International Journal of Multimedia and Ubiquitous Engineering
Vol. 14, No. 1 (2019), pp.7-12
http://dx.doi.org/10.21742/ijmue.2019.14.1.02

Velocity-based object detection in dynamic environment using YOLO-based deep learning algorithm

Ihn-Sik, Weon[1], Soon-Geul, Lee*[2]

[1]Dept of Mechanical Engineering, Graduated School, Kyung Hee University, Yongin-si, Korea
*[*2]Dept of Mechanical Engineering, Kyung Hee University, Yongin-si, Korea*
*[1]opt1064@gmail.com,[*2]sglee@khu.ac.kr*

Abstract

In this study, to solve the constraints of an image sensor and resolve an obstacle detection error according to the traveling speed of an autonomous vehicle, we applied the object recognition technology of the single-shot detector technique based on the you only look once (YOLO) algorithm to pedestrians, bicycles, traffic lights, and pedestrian crossings for effective obstacle avoidance and perception. The proposed technique was experimentally proven on campus at Kyung Hee University, with the results confirming the accuracy of object recognition using a number of learning datasets.

Keywords: *Single-shot detector(SSD), Object detection, Deep-learning, Vision System, FMCW Radar.*

1. Introduction

Growing research on the autonomous driving technology has increased the demand for an advanced object detection technology for various obstacles (e.g., signs, vehicles, and pedestrians).[1] Such detection technology is instrumental in operating autonomous vehicles, and errors may lead to serious accidents.[2]

Therefore, considerable research on various sensors to increase the accuracy of discrimination patterns has been conducted thus far. In particular, obstacle detection with three-dimensional (3D) light detection and ranging (Lidar) identifies obstacles by processing data in the form of point clouds usually by using a Gaussian filter [3] or a K-means clustering algorithm.[4] However, 3D Lidar requires the processing of large amounts of data, and the time requirement for such data processing is incompatible with the time requirement for obstacle detection during driving.[5]

The radar system primarily used for mobile robots can detect obstacles both near and far, and is made of a wide range of materials. However, the system offers unclear directional information about an obstacle because of the wide distribution of the measurement points, and has a considerable amount of noise to reliably detect objects.[6]

Therefore, research is underway to combine 3D Lidar sensors with radar for object detection.[7]

---

[1] **Article history:**
Received (January 25, 2019), Review Result (February 25, 2019), Accepted (March 29, 2019)

Print ISSN: 1975-0080 eISSN: 2652-1954 IJMUE

The use of a processor to manage the large amount of Lidar point data and a radar data processing unit for noise filtering increases the size of the entire system, making it expensive. Consequently, many researchers have turned to image sensors that can detect objects at a low cost. Unlike Lidar and radar, an image sensor can perform multiple functions simultaneously and allows one to define the objects to be detected in detail (e.g., pedestrians, traffic lights, and vehicles).[8] However, image sensors have many weather and environmental constraints and require running many mathematical algorithms (e.g., boundary and curve detections) on the acquired two-dimensional (2D) data. Thus, the real-time use of these sensors is limited by their inability to detect multiple objects.[9]

In this study, to address the constraints of the image sensor and eliminate obstacle detection errors according to the traveling speed of autonomous vehicles, we applied the single-shot detection (SSD) technique for object detection based on the you only look once (YOLO) algorithm to detect pedestrians, vehicles (buses and passenger cars), bicycles, and pedestrian crossings for the effective avoidance and recognition of objects. The accuracy of object detection was assessed using a number of datasets for learning and an experiment conducted on campus at Kyung Hee University.

2. Basic object detection algorithm

2.1. Relevant research

Computer vision research focused on optimizing the speed of processing vision has been conducted using SIFT, HOG, SURF, ORB, and BRISK. Such research has generally proven to be useful in various applications. A wide range of deep-learning methods have been used for object detection, and early studies used the method of designating candidate areas for selective detection and extraction. In addition, separate algorithms such as edge box have been used for search and deep learning for classification.

Region-based CNN (R-CNN) is a sliding window method, a conventional method for object detection algorithms, in which objects are detected by searching the entire area in an image by using a window of a certain size. However, its inefficiency has led to the development of the region proposal algorithm. R-CNN extracts region proposals that are estimated to have objects from an input image by using a selective search method. A region proposal is an image in a rectangular bounding box; region proposals are classified using CNN after the areas of these proposals are processed to become identical. R-CNN

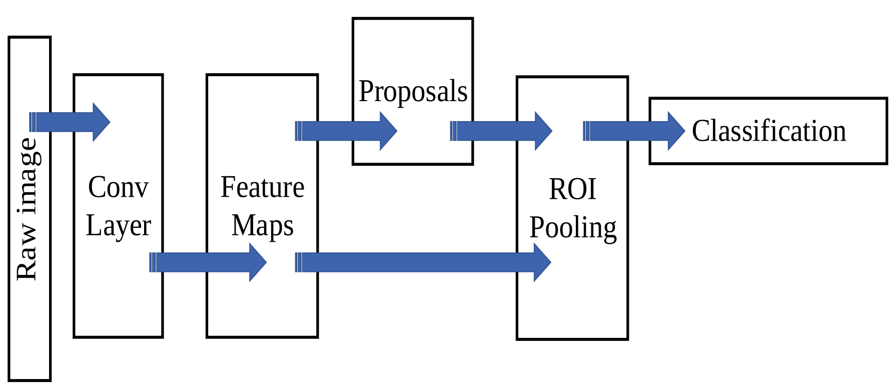

Figure 1. Basic structure of Faster R-CNN.

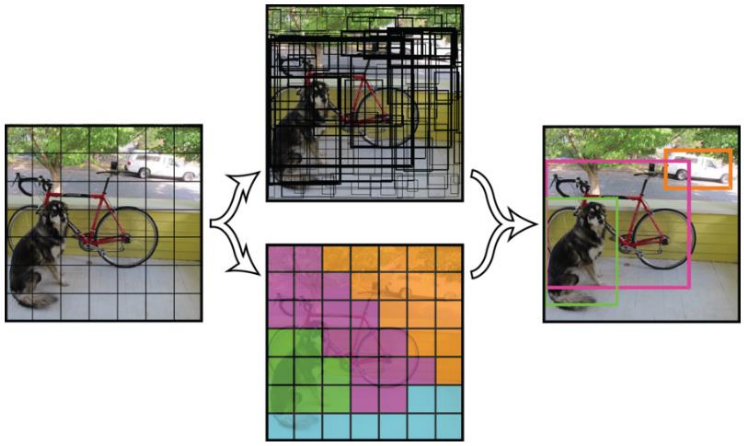

Figure 2. Basic model of YOLO network.

also generates a model that predicts the exact position and size of the bounding box. The problem is that the process delay increases because of the regression model added for this task.

Therefore, faster R-CNN was developed to resolve the delay caused by using a neural network. Faster R-CNN is a model that involves the creation of region proposals as a network in CNN; it is also called a region proposal network (RPN). It can extract specific areas and search locations and size quickly because the layer that performs the ROI pooling through RPN and the layer that extracts the bounding box share the same feature map.

However, R-CNN uses the region proposal method to create bounding boxes that are likely to have objects inside their images. The boxes are repeatedly reclassified to detect and specify objects. However, the complex structure of R-CNN makes it difficult to optimize the algorithm and requires a long time to identify an object.

In this present study, YOLO was used in the object detection algorithm because it enables a single network to extract features, create bounding boxes, and classify them simultaneously through location search and classification in the final output phase.

2.2. Object detection algorithm

In this study, the YOLO network, a deep-learning object detection algorithm, was used for object detection and classification. YOLO performed location search and classification simultaneously in the final output phase of the network. A single network extracted the features, created the bounding boxes, and classified them simultaneously.[10]

As shown in Figure 2, the algorithm generated two datasets from a raw input image. They contained encoded data on the class of each grid cell when a number of bounding boxes and images were divided into an $n \times n$ grid.

Figure 2 shows that the network classified the image into a 7×7 grid. The network generated two bounding boxes in different sizes in each grid cell. As there were 49 grid cells

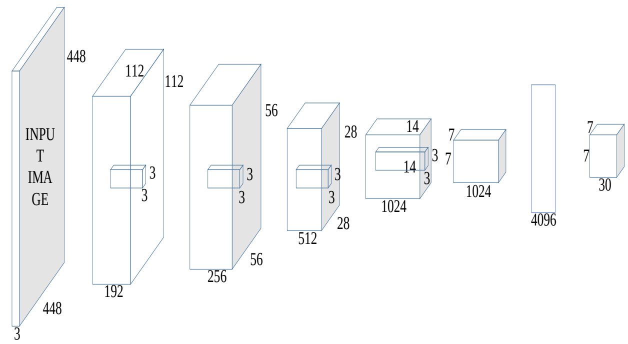

Figure 3. Basic structure of Faster YOLO network.

in the image, 98 bounding boxes were generated. The higher the probability of the object was, the thicker was the border of the bounding box. The non-maximal suppression (NMS) algorithm was used to identify objects with the remaining candidate bounding boxes to generate the final image shown in Figure 2.

As shown in Figure 3, Conversion Layer 4 and Full Connection Layer 2 were performed in the input image to generate the predicted result of $7 \times 7 \times 30$. The resulting bounding boxes were classified, and many classes were filtered by applying a predetermined threshold value for the probabilities. However, if the size of an object was large, the number of classes that were likely to represent an object increased. The classes were filtered using the NMS algorithm to finalize the bounding boxes for the object.

3. Experiment and results

3.1. System configuration

To determine the correlation between the detection rate of the object detection algorithm and the vehicle speed, a USB 3.0 camera was installed on the top exterior of the vehicle. The RGB image data were obtained from the camera with the specifications described in Table 1.

The image data of driving on the actual road were obtained using the system with the configurations described in Table 1. The image data were acquired by driving on the road near Kyung Hee University, and the experiment was conducted at the maximum vehicle speed of 60 km/h. The machine learned about pedestrians, vehicles, traffic lights, and pedestrian crossings from the RGB images. The learning data were the driving images, which were encoded for learning by frame.

The original RGB images were resized into 416×416 for learning, and the weights file was created by boxing the necessary learning data in 16,650 frames. On the basis of the learned data,

Table 1. Camera specifications.

Classification	Description
Resolution	1280 × 960
FPS	30 fps
Interface	USB 3.0

Table 2. Configuration of learning data.

Class	Number of training data items
Vehicle	5,220
Crosswalk	826
Pedestrian	650
Traffic Light	220

we calculated the object detection rates according to the vehicle speed and compared the detection rates according to the amount of learning data.

Vehicle speeds—0, 20—were used along with 16,650 frames and 33,300 frames of learning data.

3.2. Experimental results

As shown in Table 2, the classification and learning of four types of datasets in each frame was conducted using the corresponding amount of data Table 2. We measured the object detection rates according to the learning data while driving back and forth on the road between Deokyoungdae-ro and Seocheon-dong in front of the International Campus of Kyung Hee University.

(a). Object detection at 0 km/h (b). Object detection at 20 km/h

Figure 4. Result of object detection

In the experiment, we set the vehicle speed between 0 and 20 km/h, and the detection rates for the fixed objects such as pedestrian crossings and traffic lights according to the vehicle speeds ranged between 0 and 30 km/h.

4. Conclusion

In this study, we determined the object detection rates on actual roads by using a relatively small amount of learning data. However, considering the current amount of learning data and the fact that the detection rates of the fixed and dynamic objects at 0–30 km/h exceeded 80%, we expect the detection rates to remain high irrespective of the vehicle speed, if the original image data with a relatively high resolution and a large amount of learning data are used.

Acknowledgements

This study was funded by the Korea Institute of Industrial Technology Evaluation and Planning’s Industrial Technology Innovation Program [10067781] and Kyung Hee

University's Convergent Research Program for "Smart (Autonomous Driving) Vehicle Development" [20161737] and National Research Foundation's [NRF-2019R1A2C2010195]

References

[1] G. H. Anh.; S. W. Lee.; Technology Trends of Self-Driving Vehicle. Report of ETRI, **(2013)**, 135(4), 35-54.

[2] J. M. Kim.; J. M Heo.; S. Y. Jung.; S. S. Kim.; Path-Planning Using Modified Genetic Algorithm and SLAM Based on Feature Map for Autonomous Vehicle. Journal of Korean Institute of Intelligent Systems, **(2009)**, 19(3), 381-387.

[3] D. J. Yoon.; J. H. Kim.; J. H. Kim.; LiDAR Point Cloud Data Clustering and Classification for Obstacle Recognition of UGV. Conference of Korean Society of Automotive Engineers, **(2012)**, 1339-1343.

[4] A. Ahmad.; L. Dey.; A K-Means Clustering Algorithm for Mixed Numeric and Categorical Data. Data & Knowledge Engineering, **(2007)**, 63(2), 503-527.

[5] S. H. Yang.; B. S Song.; J. Y. Um.; Radar and Vision Sensor Fusion for Primary Vehicle Detection. Journal of Institute of Control, Robotics and Systems, **(2010)**, 16(7), 639-649.

[6] O. Chavez-Garcia.; O. Aycard.; Multiple Sensor Fusion and Classification for Moving Object Detection and Tracking. IEEE Transactions on Intelligent Transportation Systems, **(2016)**, 17(2), 525-534.

[7] I. S. Weon.; J. K. Ryu.; S. G. Lee.; Recognition of Object Tracking with Radar for USV. Conference of Korean Society for Precision Engineering, **(2017)**, 299-300.

[8] I. S. Weon.; J. K. Ryu.; S. G. Lee.; Obstacle Avoidance of Unmanned Surface Vehicle Based on 3D Lidar for VFH Algorithm. Asia-Pacific Journal of Multimedia Service Convergent with Art, Humanities, and Sociology, **(2018)**, 8(3), 945-953.

[9] I. S. Weon.; J. K. Ryu.; S. G. Lee.; Virtual Bubble Filtering Based on Heading Angle and Velocity for Unmanned Surface Vehicle (USV). International Conference on Control, Automation, and Systems (ICCAS), **(2017)**, 1954-1958.

[10] J. Redmon.; You Only Look Once: Unified, Real-Time Object Detection. IEEE Conference on Computer Vision and Pattern Recognition, **(2016)**.

Authors

Ihn Sik Weon
is Master and Doctor unify course in Department of Mechanical Engineering, KyungHee University, Korea. He has done his B.S in Control and instrument engineering from Korea University. Presently he is carrying his research work as development of position accuracy for autonomous driving and control vehicle, mobility system.

Soon Geul Lee
is a professor in Department of Mechanical Engineering, KyungHee University, Korea. He has done his Ph.D in Human Engineering from University of Michigan. His areas of specialization are Control mobility system, balancing robot and mobility system by CMG and Detection environment indoor and outdoor for autonomous driving.