, Stage 3, and Stage 4. Stage 0 begins with an input image characterized by channel (C), height (H), and width (W) dimensions, specifically (3, 224, 224). The data first undergoes operations in the initial layer, including convolution (CONV), batch normalization (BN), and activation through the ReLU function. This is followed by max-pooling in the second layer. Xu et al. [11] conducted significant research on the application of multimodal deep learning in image recognition systems. Their study highlights the effectiveness of integrating multiple data modalities to enhance the accuracy and reliability of image-based recognition tasks. In the context of autonomous driving, such multimodal approaches are particularly valuable, as they allow for the synthesis of information from

various sources (e.g., cameras, LiDAR, radar) to create a more comprehensive understanding of the driving environment. This comprehensive understanding is crucial for accurately detecting and classifying obstacles in real time, a task that becomes increasingly challenging in dynamic and unpredictable road conditions.

In the first layer of Stage 0, the convolutional kernel has a size of 7×7, with 64 filters, and a stride of 2. The second layer applies max-pooling with a 3×3 kernel and a stride of 2. As a result of these operations, the output shape at the end of Stage 0 is (64, 56, 56), which corresponds to 64 channels, a height of 56, and a width of 56. The reduction in height and width to 56 from the original 224 is due to the halving of the image dimensions twice—once by the convolutional layer and once by the max-pooling layer—each with a stride of 2, effectively reducing the input scale by a factor of four.

Stages 1, 2, 3, and 4 of the ResNet50 network each consist of two key types of bottlenecks: BTNK1, where the input and output channel numbers differ, and BTNK2, where the input and output channel numbers are the same. In BTNK2, the input image has dimensions (C, W, W), with C representing the number of channels and W the width and height of the image. Given an input x with dimensions (C, W, W), the left-side convolutional block of BTNK2, along with its corresponding activation function, can be expressed as $f(x)$. The output of BTNK2 is then computed as f(x) + x), with a single ReLU activation applied, maintaining the original shape of the input, (C, W, W).

In contrast, BTNK1, with its variable parameters C, W, C1, and S4, includes an additional convolutional layer denoted as g(x). Due to the difference in input and output channel numbers in BTNK1, the convolutional layer transforms the input x into g(x), resulting in different input and output dimensions. However, the output channels of g(x) match those of f(x), allowing the final output of BTNK1 to be expressed as f(x) + g(x).

Stage 1 begins with one BTNK1 layer, followed by two sequential BTNK2 layers, producing an output with dimensions (256, 56, 56). This output is then passed to Stage 2, which comprises one BTNK1 layer followed by three sequential BTNK2 layers, yielding a shape of (512, 28, 28). The output from Stage 2 is sent to Stage 3, where one BTNK1 layer is followed by five sequential BTNK2 layers, resulting in a shape of (1024, 14, 14). Finally, in Stage 4, the combination of one BTNK1 layer and two sequential BTNK2 layers produces the output with dimensions (2048, 7, 7). This output represents the final feature map, which is then used in subsequent network layers for further processing, such as classification or detection tasks.

2.2.2.Region Proposal Networks (RPN)

In OpenCV, while sliding windows and image pyramid methods in AdaBoost can improve the generation of discriminative detection boxes, they involve redundant computations that consume significant resources and time. To address this inefficiency, this paper employs Region Proposal Networks (RPN) to generate detection boxes more effectively.

The RPN consists of two main branches. The first branch uses the softmax activation function to classify anchors, distinguishing between correct and incorrect classifications. The second branch handles the bounding box regression offsets generated by the anchors, adjusting these offsets to produce more accurate proposals. The Proposal layer then aggregates the correctly classified anchors, applies the corresponding offsets, and generates proposals. Proposals that are either too small or fall outside the image boundaries are subsequently filtered out.

Anchors, which are a set of rectangles generated by the RPN, are arranged in a 9x4 matrix. The coordinates of each rectangle's four corners are represented as (x1, y1, x2, y2), and the rectangles feature aspect ratios of 1:1, 1:2, and 2:1. To adjust the position of the detection boxes, the Conv layers iterate over the feature maps, matching all points with these nine initial boxes. The final detection box positions are refined through two rounds of bounding box regression, ensuring accurate localization of the targets.

2.2.3.ROI Pooling

The ROI Pooling layer plays a critical role in aggregating and processing the proposal feature maps generated from the proposals, subsequently forwarding them to the next layers in the network. This layer receives two inputs: the original feature maps and the proposal boxes generated by the RPN, which can vary in size. Unlike traditional CNN networks like AlexNet and VGG, which require a fixed input image size after training and produce a fixed-size output vector or matrix, the ROI Pooling layer must handle varying input image sizes while preserving the original shape information. This is essential to avoid cropping or warping the image, which could lead to the loss of important structural details.

This process allows the network to handle inputs of varying sizes while maintaining the integrity of the image's structural information, making it particularly useful in object detection tasks where the size and shape of objects can vary significantly.

2.2.4.Classification

In the Classification section, the proposal feature maps obtained from the ROI Pooling layer are processed to determine the specific category of each proposal. This is achieved by passing the feature maps through fully connected layers, followed by the application of the softmax function. The softmax function calculates the probability distribution across the various possible categories (e.g., person, car, bicycle), resulting in the cls_prob probability vector, which indicates the likelihood of each proposal belonging to a particular class.

Simultaneously, the network employs bounding box regression to refine the position of each detection box. This involves calculating the position offsets, denoted as bbox_pred, for each proposal. These offsets adjust the initial bounding boxes generated by the RPN, enabling the regression process to produce more accurate object detection boxes. Through this dual process of classification and regression, the model enhances both the categorical accuracy and the spatial precision of the detected objects.

2.3.Model Training

In this study, the VOC2007 dataset is employed for both model training and testing. The process begins by organizing the training labels, which are stored in the Annotation folder. This folder contains all the label information for the training set. The images used for training are placed in the JPEGImages folder. The dataset is split into training and validation sets with a ratio of 9:1, ensuring a robust model evaluation.

Given that obstacles encountered during vehicle operation can vary widely in shape, complexity, and unpredictability, this study emphasizes the importance of distinguishing between different detection categories. To address this, 12 common detection categories have been defined: bicycle, boat, bottle, bus, car, cat, chair, dog, horse, motorbike, person, and train. These categories are documented in a text file named cls_classes.txt.

The input image size for the model is set to [600, 600], providing a standardized input shape. Additionally, the anchor sizes, which are critical for generating bounding boxes of varying scales, are set to [8, 16, 32]. These anchor sizes are designed to accommodate the detection of objects of different sizes, thereby enhancing the model's ability to accurately identify and classify a wide range of obstacles.

Model training is conducted in two phases: freezing and unfreezing. In the freezing phase, the backbone network remains fixed to preserve feature extraction, with settings of an initial epoch at 0, a freezing epoch at 50, a batch size of 4, and a learning rate of 1e-4. During the unfreezing phase, the backbone network is unlocked to allow updates in feature extraction, with an unfreezing epoch set at 100, a reduced batch size of 2, and a lower learning rate of 1e-5.

Table 1. Test Results on VOC2007 trainval

	Backbone	Anchor boxes	mAP/%(VOC2007trainval)
Faster-RCNN	VGG16	12	68.2
Efficientdet	Efficientnet	9	69.83
this paper	Resnet50	9	80.35

2.4.Prediction and Results

In this study, AP (Average Precision), mAP (mean Average Precision), and LAMR (log Average Miss Rate) are employed as the key evaluation metrics. The Faster-RCNN and EfficientDet networks, which are designed and trained in this study, are assessed using these metrics. The appropriate weight files are selected to evaluate the models' performance in predicting obstacle detection scenarios in road images, as illustrated in Figures 2 and 3. According to the detection accuracy observed in these figures, the predictions from this study surpass the alternatives.

Figure 2. EfficientDet Prediction Results

Figure 3. Prediction Results in This Study

The study further conducts an AP analysis for 12 common road obstacles, as shown in Figure 4. Additionally, an LAMR analysis for these obstacles is presented in Figure 5. Table 1 summarizes the mAP values for various models on the VOC2007 dataset.

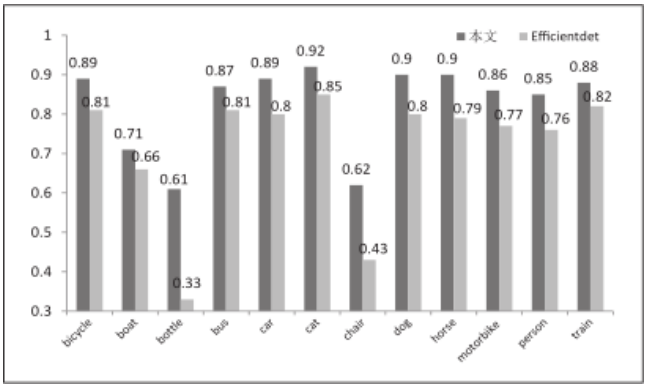

Figure 4. AP for 12 Common Road Obstacle Classes in Faster-RCNN and EfficientDet Networks

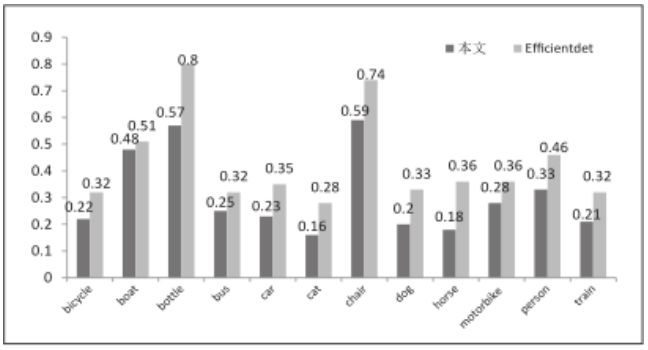

Figure 5. LAMR Values for 12 Common Road Obstacle Classes in Faster-RCNN and EfficientDet Networks

Figure 4 demonstrates that the Faster-RCNN model developed in this study significantly outperforms EfficientDet in terms of detection accuracy, particularly for pedestrians, bottles, buses, cars, motorcycles, and bicycles. Additionally, Figure 5 shows that the log values of the

average miss rate (LAMR) for the Faster-RCNN are consistently lower than those for EfficientDet. Table 1 compares the mAP values across three different networks. The network proposed in this study achieves a 12.15% increase in average precision compared to Faster-RCNN with VGG16 as the feature extraction network and 12 anchor boxes. Similarly, it shows a 10.52% improvement compared to EfficientDet, which uses EfficientNet for feature extraction with 9 anchor boxes.

3. Conclusion

The study underscores the significance of enhancing obstacle detection in unmanned driving by leveraging an improved Faster-RCNN model with ResNet50 as the feature extraction backbone. By increasing the convolutional layers' depth, the model effectively optimizes feature information, leading to a notable 12.15% improvement in detection accuracy over traditional approaches. The experimental validation on the VOC2007 dataset demonstrates the model's superior capability in identifying a range of objects critical to autonomous driving, including bicycles, buses, and pedestrians. These results affirm the model's potential for widespread adoption in intelligent vehicle systems, contributing to safer and more reliable autonomous driving experiences. This research not only advances the technical understanding of obstacle detection but also lays a solid foundation for future innovations in the field of intelligent transportation systems.

References

[1] H. Jiang, F. Qin, J. Cao, Y. Peng, and Y. Shao, "Recurrent neural network from adder's perspective: Carry-lookahead RNN," Neural Networks, vol. 144, pp. 297-306, 2021.

[2] J. Cao, R. Xu, X. Lin, F. Qin, Y. Peng, and Y. Shao, "Adaptive receptive field U-shaped temporal convolutional network for vulgar action segmentation," Neural Computing and Applications, vol. 35, no. 13, pp. 9593-9606, 2023.

[3] B. Chen, F. Qin, Y. Shao, J. Cao, Y. Peng, and R. Ge, "Fine-grained imbalanced leukocyte classification with global-local attention transformer," Journal of King Saud University-Computer and Information Sciences, vol. 35, no. 8, p. 101661, 2023.

[4] Y. Tao, "SQBA: sequential query-based blackbox attack," in Fifth International Conference on Artificial Intelligence and Computer Science (AICS 2023), SPIE, 2023, vol. 12803, pp. 721-729.

[5] Y. Tao, "Meta Learning Enabled Adversarial Defense," in 2023 IEEE International Conference on Sensors, Electronics and Computer Engineering (ICSECE), IEEE, 2023, pp. 1326-1330.

[6] M. Xiao, Y. Li, X. Yan, M. Gao, and W. Wang, "Convolutional neural network classification of cancer cytopathology images: taking breast cancer as an example," in Proceedings of the 2024 7th International Conference on Machine Vision and Applications, 2024, pp. 145-149.

[7] X. Yan, W. Wang, M. Xiao, Y. Li, and M. Gao, "Survival prediction across diverse cancer types using neural networks," in Proceedings of the 2024 7th International Conference on Machine Vision and Applications, 2024, pp. 134-138.

[8] J. Yao et al., "Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space," in 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 2023, pp. 9421-9431.

[9] Block, D. (2021). Innovations and challenges in identity research. Routledge.

[10] Anous, T., & Haehl, F. M. (2020). On the Virasoro six-point identity block and chaos. Journal of High Energy Physics, 2020(8), 1-34.

[11] Xu, T., Li, I., Zhan, Q., Hu, Y., & Yang, H. (2024). Research on intelligent system of multimodal deep learning in image recognition. Journal of Computing and Electronic Information Management, 12(3), 79-83.