Journal of Theoretical and Applied Information Technology
15th December 2019. Vol.97. No 23

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

ENHANCED VEHICLE DETECTION APPROACH USING DEEP CONVOLUTIONAL NEURAL NETWORKS

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

Vehicle detection plays an important role in autonomous driving systems. Recently, vehicle detection methods achieved large successes with the fast development of deep convolutional neural network (CNN). However, due to small size, heavy occlusion or truncation of vehicle in an image, recent CNN detectors still show a limited performance. This paper presents an improved framework based on deep CNN for vehicle detection. Firstly, deconvolutional modules are added at multiple output layers of the reduced VGG 16 architecture to enhance additional context information which is helpful to improve the detection accuracy. Secondly, region proposal modules are applied at different feature maps to address the vehicle occlusion challenge. Due to heavy object occlusion in test dataset, soft Non-Maximum Suppression (NMS) algorithm is used to solve the issue of duplicate proposals. Finally, a deep CNN-based classifier including a region of interest (ROI) pooling layer and a fully connected (FC) layer is used for classification and bounding box regression. The proposed method is evaluated on the KITTI vehicle dataset. Experimental results show that the proposed method achieves better performance compared to other the state-of-the-art approaches in vehicle detection.

Keywords: *Vehicle Detection, Convolutional Neural Network, Intelligent Transportation Systems, Object Detection, Deep Learning*

1. INTRODUCTION

Visual vehicle detection from images is an essential prerequisite for many intelligent transportation systems, with a wide range of real-world applications, such as ADAS and autonomous driving [1], intelligent traffic management systems and so on. In recent years, deep convolutional neural networks (CNNs) have achieved incredible success on vehicle detections as well as various other object detection tasks. However, when applying CNNs to vehicle detection, real time vehicle detection in driving environment is still very challenging. It is observed that the object detection performance of the popular CNN detectors including Faster-RCNN [6] and SSD [7] without modification is not very good over the KITTI benchmark datasets [1]. KITTI is the largest public dataset dedicated to ADAS and autonomous driving benchmarking. One of the main challenges is that traditional CNNs are sensitive to scales while it is quite common that in-car videos or transportation surveillance videos contain vehicles with a large variance of scales. Current methods are based on modifying the popular CNN detectors to enhance the performance of detection results. These methods focus on making the network fit different scales by utilizing input images with multiple resolutions. However, these methods introduce expensive computational overhead and thus are still incapable of fast vehicle detection, which is essential for autonomous driving systems, real time surveillance and prediction systems.

In view of the above research challenges, this paper proposes an enhanced approach to deep CNN framework to increase the visual vehicle detection accuracy. In this paper, deconvolution of CNN features is applied at smaller feature output scales, which is further fused with features at larger feature output scales, to provide richer context for vehicle detection at individual feature output scale. Such enhancement can effectively address the large object scale variation challenge. Furthermore, to address the object occlusion challenge, soft-NMS is applied at object proposals from different feature output scales to strike a balance on the number and quality of vehicle proposals. The proposed approach is

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

evaluated with various image input sizes by experiments over KITTI benchmark dataset. With this dataset, the proposed method achieves better detection results than other state-of-the-art methods.

This paper is organized as follows: an overview of previous methods is presented in Section 2. Section 3 describes detail the proposed method. Section 4 demonstrates experimental results. Finally, the conclusion is made in Section 5.

2. RELATED WORK

In this section, this paper introduces previous approaches, which are related to vehicle detection, including traditional methods and recently proposed methods based on deep CNN.

Traditional methods include motion-based methods and statistical learning-based methods. Motion-based methods use the motion to detect the vehicles. Adaptive background models such as Gaussian Mixture Model (GMM) [8], [9], [10], Sigma-Delta Model [11] are widely used in vehicle detection by modeling the distribution of the background as it appears more frequently than moving objects. In [8], the authors proposed to model each pixel as a mixture of Gaussians and using an on-line approximation to update the model. The Gaussian distributions of the adaptive mixture model are then evaluated to determine which are most likely to result from a background process. Z. Chen et al. [9] presented a new background Gaussian Mixture Model and shadow removal method to deal with sudden illumination changes and camera vibration. A Kalman filter tracks a vehicle to enable classification by majority voting over several consecutive frames, and a level set method has been used to refine the foreground blob. C. Premebida and U. Nunes [10] proposed a flexible multi-module architecture for a multi-target detection and tracking system complemented with a Bayesian object classification layer based on finite Gaussian mixture models. The parameters of Gaussian mixture model are estimated by an expectation maximization algorithm, hence finite-component models were generated based on feature-vectors extracted from object's classes during the training stage. In [11], the authors proposed an enhanced version of the sigma delta background estimation method, suitable for urban traffic scenes. Some heuristics have been added to the basic algorithm in order to make a selective background model updating at the pixel level. In [29], two-dimensional discrete wavelet transform is used first for extracting features from the images which has a good location property in time and frequency domains. Moreover, road detection is proposed to determine the zone of interest, this technique is used one time at the beginning of the processing to solve the problem of unimportant movement of the background and also to reduce the processing time. To detect vehicles, the Background subtraction method is used, followed by the connected components method to improve the results of the detection. Optical flow [12] is a common technique to aggregate the temporal information for vehicle detection by simulating the pattern of object motion over time. Optical flow is also combined with symmetry tracking [13] and hand-crafted appearance features [14] for better performance. However, this kind of approach is unable to distinguish the fine-grained categories of the moving objects such as car, bus, van or person. In addition, these methods need lots of complex post-processing algorithms like shadow detection and occluded vehicle recognition to refine the detection results. Statistical learning-based methods are based on the handcrafted features to detect the vehicles from the images directly. These methods first describe the regions of the image by some feature descriptors and then classify the image regions into different classes such as vehicle and non-vehicle. Features like HOG [15], [16], SURF [17], Gabor [16] and Haar-like [18], [19] are commonly used for vehicle detection followed by classifiers like SVM [15], [17], artificial neural network [16] and Adaboost [18], [19]. These features, however, have limited ability of feature representation, which is difficult to handle complex scenarios.

Recently, deep CNN-based methods have become the leading method for high quality general object detection [28]. Faster region-based convolutional neural network (Faster R-CNN) [6] defined a region proposal network (RPN) for generating region proposals and a network using these proposals to detect objects. RPN shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. This method has achieved state-of-the-art detection performance and become a commonly employed paradigm for general object detection. SSD framework [7] predicted category scores and box offsets for a fixed set of default bounding boxes using small convolutional filters applied to different scales from feature maps of different scales, and explicitly separate predictions by aspect ratio. This framework showed much faster and comparably performance with other methods. Most of these deep learning models target general object detection including vehicle. To better handle the detection problem of vehicles in complex conditions, a

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

second-layer conditional random field (CRF) was used over root and part score configurations provided by a DPM model in [21]. Recently, an AND-OR structure was proposed in [22] and [23] to model occlusion configurations effectively compared against the classical DPM. In [22], the authors proposed to model object occlusion with an AND-OR structure which represents occlusion at semantic part level and captures the regularities of different occlusion configurations. The model parameters are learned from real images under the latent structural SVM framework. Furthermore, an efficient dynamic programming algorithm is utilized in inference time. In [23], a method for learning And-Or models to represent context and occlusion for car detection and viewpoint estimation was presented. The structure of the And-Or model is learned with three components: mining multi-car contextual patterns based on layouts of annotated single car bounding boxes, mining occlusion configurations between single cars, and learning different combinations of part visibility based on car 3D CAD simulation. Furthermore, the model parameters are jointly trained using Weak-Label Structural SVM. Chu et al. [20] proposed a vehicle detection scheme based on multi-task deep CNN which learning is trained on four tasks: category classification, bounding box regression, overlap prediction and subcategory classification. A region-of-interest voting scheme and multi-level localization are then used to further improve detection accuracy and reliability. Experimental results on standard test dataset showed better performance than other methods. Yi Zhou et al. [24] presented a fast framework of Detection and Annotation for Vehicles (DAVE), which effectively combines vehicle detection and attributes annotation. DAVE consists of two convolutional neural networks: a fast vehicle proposal network for vehicle-like objects extraction and an attribute learning network aiming to verify each proposal and infer each vehicle's pose, color and type simultaneously. In [25], strategies for occlusion and orientation handling are explored by learning an ensemble of detection models from visual and geometrical clusters of object instances. An AdaBoost detection scheme is employed with pixel lookup features for fast detection.

3. APPROACH

Figure 1 shows the overall framework of the proposed approach. First, each image is forwarded through the base network to generate feature maps. Then, the enhanced module including convolution layers, pooling layers, and deconvolution layers are added on top of the base network to provide richer context for vehicle detection at individual feature output scale. Four region proposal modules are adopted in the region proposal networks to produce high quality region proposals. Finally, a deep CNN-based classifier including a region of interest (ROI) pooling layer and a fully connected (FC) layer is used for classification and bounding box regression. Details of the proposed approach are explained in next sub-sections.

3.1 The Base Network

The base network is based on the popular VGG-16 architecture [4], which has 16 weight layers in its original form. VGG-16 is a simpler architecture model, since it is not using much hyper parameters. It always uses 3 x 3 filters with stride of 1 in convolution layer and uses same padding in pooling layers 2 x 2 with stride of 2. Additional convolution layer and pooling layer are added on top of the VGG-16 net. The architecture of the base network is shown in Figure 2. The feature outputs of convolution layers and pooling layer in the base network are directly used for generating object proposal. The layers selected as feature output layers include conv4-3, conv5-3, conv6-1 and pool6. The first number in the labels such as 4 and 6 represents the associated hidden layer in VGG-16 architecture, and the second number represents the ID of the convolution layer in a hidden layer.

3.2 Enhanced Module

Current deep CNN-based object detectors exploit multi-scale features to produce predictions of different scales, which showed improved object detection performance over Faster-CNN and SSD. However, shallow feature maps from the low layers of feature pyramid inherently lack fine semantic information for object recognition. Thus, this paper adds three deconvolutional modules at the end of the base network. With this enhanced module, the semantics from higher layers can be conveyed into lower layers to increase the representation capacity. Figure 3 illustrates the architecture of each deconvolutional module. As shown, a 1 x 1 convolution layer and rectified linear activation are used. For the deconvolution branch, the encoder-decoder structure with 4 x 4 deconvolution is used followed by a 1 x 1 convolution. A batch normalization layer (BN) is added after each convolution layer. Higher-level feature maps are extracted after pool6 layer, conv4-3 layer and conv5-3 layer. Then, the deconvolution layer is added to enlarge the feature map size in order to match the

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

size of the lower-level feature maps. Finally, element-wise product is performed as a combination method, which is followed by rectified linear activation to generate the new output feature layer.

3.3 Region Proposal Networks

The region proposal network (RPN) receives feature maps output from the deconvolutional modules and pool6 layer to produce high quality region proposals. In this paper, four region proposal modules which have the same architecture, but different parameters are used in the region proposal networks. The architecture of each region proposal module is shown in Figure 4. Firstly, the region proposal module takes feature maps generated after deconvolutional modules and pool6 layer to generate a set of anchor boxes. An anchor is centered at the sliding window and is associated with a scale and aspect ratio. Because vehicle is usually in rectangular or square shape, this paper uses one scale and two aspect ratios for each anchor, yielding 2 anchors at each sliding position of each region proposal module as shown in Figure 5. More specific, the aspect ratios are set at 1 and 0.5 in this paper. Next, each region proposal module takes all the anchor boxes and outputs two different outputs for each of the anchors. The first one is objectness score, which means the probability that an anchor is an object. The second output is the bounding box regression for adjusting the anchors to better fit the object. The anchors with estimated classification scores and the bounding box for each feature map location then are processed to form good quality proposals. Since anchors usually overlap, proposals end up also overlapping over the same object, soft Non-Maximum Suppression (NMS) [5] is used to solve the issue of duplicate proposals. Due to heavy object occlusion in KITTI dataset, NMS may remove positive proposals unexpectedly. Thus, with soft-NMS, the neighbor proposals of a winning proposal are not completely suppressed. The proposal whose region overlaps a ground truth region more than 70% is regarded as a positive proposal. Otherwise, it is regarded as a negative proposal. After applying soft-NMS, this paper keeps the top 256 proposals sorted by score.

3.4 Deep CNN-based Classifier

The deep CNN-based classifier is the final stage in the proposed framework. The deep CNN-based classifier has a region of interest (ROI) pooling layer and a fully connected (FC) layer. The RoI pooling layer takes a section of the input feature map that corresponds to positive proposals proposed by the RPN as input. Then, the ROI pooling layer extracts the feature maps of the object proposals using these inputs as shown in Figure 6. After applying ROI pooling layer, a list of regions with different sizes is transformed into a list of corresponding feature maps with a fixed size. Fixed size feature maps are needed for the classifier in order to classify them into a fixed number of classes. The classifier has two different goals: Classify proposals into vehicle and background class and adjust the bounding box for each of detected vehicle. The proposed classifier has two fully connected (FC) layers, a box classification layer and a box regression layer. The first FC layer has two outputs, which are fed into the softmax layer to compute the confidence probabilities of being vehicle and background. The second FC layer with linear activation functions regresses the bounding boxes of vehicle. All convolutional layers are followed by a batch normalization layer and a ReLU layer.

4. EXPERIMENTAL RESULTS

In order to compare the effectiveness of the proposed method with other state-of-the-art methods, this paper conducts experiments on the widely used public dataset: the KITTI dataset [1]. The proposed approach is implemented on a Window system machine with Core i5 6400 processor, NVIDIA GTX 1050Ti gpu and 8 Gb of RAM. TensorFlow is adopted for implementing deep CNN frameworks, and OPENCV library is used for real time processing.

4.1 Dataset for Evaluating

KITTI dataset [1] is a widely used dataset for evaluating vehicle detection algorithms. 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 vehicles 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 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 was considered as a moderate object, and a vehicle with the bounding boxes size smaller than 25 pixels and an invisible vehicle that was difficult to see with the naked eye were considered as hard objects. Figure 7 shows example images in these groups. Since the ground truth of the KITTI test set are not publicly available, this paper splits the KITTI training images into a train set and a test set to conduct experiments as in [2], which

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

results in 3682 images for training and 3799 images for testing.

4.2 Evaluation Metrics

This paper uses the average precision (AP) and intersection over union (IoU) metrics [3] 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 [1], [3]. The IoU is defined as the following equation:

$$IoU = \frac{area\,(B\_B_{gt} \cap B\_B_{dt})}{area(B\_B_{gt} \cup B\_B_{dt})} \quad (1)$$

where B_B_{gt} and B_B_{dt} represent the area of ground truth bounding box and detected bounding box respectively. Higher value of IoU mean better quality of the detection. The IoU is set to 0.7 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% is considered as a correct detection.

4.3 Detection Results

Recently, deep CNN-based object detection methods such as Faster R-CNN, SSD and MS-CNN [27] have become dominating. Faster R-CNN framework contains two stages: region proposal generating and object detection network. While Fast R-CNN algorithm [26] is based on the selective search algorithm, the Faster R-CNN introduces the Region Proposal Network, which has improved over the traditional methods. The Single Shot MultiBox Detector (SSD) framework combines region proposals and region classifications in a 'single shot'. The core of SSD is predicting category scores and box offsets for a fixed set of default bounding boxes using small convolutional filters applied to different scales from feature maps of different scales, and explicitly separate predictions by aspect ratio. MS-CNN extends the detection over multiple scales of feature layers, which produces good detection performance improvement.

Figure 8 shows examples of detection results of the proposed method on the KITTI test dataset. As shown from this figure, the proposed approach can handle the situation of occlusion effectively. Furthermore, the proposed method can detect small vehicle and avoid producing multiple bounding boxes for one vehicle. Figure 9 presents example images in which some vehicles are not correctly detected. The main challenges of the vehicle detection in KITTI dataset come from the small size, heavy occlusion or truncation of the vehicles. Moreover, other external factors such as illumination change and cluttered background can affect the accuracy of the proposed method.

Table 1 shows the detection results on the KITTI test dataset of the proposed method and other state-of-the-art deep CNN-based object detectors. As shown from Table 1, the performance of the proposed method is improved comparing with the Faster R-CNN framework by 2.72%, 10.21%, 1.43% in 'easy', 'moderate', and 'hard' groups respectively. Furthermore, comparing with the SSD framework, the proposed algorithm improves by 6.73%, 22.15%, 21.53% in 'easy', 'moderate', and 'hard' groups respectively. Comparing with the MS-CNN, the proposed algorithm improves by 0.16%, 0.49%, 5.86% in 'easy', 'moderate', and 'hard' groups respectively. For the computational efficiency, the proposed method takes 0.41 second for processing an image with a low-end hardware machine. Thus, the proposed approach meets the real-time detection standard and can be applied to the road driving environment of actual vehicles.

5. CONCLUSIONS

Vehicle detection plays an important role in an autonomous driving system. Recently, deep CNN-based object detectors achieved a big improvement compared to traditional methods on vehicle detection. However, due to the challenging driving environment, popular CNN detectors such as Faster R-CNN, SSD and MS-CNN do not produce good detection performance over the KITTI driving benchmark dataset. This paper proposes an improved deep CNN-based framework for vehicle detection. To improve the detection accuracy, three deconvolutional modules are added at multiple output layers of the reduced VGG 16 architecture to enhance additional context information. Four region proposal modules are then applied in the region proposal networks to produce high quality region proposals. Furthermore, soft-NMS is used in the region proposal networks to solve the issue of duplicate proposals. The proposed method is evaluated with the KITTI dataset. Compare with other state-of-the-art detectors, the proposed approach showed better experimental results. In the future, this paper will investigate more CNN models and enhancements to improve vehicle detection.

REFERENCES:

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI

vision benchmark suite", *Proc. CVPR*, Jun. 2012, pp. 3354–3361.

[2] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection", *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 924–933.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge", *Int. J. Comput. Vis.*, vol. 88, no. 2, Sep. 2009, pp. 303–338.

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *NIPS*, 2015.

[5] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Improving object detection with one line of code", arXiv preprint arXiv:1704.04503, 2017.

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks", *Advances in neural information processing systems*, 2015, pp. 91–99.

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector", *European conference on computer vision*, 2016, pp. 21–37, Springer.

[8] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking", *Proc. CVPR*, 1999, p. 252.

[9] Z. Chen, T. Ellis, and S. A. Velastin, "Vehicle detection, tracking and classification in urban traffic", *Proc. ITSC*, Sep. 2012, pp. 951–956.

[10] C. Premebida and U. Nunes, "A multi-target tracking and GMMclassifier for intelligent vehicles", *Proc. ITSC*, Sep. 2006, pp. 313–318.

[11] M. Vargas, J. M. Milla, S. L. Toral, and F. Barrero, "An enhanced background estimation algorithm for vehicle detection in urban traffic scenes", *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, Oct. 2010, pp. 3694–3709.

[12] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using optical sensors: A review", *Proc. ITSC*, Oct. 2004, pp. 585–590.

[13] S. Kyo, T. Koga, K. Sakurai, and S. Okazaki, "A robust vehicle detecting and tracking system for wet weather conditions using the IMAP-VISION image processing board", *Proc. ITSC*, Oct. 1999, pp. 423 428.

[14] J. Cui, F. Liu, Z. Li, and Z. Jia, "Vehicle localisation using a single camera", *Proc. IEEE Intell. Vehicles Symp*., Jun. 2010, pp. 871–876.

[15] Q. Yuan, A. Thangali, V. Ablavsky, and S. Sclaroff, "Learning a family of detectors via multiplicative kernels", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, Mar. 2011, pp. 514–530.

[16] R. M. Z. Sun and G. Bebis, "Monocular precrash vehicle detection: Features and classifiers", *IEEE Trans. Image Process.*, vol. 15, no. 7, Sep. 2006, pp. 2019–2034.

[17] J.-W. Hsieh, L.-C. Chen, and D.-Y. Chen, "Symmetrical SURF and its applications to vehicle detection and vehicle make and model recognition", *IEEE Trans. Intell. Transp. Syst.,* vol. 15, no. 1, Feb. 2014, pp. 6–20.

[18] W. C. Chang and C. W. Cho, "Online boosting for vehicle detection", *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, Jun. 2010, pp. 892–902.

[19] S. Sivaraman and M. M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking", *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, Jun. 2010, pp. 267–276.

[20] Chu W., Liu Y., Shen C. et al., "Multi-Task Vehicle Detection With Region-of Interest Voting", *IEEE Transactions on Image Processing*, 2018, pp. 432-441.

[21] H. T. Niknejad, T. Kawano, Y. Oishi, and S. Mita, "Occlusion handling using discriminative model of trained part templates and conditional random field", *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2013, pp. 750–755.

[22] B. Li, W. Hu, T. Wu, and S.-C. Zhu, "Modeling occlusion by discriminative and-or structures", *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2560–2567.

[23] T.Wu, B. Li, and S.-C. Zhu, "Learning and-or model to represent context and occlusion for car detection and viewpoint estimation", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, Sep. 2016, pp. 1829–1843.

[24] Y. Zhou, L. Liu, L. Shao, and M. Mellor, "DAVE: A unified framework for fast vehicle detection and annotation", *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 278–293.

[25] E. Ohn-Bar and M. M. Trivedi, "Learning to detect vehicles by clustering appearance patterns", *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, Oct. 2015, pp. 2511–2521.

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

[26] R. Girshick, "Fast R-CNN", *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[27] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection", *European Conference on Computer Vision*, 2016, pp. 354–370.

[28] Hanafi, Nanna Suryana, Abd Samad Bin Hasan Basarideep, "Learning for recommender system based on application domain classification perspective: a review", *Journal of Theoretical and Applied Information Technology*, Vol. 96, No. 14, 2018, pp. 4513-4529.

[29] Slimani Ibtissam, Zaarane Abdelmoghit, Hamdoun A., Issam Atouf, "Traffic surveillance system for vehicle detection using discrete wavelet transform", *Journal of Theoretical and Applied Information Technology*, Vol. 96, 2018, pp. 5905-5917.

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

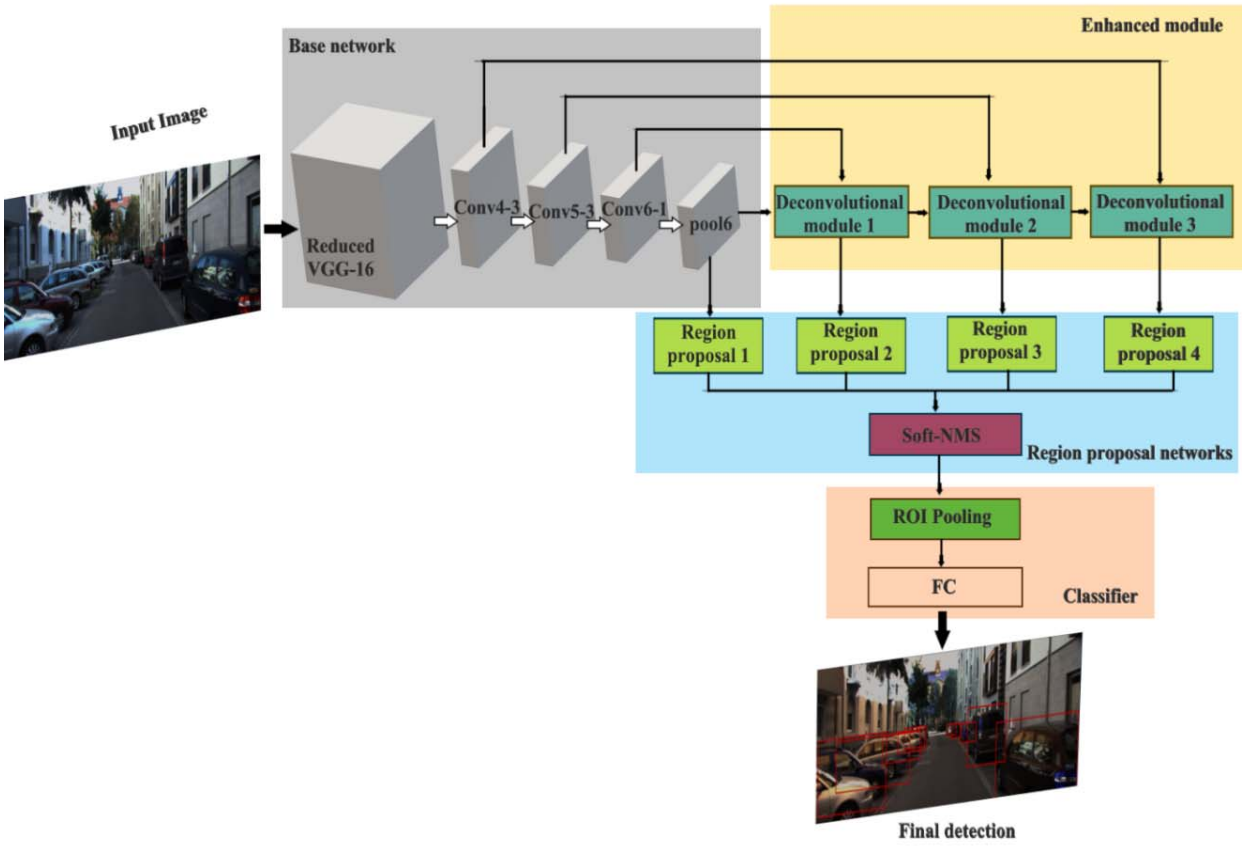

Figure 1: The Overall Framework of The Proposed Approach.

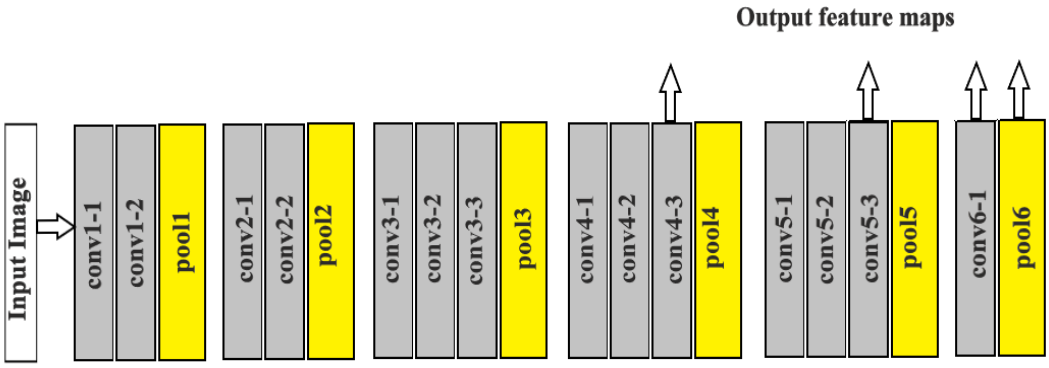

Figure 2: The Architecture of The Base Network.

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

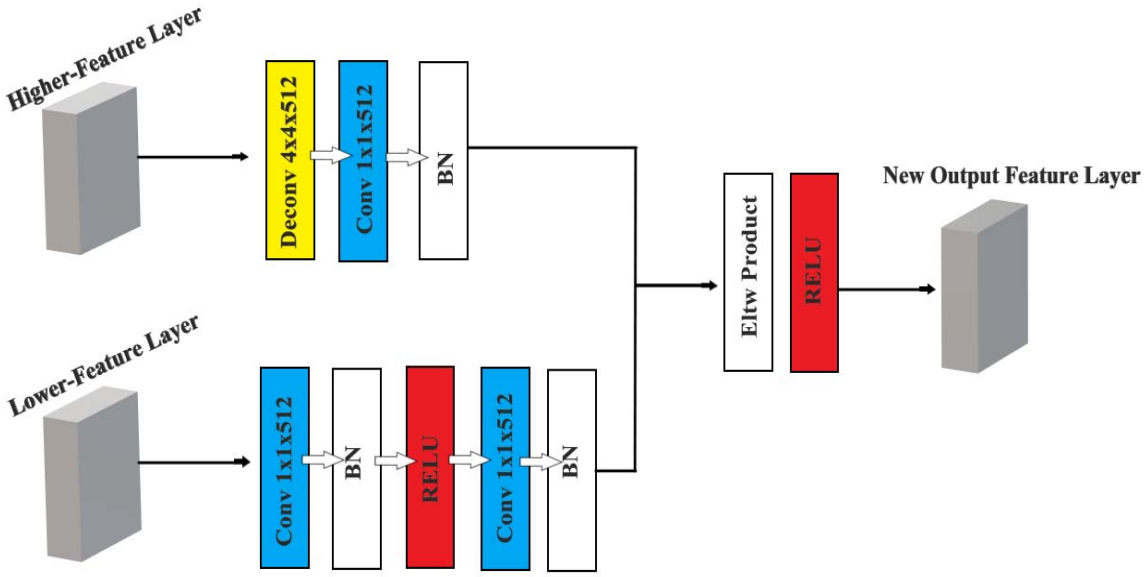

Figure 3: The Architecture of Deconvolutional Module. The Enhanced Module Includes Three Deconvolutional Modules.

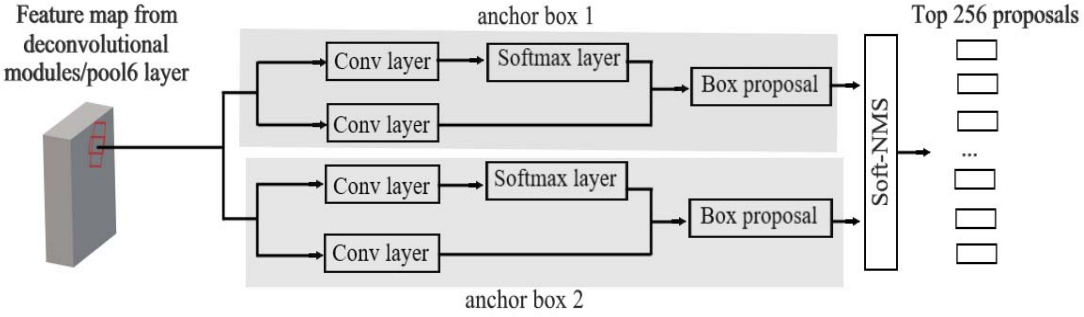

Figure 4: The Architecture of The Region Proposal Module. The Region Proposal Network Includes Four Region Proposal Modules.

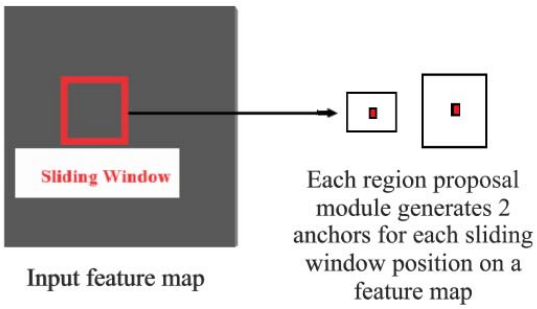

Figure 5: Anchor Boxes Generated by Region Proposal Module. In This Paper, Each Position of The Sliding Window Generates 2 Anchor Boxes with Different Aspect Ratios.

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

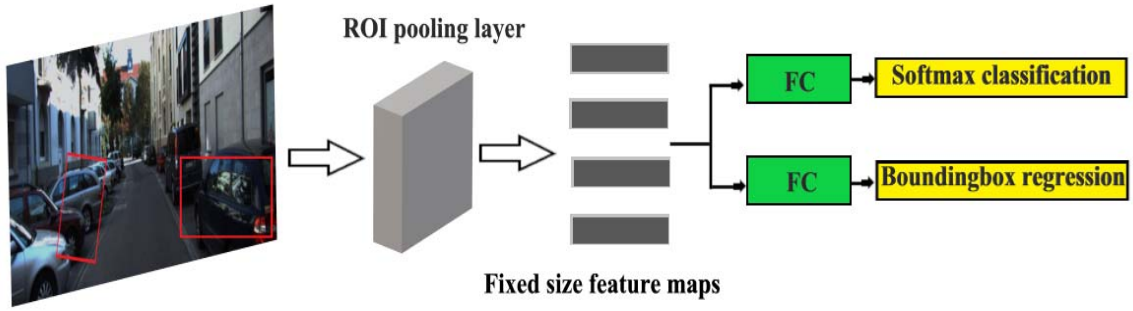

Figure 6: Framework of The Classifier.

Figure 7: Example Images in KITTI Dataset. The Images Contain Vehicle in Different Scale, Perspectives, Backgrounds.

Journal of Theoretical and Applied Information Technology
15th December 2019. Vol.97. No 23

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

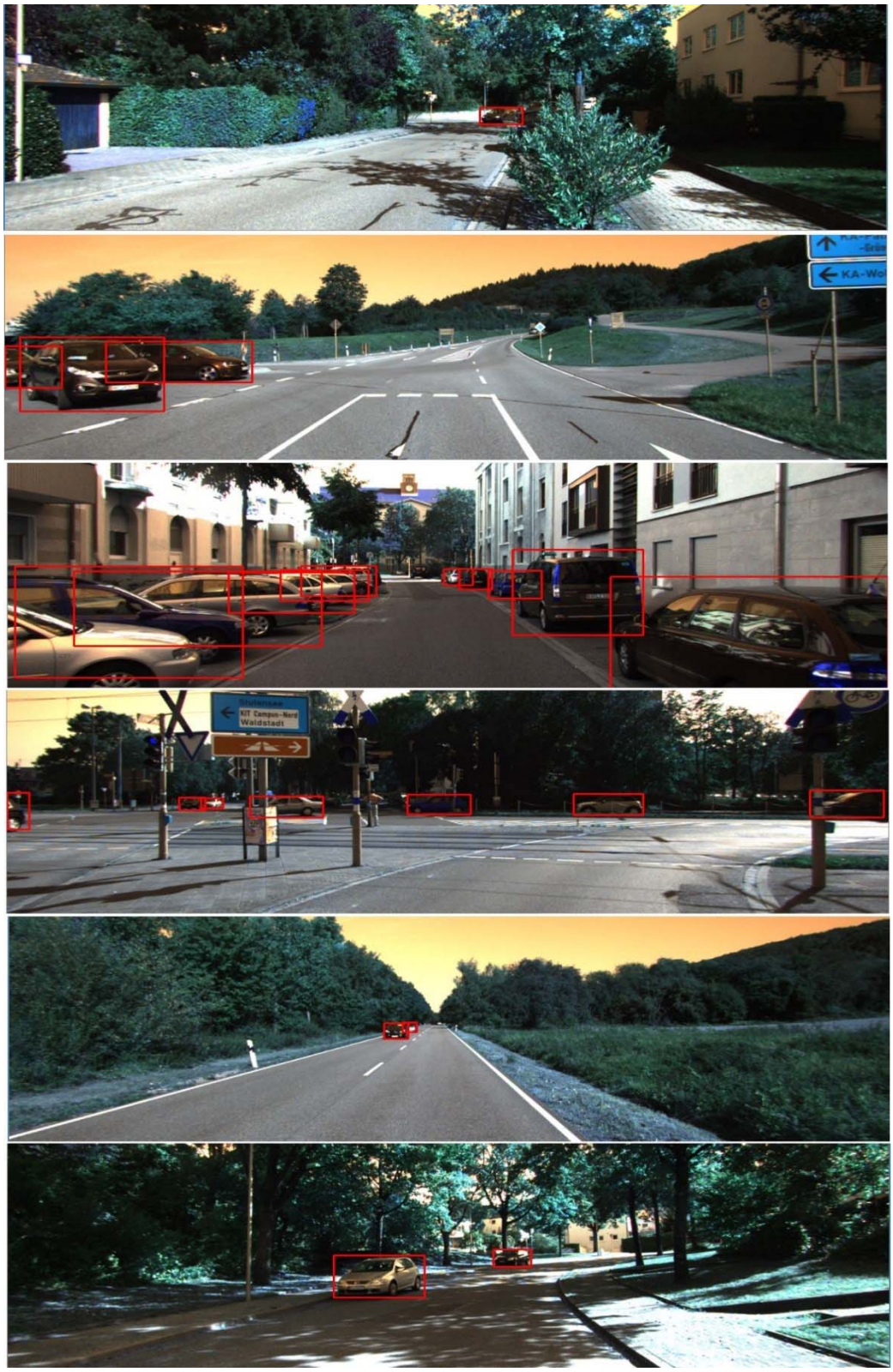

Figure 8: Examples of Detection Results of The Proposed Method on The KITTI Test Dataset.

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

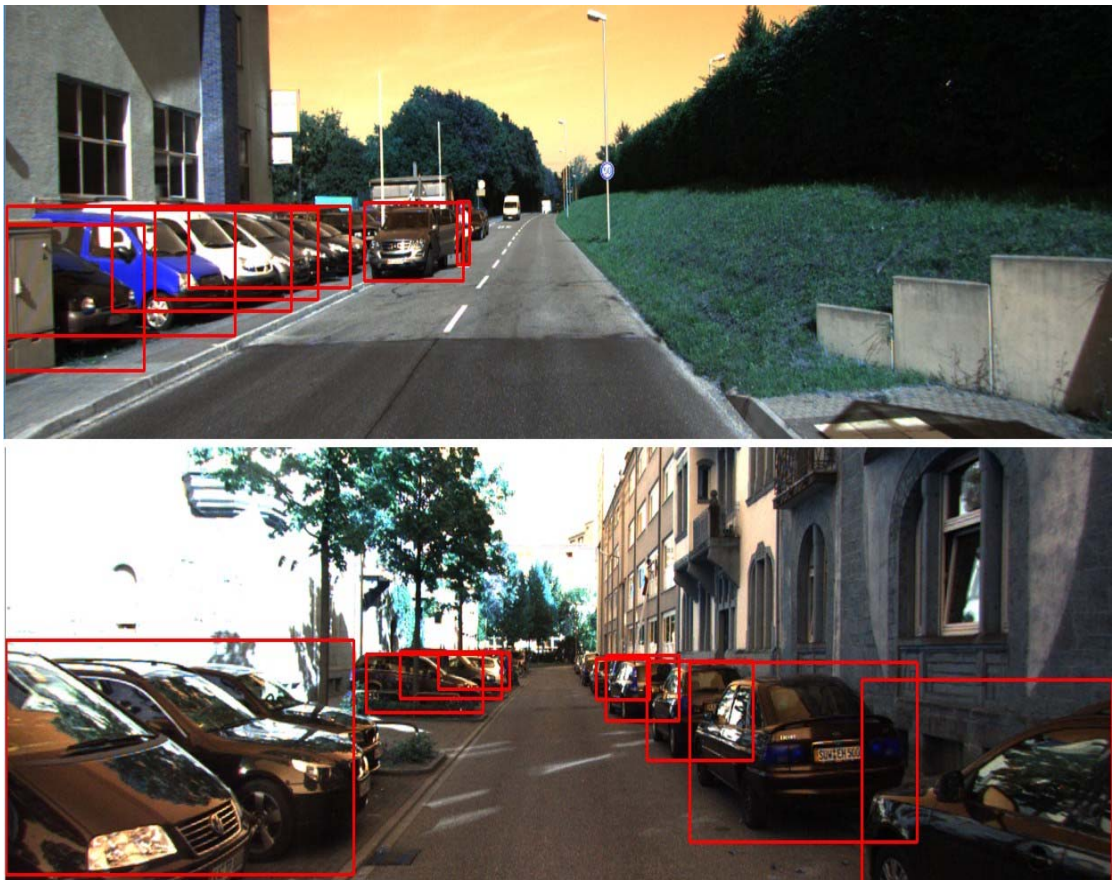

Figure 9: Undetected Vehicle Due to Small Size, Heavy Occlusion or Truncation of The Vehicles.

Table 1: Detection Results of The Proposed Method and Other Methods.

Method	Difficulty-level groups			Processing time (s)
	Easy (%)	Moderate (%)	Hard (%)	
Faster R-CNN [6]	87.90	79.11	79.19	2
SSD [7]	83.89	67.17	59.09	0.06
MS-CNN [27]	90.46	88.83	74.76	0.4
Proposed method	90.62	89.32	80.62	0.41 (low-end machine)