Deep-Learning Based Pedestrian Direction Detection for Anti-collision of Intelligent Self-propelled Vehicles

Shih-Chieh Lin, Min-Chi Lin, Yin-Tsung Hwang, and Chih-Peng Fan*, Member, IEEE
Department of Electrical Engineering, National Chung Hsing University, Taiwan, R.O.C.
Email: cpfan@dragon.nchu.edu.tw*

***Abstract*— In this work, an effective deep learning based object detection and recognition method by the modified YOLO (You only look once) based model is developed for pedestrian direction detections. The proposed YOLO based detection model has lightweight network structure, needs less computing dependence, and performs high efficiency for detections. The proposed pedestrian direction detection technology provides the direction and bounding box information of pedestrians, and the proposed design enables the anti-collision function of intelligent self-propelled vehicles when it is moving in crowds. Compared with the direct YOLOv2 design, the performance of precision is better, and the frames per second in software implementation are also larger by the modified YOLO based detector.**

Keywords—Deep learning, Pedestrian direction and object recognition, YOLO-based model, Software implementation

I. INTRODUCTION

In recent years, the deep learning technologies [1] have become core technologies in artificial intelligence applications, which can simulate the operations of human neural networks, especially in the fields of visual recognitions and speech recognitions. In [2], the designers implemented the network architecture, named YOLO, without using the previously complex deep learning frameworks. The YOLO series networks [2, 3, 4], which used deep learning algorithms, can be applied for object detections efficiently. The YOLO network [2] is lightweight, high efficient, and valuable in many applications, such as the pedestrian detection, industrial image recognition, etc. In this work, the YOLOv2-based network [3] is applied to develop the inference engine for pedestrian direction and object recognitions, and the benefits of deep learning inference are maximized by a large number of labeled pedestrian data, and then the directions and objects of pedestrians are recognized. Besides pedestrian object detection, the information of pedestrian directions is also very useful for anti-collision application of intelligent self-propelled vehicles, as shown in Figure 1.

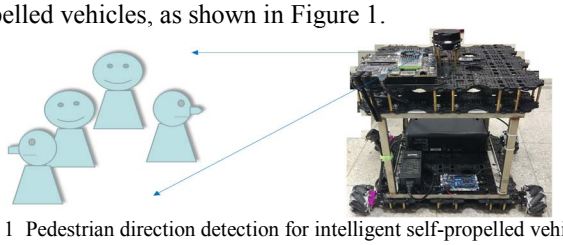

Fig. 1 Pedestrian direction detection for intelligent self-propelled vehicle

II. PEDESTRIAN DIRECTION DETECTION BY DEEP-LEARNING

A. *Preparing Image Databases for Traing and Testing Pedestrian Direction Detector*

The PASCAL VOC database [5] is a well-known data set for image recognitions, where the VOC 2012 version is used in our design. Moreover, the Penn-Fudan database [6] is the useful image database for pedestrian detection experiments. The Penn-Fudan database has a total of 170 images, and all images include the scenes around the campus and city streets. Each picture in the Penn-Fudan database at least has one pedestrian. Besides, the INRIA Person dataset [7] is also applied for this study, and the dataset is collected as testing images to develop the object detector of upright pedestrians. By integrating the above mentioned databases as shown in Figure 2, the suitable images are relabeled by the requirements, and the unsuitable images are removed and are not used to train the proposed pedestrian direction detector.

Fig. 2 Image databases [5, 6, 7] for pedestrian direction detections

B. *YOLO based Pedestrian Direction Recognition*

YOLO [2] is an effective deep learning algorithm for object detections, and the network is illustrated in Figure 3. The traditional object detection scheme consists of three processing stages, which are object localization, features extraction, and image classification. Compared with previous object detection methods, YOLO can directly predict the bounding boxes and class probabilities of objects in each image by simply evaluating a single neural network, and then YOLO can greatly shorten the operational time.

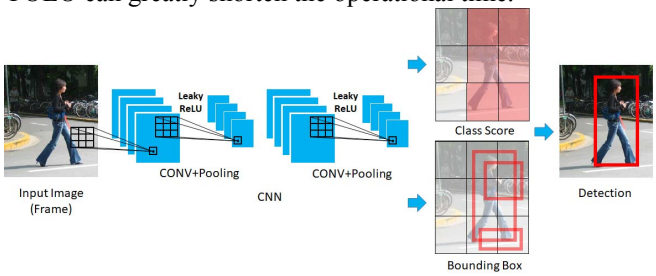

Fig. 3 Object detection by YOLOv2 deep learning network [4]

In previous deep learning methods, R-CNN (Region Convolutional Neural Networks) is well-known, and it firstly proposes several region proposals containing objects, and then the network classifies each region through convolutional neural networks (CNN). Finally, through the regression operation, the CNN network corrects the positions of bounding boxes for objects. The most significant characteristic of YOLO is to perform object detection directly through end-to-end models, and the network uses the entire image as input to the neural network to predict the positions of bounding boxes directly. The bounding box contains the confidence score of object and the category to which the object belongs. The calculation of YOLO network is fast, and it can reach the real time applications.

Compared with the direct YOLOv2 model [3, 4], the proposed YOLO-based model are improved by: (1) reduce three convolution layers to speed up the operations, (2) reduce the number of filters in the converution layers to decrease the dimension of feature map output in each layer for low complexity, (4) add a residual structure (i.e. bypass branch) to raise the performance of precision, and the bypass branch uses 1/8 size feature map, which is extracted from the original YOLOv2. The used design flow is described as follows:

978-1-7281-3575-5/19/$31.00 ©2019 IEEE

Stage 1 - Preparing labeled pedestrian images: In Figure 4, the pedestrian directions are divided into six directions, which are front (F), left front (FL), right front (FR), left (L), right (R), and rear (X) directions. Each collected image contains several pedestrians, and different pedestrians in an image are labeled with the individual directions and the corresponding bounding boxes by the semi-automatic tool, i.e. LabelImg [9].

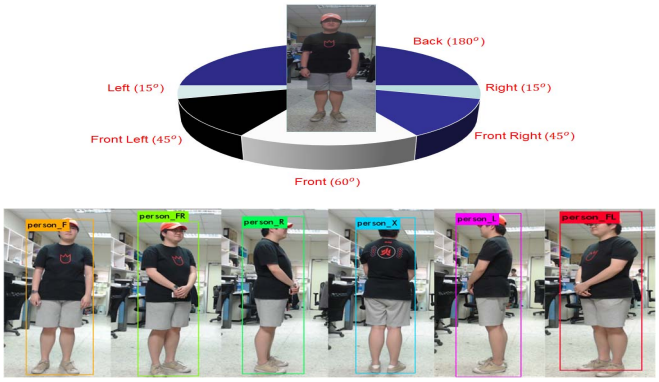

Fig. 4 Six pedestrian directions for recognitions in our design

Stage 2 - Training and testing an inference model by the YOLOv2 based network: Figure 5 shows the training/testing flows to obtain the YOLOv2-based inference model. The procedure has three parts, which are data pre-processing, training the inference model, and testing the inference model. After setting the required convolution layer and normalizations, the weights of YOLOv2-based network, which are pre-trained by Darknet, are loaded when the pre-processing process is enabled. Since the last layer of the convolution layer determines the final output, the convolution layer must be re-adjusted and trained to achieve a fine-tuned training model. After the inference model is trained, the testing procedure is followed. In the testing stage, the highest final confidence score is computed, and then the detection accuracy and recall rate [8] will be calculated and obtained.

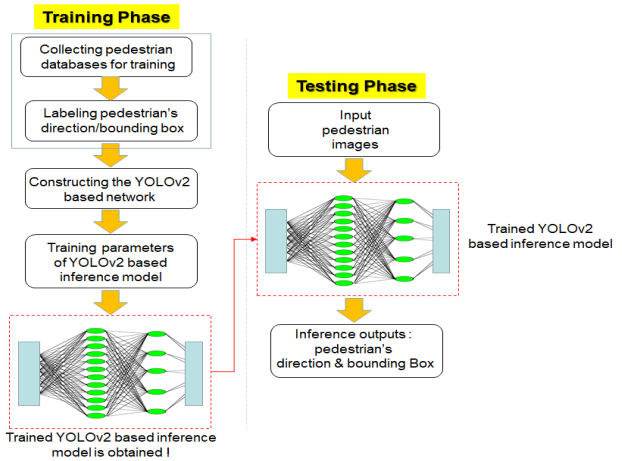

Fig. 5 Training/testing flows to obtain the YOLOv2-based inference model

III. Experimental Results and Comparisons

The personal computer with a CPU by 3.70GHz operational frequency and a GPU acceleration interface card is used in experimental simulations and functional verifications. In the pedestrian image databases, after screening, 5261 images are available for training and testing, where 4736 images are used for training samples, and the remaining 525 images are used for validations. The proposed YOLOv2 based model is trained under the Darknet architecture. After training, the values of recall and precision for validations are calculated, and the performance comparisons are shown in Table 1. For the proposed design, the recall is 77%, and the precision is up to 82%, which is better than the direct YOLOv2 model. Because the number of convolutional layers and kernels in YOLO-PD is reduced, the recall of YOLO-PD is lower than that of YOLOv2. On the other size, due to the improved architecture of bypass and anchor boxes in YOLO-PD, the precision of YOLO-PD is higher than that of YOLOv2. Figure 6 demonstrates the testing results. In Table 2, for software implementations by NVIDIA Xavier platform, the frames per second (FPS) by the proposed design are also higher than those by the direct YOLOv2 design.

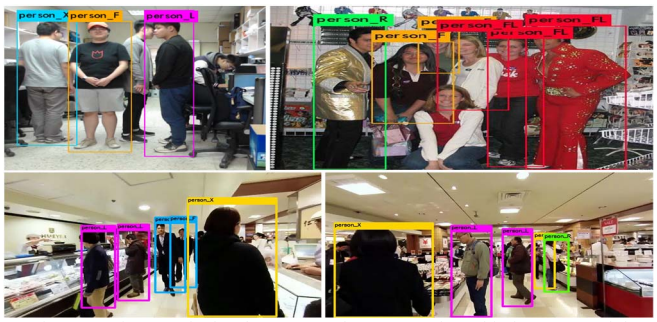

Fig. 6 Testing results by the proposed YOLOv2-based inference model

Table 1 Performance comparison of pedestrian direction detection between two different YOLOv2-based designs

Methods	YOLOv2	Proposed
Recall	81%	77%
Precision	78%	**82%**

Table 2 Performance comparison of frames per second when software inference models are implemented with NVIDIA Xavier platform

Software implementation with NVIDIA Xavier platform	YOLOv2	Proposed
Frames per second (FPS)	20	**30**

IV. Conclusion

In this paper, the modified YOLOv2-based deep learning network is used to implement the pedestrian direction detector, and the precision of the direction recognition can be up to 82%. Compared with the direct YOLOv2 design, the frames per second in software implementation are also higher by using the modified YOLO based deep learning detector.

Acknowledgment

This work was supported by Ministry of Science and Technology, Taiwan, R.O.C. under Grant MOST 107-2218-E-005-014. Thank Prof. Kuang-Hao Lin in National Formosa University for supporting the platform of intelligent self-propelled vehicle.

References

[1] Szegedy, Christian, et al. "Going Deeper with Convolutions," IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[2] Joseph Redmon, et al. "You only look once: Unified, Real-Time Object Detection." IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[3] Joseph Redmon, and Ali Farhadi. "YOLO9000: Better, Faster, Stronger," IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[4] Hiroki Nakahara, et. al, "A Lightweight YOLOv2: A Binarized CNN with A Parallel Support Vector Regression for an FPGA," the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp.31-40, 2018.

[5] http://host.robots.ox.ac.uk/pascal/VOC/voc2012/

[6] https://www.cis.upenn.edu/~jshi/ped_html/

[7] http://pascal.inrialpes.fr/data/human/

[8] David M. W. Powers, et al, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," Journal of Machine Learning Technologies, Vol. 2, No. 1, pp. 37-63, 2011.

[9] https://github.com/tzutalin/labelImg