Journal of Theoretical and Applied Information Technology
30th November 2020. Vol.98. No 22
© 2005 – ongoing JATIT & LLS

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

DETECTION APPROACH BASED ON MULTI-HEAD STRUCTURE AND ENHANCED FEATURES IN DRIVING ENVIRONMENTS

HOANH NGUYEN

Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

E-mail: nguyenhoanh@iuh.edu.vn

ABSTRACT

Most recent state-of-the-art detection approaches based on deep convolutional neural networks are manually designed. These approaches include two-stage frameworks and one-stage frameworks. While one-stage frameworks provide real-time performance in most recent systems, two-stage frameworks usually show better detection accuracy. Most recent two-stage object detection frameworks share a head for both classification and bounding box regression in detection stage. Inspired by recent improvement in double-head structures, this paper proposes a detection framework based on multi-head structure for localizing objects in driving environments. First, the extracted feature maps generated by feature extraction network are enhanced by the enhancement module, which effectively enlarges the receptive field and refines the representation ability of thin feature maps by leveraging both local and global context. The enhanced feature map is then fed to a detection network. Next, the detection network is designed based on double-head structure, where a fully connected head is adopted for classification and a convolution head is used for bounding box regression. In addition, this paper proposes to improve RoI pooling algorithm based on deformable RoI pooling. With the improved RoI pooling process, the harsh quantization of RoI pooling is removed, and the extracted features are properly aligning with the input, thus leading to large improvements. Experiments on public datasets show the effectiveness of the proposed method for localizing objects in driving environments.

Keywords: *Detection Approaches, Double-Head Structures, Multi-Head Structure, Deep Convolutional Neural Networks, Two-Stage Framework*

1. INTRODUCTION

With the fast development of deep learning in recent years, a variety of detection approaches based on deep learning have been proposed. The deep convolution neural networks (CNNs) can learn the features of the objects to be detected with the dataset autonomously and improve the performance of its model gradually. CNNs mainly consist of three type of layers: convolutional layers, which uses a filter of weights to extract features from image; nonlinear layers, which apply an activation function on feature maps to enable the modeling of non-linear functions by the network; and pooling layers, which replace a small region of a feature map with some statistical information to reduce spatial resolution. Each unit in every layer receives weighted inputs from a small region of units in the previous layer. This small region is called receptive field. In CNNs, the higher-level layers learn features from increasingly wider receptive fields. The main computational advantage of CNNs is that all the receptive fields in a layer share weights, resulting in a significantly smaller number of parameters than fully connected neural networks. Since the development of fully convolutional networks [12], the accuracy of detection approaches has been improved rapidly. These detection approaches include one-stage frameworks and two-stage frameworks. In one-stage frameworks, the detection head is applied directly on multi-scale feature maps generated by the base network, thus enhancing the detection speed. OverFeat [13] detects objects by sliding windows on feature maps. SSD [14] and YOLO [15] have been tuned for speed by predicting object classes and locations directly. RetinaNet [16] alleviates the extreme foreground-background class imbalance problem by introducing focal loss. Point-based methods [17] model an object as keypoints (corner, center), and are built on keypoints estimation networks. In two-stage framework, the detection head is applied after region proposals generation stage, where a small network is applied to generate proposals. Two-stage frameworks usually provide better detection accuracy compared with that of one-stage frameworks. RCNN [18] applies a deep neural network to extract features from proposals generated by selective search. SPPNet [19] speeds up RCNN significantly using spatial pyramid pooling. Fast RCNN [20] improves the speed and performance utilizing a differentiable RoI Pooling. Faster RCNN [21] introduces Region Proposal Network (RPN) to generate proposals. RFCN [22] employs position sensitive RoI pooling to address the translation-variance problem. FPN [4] builds a top-down architecture with lateral connections to extract features across multiple layers. Both two-stage and one-stage frameworks require a state-of-the-art CNN architecture as the base network for the best performance. Recent deep CNN-based architectures require a large amount of computational cost. While these architectures achieved high performance on for resource constrained devices such as mobile devices and embedded computers. It is required that the deep CNN architecture should be lightweight and efficient while achieving comparable accuracy to implement on resource constrained devices. Thus, many enhanced networks for mobile devices have been introduced recently. Mobilenets [23] used depth-wise separable convolutions that factor a convolution into two steps to reduce computational complexity: depth-wise convolution that performs light-weight filtering by applying a single convolutional kernel per input channel and pointwise convolution that usually expands the feature map along channels by learning linear combinations of the input channels. Mobilenetsv2 [24] proposed a lightweight network based on an inverted residual structure where the shortcut connections are between the thin bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. In [25], a lightweight and efficient network based on depthwise dilated separable convolution was proposed. Shufflenet [26] and Shufflenetv2 [3] proposed new architecture that utilizes two new operations, pointwise group convolution and channel shuffle, to greatly reduce computation cost while maintaining accuracy.

In the line of two-stage deep learning-based object detectors, R-CNN is a pioneer deep learning model, which increases object detection accuracy over traditional detectors by a large margin. In the first stage, R-CNN applies selective search method [28] to generate sufficient proposal candidates that contain all the objects. In the second stage, R-CNN forwards each proposal through convolutional networks, followed by classifying the proposals with SVMs [29] and predicting bounding boxes offsets with linear regression. However, this method is very time-consuming, as every proposal is processed by the entire network. Fast R-CNN extends R-CNN by using one single convolution network to perform shared computation in the second stage, which increases the speed significantly. The problem with Fast R-CNN is that the proposals are generated by a traditional time-consuming selective search algorithm. Faster R-CNN was proposed to further improve upon Fast R-CNN. Faster-RCNN proposed region proposal network (RPN) to replace selective search method in R-CNN and makes the whole network trainable in an end to end approach. Recently, several approaches have been proposed to increase the accuracy of Faster R-CNN. Instead of using VGG-16 architecture as a base network for Faster R-CNN, adoption of different backbone networks, such as ResNet and Inception ResNet, has been proposed. He et al. [9] proposed the use of a deep residual network, such as ResNet-101, for image recognition. The authors showed that ResNet-101 has a lower complexity compared to VGG-16 and achieves good accuracy. Huang et al. [30] used an Inception ResNet v2 in the backbone of the Faster R-CNN to achieve better accuracy than that obtained using ResNet 101 with a slightly lower running time per frame. Shrivastava et al. [31] proposed a top–down modulation (TDM) network to incorporate fine details in the detection network for detecting small objects. They achieved higher accuracy compared to [30] with a slightly higher frame rate. Yauan et al. [32] proposed two refinement methods, iterative and LSTM refinement, for the Faster R-CNN model and improved the accuracy.

2. METHODOLOGY

The overall structure of the proposed approach is shown in Figure 1. The feature extraction network first extracts features from input images. The

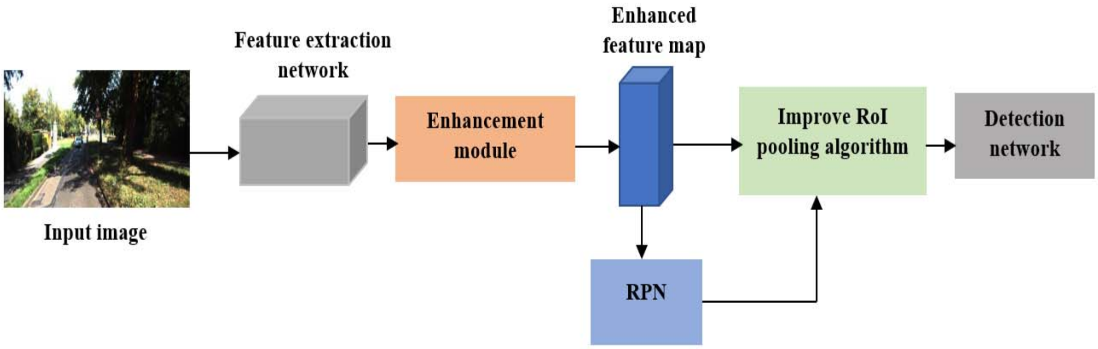

Figure 1: The Overall Structure of The Proposed Approach.

extracted feature maps are then enhanced by the enhancement module, which effectively enlarges the receptive field and refines the representation ability of the thin feature maps by leveraging both local and global context. The enhanced feature map is then fed to a detection network. The detection network is designed based on double-head structure, where a fully connected head is adopted for classification and a convolution head is used for bounding box regression. In addition, this paper proposes to improve RoI pooling algorithm based on deformable RoI pooling. With the improved RoI pooling process, the harsh quantization of RoI pooling is removed, and the extracted features are properly aligning with the input, thus leading to large improvements. Details of each module will be elaborated in the following subsections.

2.1 Feature Extraction Network

The feature extraction network extracts basic feature representations of input images and has big influence on both accuracy and efficiency of the whole framework. A lightweight network used for extraction will facilitate the inference speed and the computational cost. However, lightweight network may generate feature maps with less discriminative feature representations, thus reducing the detection performance of the framework. In addition, the receptive field size in each layer of the feature extraction network plays a crucial role in the network. Each layer can only capture information inside the receptive field. Thus, a large receptive field can leverage more context information and encode long-range relationship between pixels more effectively. This is an important problem for the localization subtask, especially for the localization of large objects. Previous works [1, 2] have also demonstrated the effectiveness of the large receptive field in semantic segmentation and object detection.

With above analysis, this paper adopts ShuffleNetV2 architecture [3] to build the feature extraction network for extracting feature representations from input images. ShuffleNetv2 is a lightweight deep CNN network which achieves the best accuracy in very limited computational budgets. By shuffling the channels, ShuffleNetv2 outperformed MobileNetV1, MobileNetv2, and ShuffleNetv1 in both accuracy and computational cost. Based on the ShuffleNetv2, this paper first replaces all 3×3 depthwise convolution layers in Shuffle Unit by 5×5 depthwise convolution layers to improve the detection performance. By using 5×5 depthwise convolution layers, the receptive field is enlarged to capture more semantic information, while providing similar computational budget to 3×3 convolution layers. The structure of the improved ShuffleNetv2 network is shown in Table 1. There are total 6 layers in the architecture of the improved ShuffleNetv2 network. The number of channels of the final layer is 1024. The last output feature maps of layer 4 and layer 5 are denoted as C4 and C5.

Next, to enhance feature map before feeding to the detection network, many studies were inclined to fuse multi-scale feature map at different layers. A common technique adopting this scheme is Feature Pyramid Network (FPN) [4]. However, prior FPN structures [4, 5, 6, 7] involve many extra convolutions and multiple detection branches, which increases the computational cost and induces enormous runtime latency. For this reason, this paper designs an efficient enhancement module to enhance feature map generated by the feature extraction network before feeding to the detection network. The key idea of the proposed enhancement module is to aggregate multi-scale local context information and global context information to generate more discriminative features. Figure 2 presents the overall structure of the enhancement module proposed in

Table 1: The Feature Extraction Network Architecture.

Layer	Type	Kernel Size	Stride	Repeat	Output Size	Output Channel
0	Conv1	3×3	2	1	112×112	24
1	MaxPool	3×3	2	1	56×56	24
2	Shuffle Unit	5×5 depthwise convolution layers	2 1	1 3	28×28	176
3	Shuffle Unit	5×5 depthwise convolution layers	2 1	1 7	14×14	352
4	Shuffle Unit	5×5 depthwise convolution layers	2 1	1 3	7×7	704
5	Conv5	1×1	1	1	7×7	1024

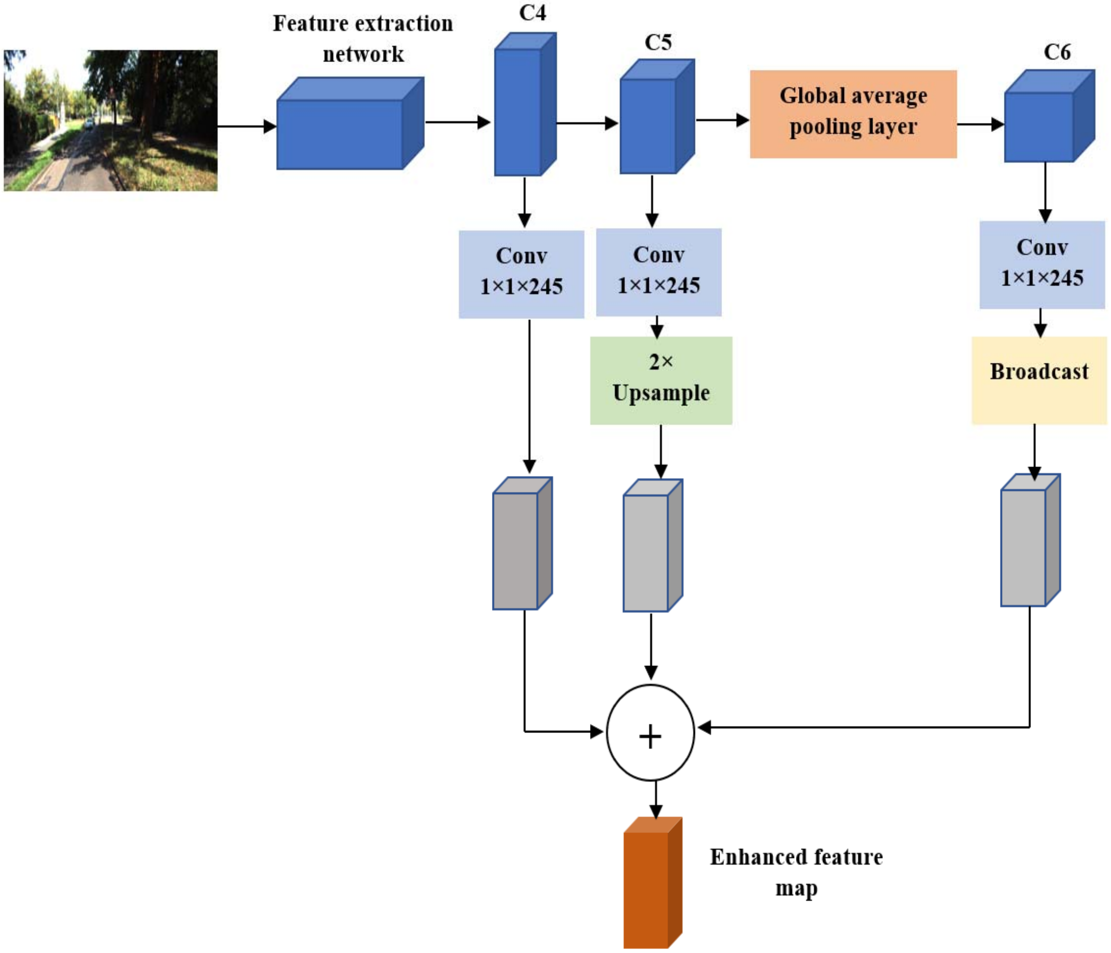

Figure 2: The Overall Structure of The Enhancement Module.

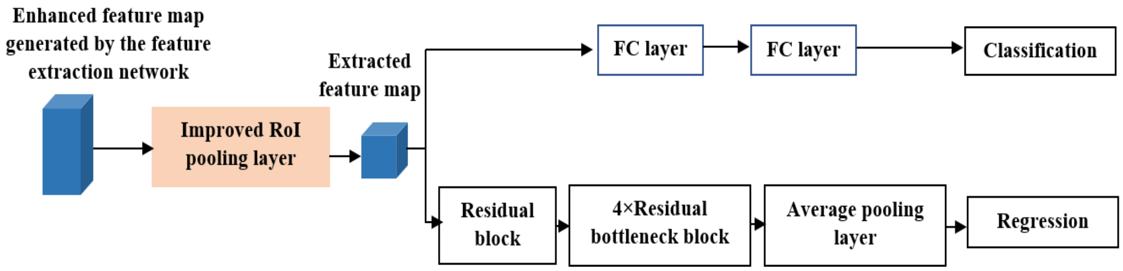
Enhanced feature map generated by the feature extraction network
Improved RoI pooling layer
Extracted feature map
FC layer
FC layer
Classification
Residual block
4×Residual bottleneck block
Average pooling layer
Regression

Figure 3: The Structure of The Double-Head Scheme Used in This Paper.

this paper. As shown, the enhancement module fuses the feature maps from three scales, including C4, C5 and C6. C4 and C5 are the last output feature map of layer 4 and layer 5 of the improved ShuffleNetv2 network, and C6 is the global context feature vector generated by applying a global average pooling on C5. A 1×1 convolution layer is then applied on each feature map to squeeze the number of channels to 245. As a result, C5 is upsampled by 2× and C6 is broadcast so that the spatial dimensions of the three feature maps are equal. Finally, the three generated feature maps are aggregated. By leveraging both local and global context, the enhancement module effectively enlarges the receptive field and refines the representation ability of the thin feature map. Compared with prior FPN structures, the enhancement module involves only two 1×1 convolution layers and a FC layer, which is more computation friendly.

2.2 Detection Network

Inspired by the double-head scheme [8], which adopts two separate branches to leverage the advantages of two head structures, this paper uses double-head structure to design the detection network. Double-head scheme uses the fully connected head for classification and the convolution head for bounding box regression. The structure of the double-head scheme is shown in Figure 3. The fully connected head has two fully connected layers. The number of channels of output feature is 1024. The convolution head used in this paper stacks 5 residual blocks [9]. The first block increases the number of channels from 256 to 1024 (shown in Figure 4a), and four other blocks are bottleneck blocks (shown in Figure 4b). At the end, an average pooling layer is used to generate the feature vector with 1024 channels.

For loss function, the detection network with double-head structure and the region proposal network (RPN) are jointly trained end to end. The total loss is defined as follows:

$$L = \partial L_{FC} + \varphi L_{CV} + L_{RPN} \quad (1)$$

where ∂ and φ are weights for the fully connected head and the convolution head, respectively. L_{FC}, L_{CV}, and L_{RPN} are the losses for the fully connected head, the convolution head and the RPN, respectively.

2.3 Improving RoI Pooling Algorithm

Deformable RoI pooling is introduced in [10] to mitigate the misalignments between the RoI and the extracted features in RoI pooling process. In Deformable RoI pooling process, pooled feature map is first generated by adopting regular RoI pooling. From the pooled feature map, a fully connected layer is used to generate the normalized offsets, which are then added to the spatial binning positions. The offset normalization is necessary to make the offset learning invariant to RoI size. After generating offsets, the deformable RoI pooling employs RoI pooling to generate the output feature map based on input regions with augmented offsets. Inspired by deformable RoI pooling, this paper proposes to improve RoI pooling algorithm based on deformable RoI pooling as shown in Figure 5. First, this paper uses a lightweight offset prediction branch which contains fewer parameters than the deformable RoI pooling. More specific, the lightweight offset prediction branch adopts RoIAlign to obtains features from k/2×k/2 sub-regions followed by a fully connected layer. With smaller input vector of features, the number of parameters in subsequence layer will decrease. Next, the standard deformable RoI pooling employs regular RoI pooling in the fixed size feature map generation branch to generate the output feature map based on input regions with augmented offsets. In

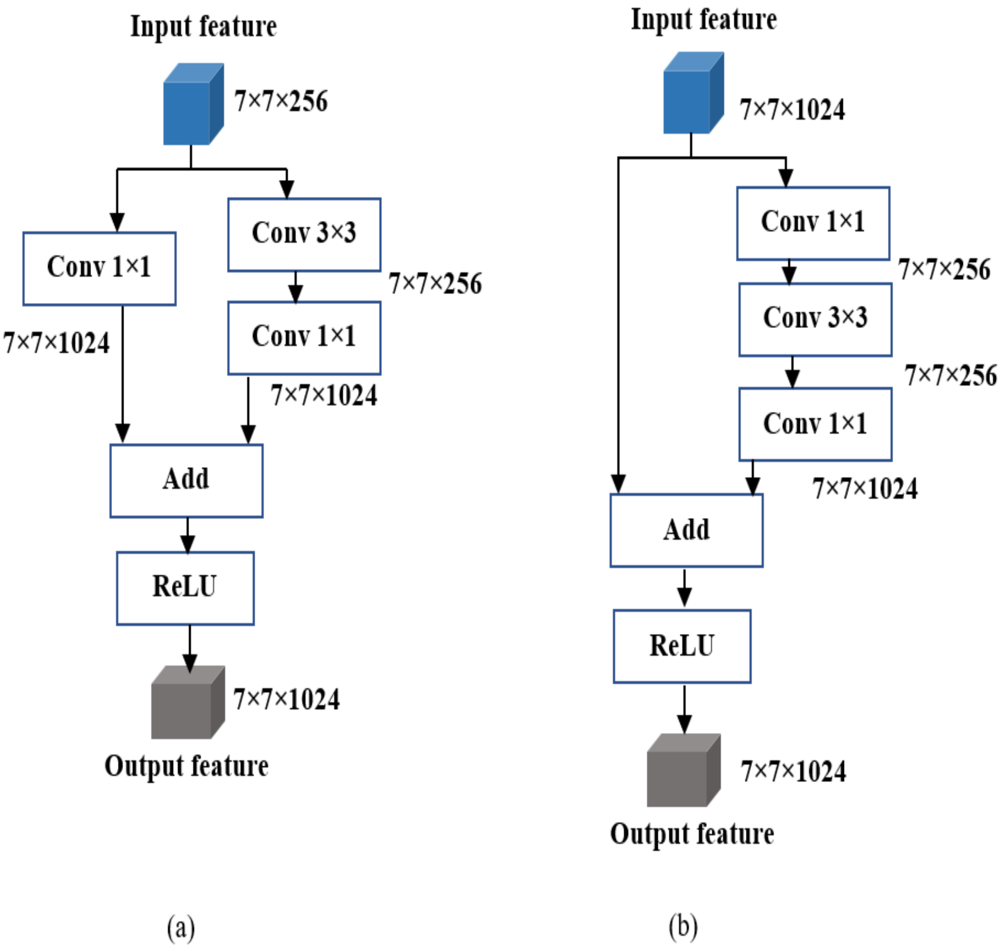

Figure 4: Residual Blocks.

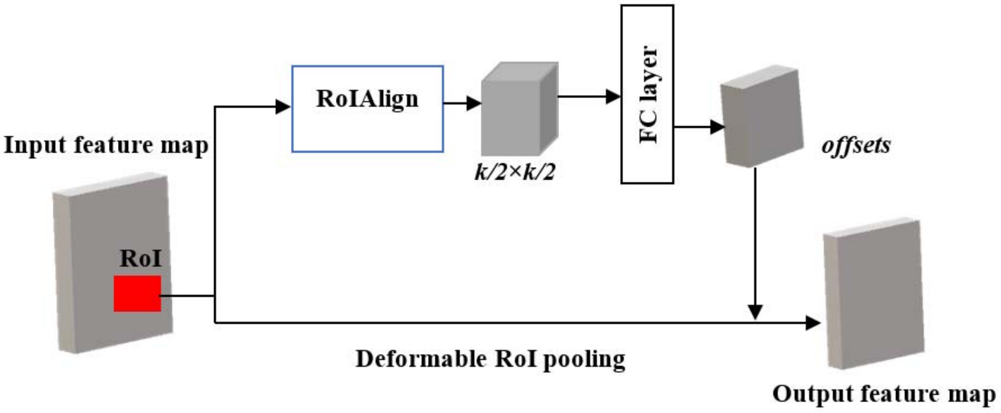

Figure 5: Improving RoI Pooling Algorithm.

Table 2: Detection Results on All Three Difficulty-Level Groups of The KITTI Test Set.

Method	AP (%)					
	Car			Pedestrian		
	Easy	Moderate	Hard	Easy	Moderate	Hard
Faster R-CNN [5]	86.71	81.84	71.12	76.21	62.14	60.33
SSD [11]	77.71	64.06	56.17	25.12	18.20	16.21
YOLOv2 [12]	76.79	61.31	50.25	22.16	16.16	15.82
MS-CNN [13]	90.03	89.02	76.11	84.12	74.98	63.48
Proposed Method	90.21	89.50	75.22	88.11	75.62	65.78

contrast, this paper adopts RoIAlign [11] in the fixed size feature map generation branch to generate the output feature map based on input regions with augmented offsets. As a result, the harsh quantization of RoI pooling is removed, and the extracted features are properly aligning with the input, thus leading to large improvements.

3. EXPERIMENTS

3.1 Dataset and Evaluation Metrics

In order to evaluate the effectiveness of the proposed approach for localizing objects in driving environments, this paper conducts experiments on widely used public dataset: KITTI dataset [27]. KITTI dataset is a widely used dataset for evaluating object detection algorithms in driving environments. This dataset consists of 7481 images for training with available ground-truth and 7518 images for testing with no available ground-truth. Images in this dataset include various scales of vehicle and pedestrian in different scenes and conditions and were divided into three difficulty-level groups: easy, moderate, and hard. If the bounding boxes size was larger than 40 pixels, a completely unshielded vehicle/pedestrian was considered to be an easy object, if the bounding boxes size was larger than 25 pixels but smaller than 40 pixels, a partially shielded vehicle/pedestrian was considered as a moderate object, and an vehicle/pedestrian with the bounding boxes size smaller than 25 pixels and an invisible vehicle/pedestrian that was difficult to see with the naked eye were considered as hard objects.

For evaluation metrics, this paper uses the average precision (AP) and intersection over union (IoU) metrics [27] to evaluate the performance of the proposed method in all three difficulty level groups of the KITTI dataset. These criteria have been used to assess various object detection algorithms. As in [27], the IoU is set to 0.7 for vehicle and 0.5 for pedestrian in this paper, which means only the overlap between the detected bounding box and the ground truth bounding box greater than or equal to 70% and 50% is considered as a correct detection.

3.2 Experimental Results

This section presents the detection results of the proposed method and recent methods on the KITTI dataset. First, this paper conducts experiments on the KITTI test set by using the proposed model and recent models to compare the detection performance. The reference models include SSD [13], Faster R-CNN [7], YOLOv2 [14], and MS-CNN [15]. All models are implemented on NVIDIA GTX 1080 GPU. Table 2 presents the detection results of the proposed model and reference models on all three difficulty-level groups of the KITTI test set. As shown in Table 2, the proposed model obtains 90.21%, 89.50%, and 75.22% of the AP on easy, moderate, and hard group, respectively for car detection. For pedestrian detection, the proposed model obtains 88.11%, 75.62%, and 65.78% of the AP on easy, moderate, and hard group, respectively. It can be observed that the proposed model achieves superior results to state-of-the-art object detectors, including both one-stage and two-stage detectors such as Faster R-CNN,

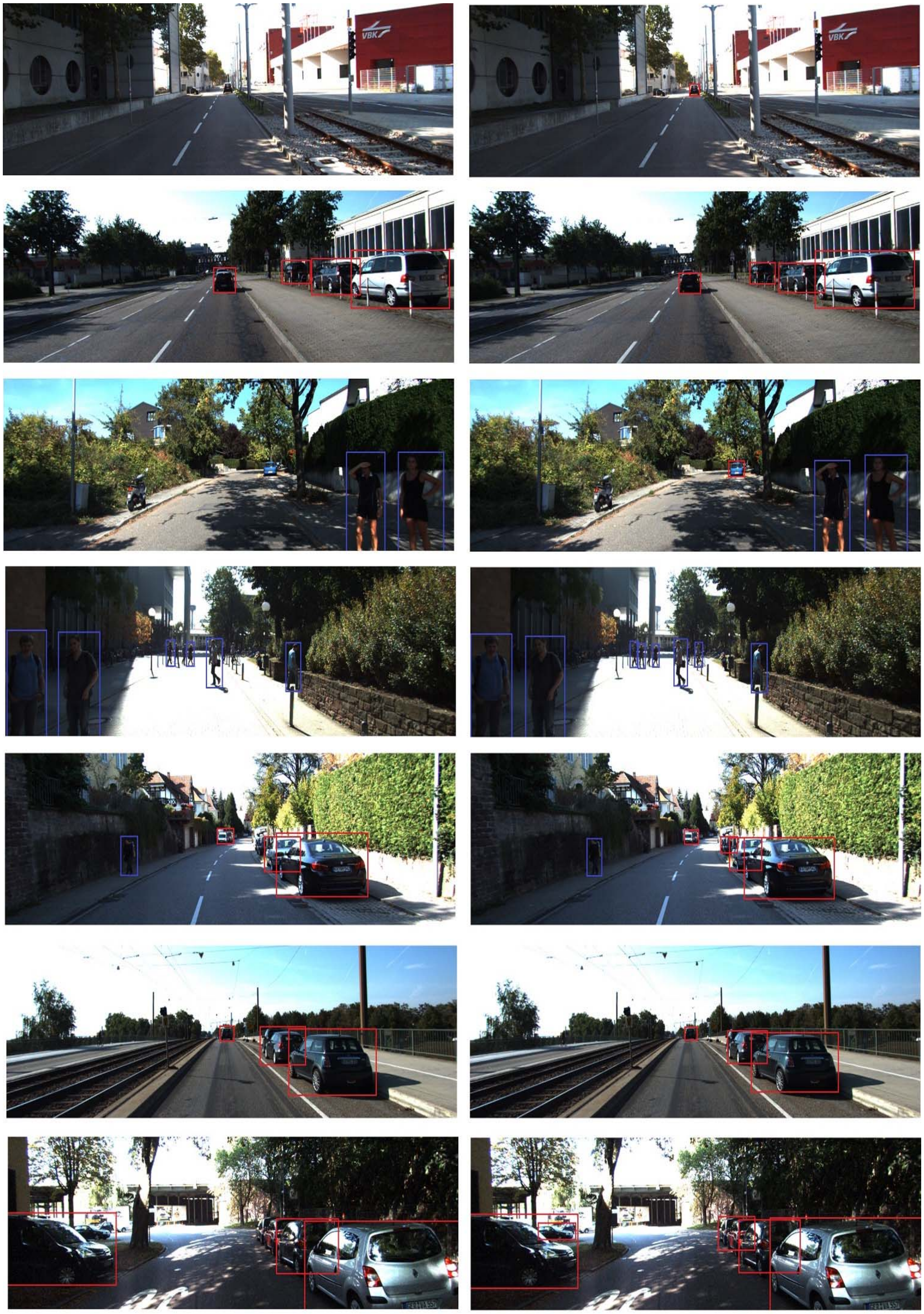

Figure 6: Visual of Detection Results on The KITTI Dataset. (Left) Faster R-CNN. (Right) The Proposed Method.

SSD, and YOLOv2. Compared with MS-CNN, the proposed method achieves better detection results overall. These results demonstrate that the proposed method achieves a much better accuracy on the KITTI dataset. Figure 6 shows visual of detection results on the KITTI dataset of Faster R-CNN (left) and the proposed method (right). As shown, Faster R-CNN misses some cars and pedestrians, while the proposed method localizes exactly cars and pedestrians in images.

4. CONCLUSIONS

This paper proposes a detection framework based on multi-head structure for localizing objects in driving environments. In the proposed framework, the feature extraction network first extracts features from input images. The extracted feature maps are then enhanced by the enhancement module, which effectively enlarges the receptive field and refines the representation ability of the thin feature maps by leveraging both local and global context. The enhanced feature map is then fed to a detection network. The detection network is designed based on double-head structure, where a fully connected head is adopted for classification and a convolution head is used for bounding box regression. In addition, this paper proposes to improve RoI pooling algorithm based on deformable RoI pooling. With the improved RoI pooling process, the harsh quantization of RoI pooling is removed, and the extracted features are properly aligning with the input, thus leading to large improvements. Experiments on the KITTI datasets show the effectiveness of the proposed method for localizing objects in driving environments.

REFERENCES:

[1] Li, Zeming, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. "Light-head r-cnn: In defense of two-stage object detector." *arXiv preprint arXiv:1711.07264* (2017).

[2] Peng, Chao, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. "Large kernel matters--improve semantic segmentation by global convolutional network." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4353-4361. 2017.

[3] Ma, Ningning, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. "Shufflenet v2: Practical guidelines for efficient cnn architecture design." In *Proceedings of the European conference on computer vision (ECCV)*, pp. 116-131. 2018.

[4] Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. "Feature pyramid networks for object detection." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117-2125. 2017.

[5] Fu, Cheng-Yang, Wei Liu, Ananth Ranga, Ambrish Tyagi, and Alexander C. Berg. "Dssd: Deconvolutional single shot detector." *arXiv preprint arXiv:1701.06659* (2017).

[6] Li, Yuxi, Jiuwei Li, Weiyao Lin, and Jianguo Li. "Tiny-DSOD: Lightweight object detection for resource-restricted usages." *arXiv preprint arXiv:1807.11013* (2018).

[7] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* (2018).

[8] Wu, Yue, Yinpeng Chen, Lu Yuan, Zicheng Liu, Lijuan Wang, Hongzhi Li, and Yun Fu. "Rethinking Classification and Localization for Object Detection." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10186-10195. 2020.

[9] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778. 2016.

[10] Dai, Jifeng, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. "Deformable convolutional networks." In *Proceedings of the IEEE international conference on computer vision*, pp. 764-773. 2017.

[11] He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask r-cnn." In *Proceedings of the IEEE international conference on computer vision*, pp. 2961-2969. 2017.

[12] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431-3440. 2015.

[13] Sermanet, Pierre, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. "Overfeat: Integrated recognition, localization and detection using convolutional

networks." *arXiv preprint arXiv:1312.6229* (2013).

[14] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "Ssd: Single shot multibox detector." In *European conference on computer vision*, pp. 21-37. Springer, Cham, 2016.

[15] Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. "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.

[16] Lin, Tsung-Yi, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. "Focal loss for dense object detection." In *Proceedings of the IEEE international conference on computer vision*, pp. 2980-2988. 2017.

[17] Law, Hei, and Jia Deng. "Cornernet: Detecting objects as paired keypoints." In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 734-750. 2018.

[18] Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587. 2014.

[19] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Spatial pyramid pooling in deep convolutional networks for visual recognition." *IEEE transactions on pattern analysis and machine intelligence* 37, no. 9 (2015): 1904-1916.

[20] Girshick, Ross. "Fast r-cnn." In *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448. 2015.

[21] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." In *Advances in neural information processing systems*, pp. 91-99. 2015.

[22] Dai, Jifeng, Yi Li, Kaiming He, and Jian Sun. "R-fcn: Object detection via region-based fully convolutional networks." In *Advances in neural information processing systems*, pp. 379-387. 2016.

[23] Howard, Andrew G., Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861* (2017).

[24] Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "Mobilenetv2: Inverted residuals and linear bottlenecks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510-4520. 2018.

[25] NGUYEN, HOANH. "A LIGHTWEIGHT AND EFFICIENT DEEP CONVOLUTIONAL NEURAL NETWORK BASED ON DEPTHWISE DILATED SEPARABLE CONVOLUTION." *Journal of Theoretical and Applied Information Technology* 98, no. 15 (2020).

[26] Zhang, Xiangyu, Xinyu Zhou, Mengxiao Lin, and Jian Sun. "Shufflenet: An extremely efficient convolutional neural network for mobile devices." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6848-6856. 2018.

[27] Geiger, Andreas, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite." In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354-3361. IEEE, 2012.

[28] Uijlings, Jasper RR, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. "Selective search for object recognition." *International journal of computer vision* 104, no. 2 (2013): 154-171.

[29] Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." *Machine learning* 20, no. 3 (1995): 273-297.

[30] Huang, Jonathan, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer et al. "Speed/accuracy trade-offs for modern convolutional object detectors." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310-7311. 2017.

[31] Shrivastava, Abhinav, Rahul Sukthankar, Jitendra Malik, and Abhinav Gupta. "Beyond skip connections: Top-down modulation for object detection." *arXiv preprint arXiv:1612.06851* (2016).

[32] Yuan, Peng, Yangxin Zhong, and Yang Yuan. "Faster r-cnn with region proposal refinement." *Tech. Rep.* (2017).