2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

An Efficient Hierarchical Convolutional Neural Network for Traffic Object Detection

Qianqian Bi, *Student Member, IEEE,* Ming Yang*, Chunxiang Wang and Bing Wang

***Abstract*— In this paper, we propose a novel hierarchical convolutional neural network for traffic object detection, which is defined as Fusion and Multi-level Alignment CNN (namely FMLA-CNN). The method extends a popular two-stage detector by incorporating a remodified feature fusion module and a multi-level alignment (MLA) strategy such that it is capable of efficiently detecting multi-scale objects in autonomous driving scenario. The feature fusion strategy in proposal generation network improves detection accuracy by inserting high-level semantics to the whole pyramidal feature hierarchy. Subsequently the MLA strategy in the second detection stage can exactly reserve spatial locations from corresponding feature layers determined by hierarchical region-of-interest proposals. In the experiments on KITTI benchmark, our FMLA-CNN achieves an impressively better trade-off between accuracy and efficiency compared with other state-of-the-art methods.**

I. Introduction

Object detection plays an increasingly important role in autonomous driving scenario, which can locate vehicles and pedestrians accurately and synchronously and thus ensure driving security. In the past five years, the performance of object detection has been significantly improved with the successful application of impressive deep convolutional neural networks (ConvNets) [1]. However, the large scale gap between vehicles due to distant view (shown in Fig.1) and the different size between vehicles and pedestrians can both lead to scale variation problem, which is a critical challenge for ConvNet object detectors.

The conventional ConvNet object detectors can be roughly classified into two-stage detectors and single-stage detectors. The two-stage detector first generates region-of-interest proposals (RoIs) by body proposal networks and then refined recognition on RoIs is completed by head detection networks. As presented in Fig.2(a), instead of multi-scale image input, the popular detectors [2]–[4] utilize one high-rise feature layer with anchor box strategy to detect multi-scale objects by a single image input. Take Faster R-CNN for example, in the first stage the region proposal network (RPN) creates anchor boxes with different aspect ratios and scales from a single feature map. Then another head detection network [5] would classify targets and locate bounding box on these candidate proposals. Although RPN mechanism is powerful, it still imposes a great burden for a single receptive field to match large scale variability.

This work was supported by the National Natural Science Foundation of China (U1764264). Ming Yang is the corresponding author.

Qianqian Bi, Ming Yang, Chunxiang Wang and Bing Wang are with the Department of Automation, Shanghai Jiao Tong University, Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240,CN (phone: +86-21-34204533; email: MingYANG@sjtu.edu.cn).

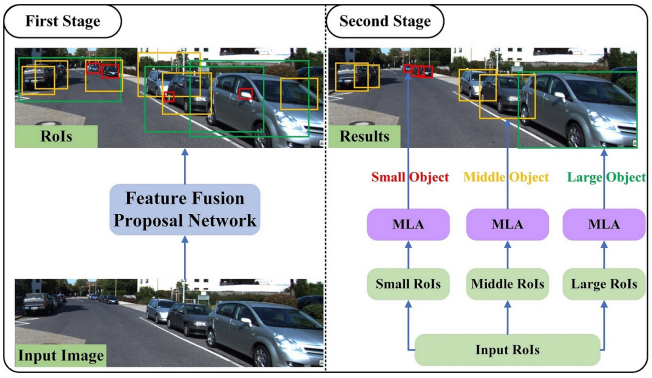

Fig. 1. The efficient two-stage **FMLA-CNN** framework for multi-scale object detection. Scale variation problem is illustrated in the **Input Image**. **RoIs** means region-of-interest proposals which are generated in the first stage. **MLA** means single multi-level alignment branch.

Within a ConvNet, shallow layers with small receptive fields and high resolution focus on small object location while deep layers with large receptive fields are comparatively suitable for large object prediction. Thus, the detectors selecting features computed from hierarchical receptive fields always perform better. For example, as illustrated in Fig.2(b), SSD [6] and MS-CNN [7] adopt pyramidal feature hierarchy without layer integration. Nevertheless, in these non-fusion methods, the shallow layers aimed for small object location always lack enough semantic context, resulting in poor performance on small object detection.

The recent powerful hierarchical feature fusion structures, like FSSD [8] in Fig.2(d) and FPN [9] in Fig.2(e), merge feature maps with various receptive fields while implementing multiple predictors. Specially, FPN [9] introduces pyramidal structure with top-down pathway, which in turn contributes to constructing high-level semantic feature maps of all scales. Although these fusion structures have already achieved good performance on scenes with high demand in classification like COCO benchmark [10], they are not effective on scenes with high location accuracy request but relatively simple classification task such as KITTI benchmark [11] which predicts car location for high overlap threshold. In addition, conventional versions of FPN all select features from the backbone of ResNet, however, substantial experimental results on KITTI benchmark indicate that frameworks based on the backbone of VGG16 could achieve state-of-the-art detection performance considering the relatively simple classification task and real-time requirement in driving scenario. Therefore, in this paper, we would first exploit how to efficiently fuse feature layers on the variant of VGG16.

For the two-stage detectors, there is another line of study which tends to handle the scale variation problem by multiple scale dependent classifiers during the second stage. Based on RoIPool operation denoted in [5], SDP [12] advocates several scale-dependent RoIPool modules to respectively process RoIs of different scales while these candidate proposals are still computed from a single receptive field in the first stage. When it comes to the combination methods of using FPN in a basic Faster R-CNN system, RoIs in different scales are always mapped into the pyramid levels to pool features. Nevertheless, these frameworks are not computationally efficient due to some redundant procedures and can not meet real-time detection requirements in autonomous driving domain.

To overcome these aforementioned drawbacks, this paper exploits two different routes to tackle scale variation problem, namely feature fusion and multi-level alignment. By incorporating the two strategies into a Faster R-CNN style baseline, a novel hierarchical deep learning framework is presented for efficient and accurate object detection in autonomous driving domain. As illustrated in Fig.1, the proposed FMLA-CNN framework consists of two modules: feature fusion proposal body network and refined detection head network. In the first module, we remodify two popular feature fusion methods and then choose the top-down pathway as best fusion version applied in our designed baseline. In the second module, we propose a multi-level alignment (MLA) strategy to locate multi-scale objects exactly. In our MLA strategy, all RoI proposals are categorized into hierarchical levels and then respectively allocated to corresponding feature layers to construct RoI aligning operation. In our experiments, the independent and combined effectiveness of these two strategies for multi-scale object detection are both verified on KITTI benchmark. Therefore, the main contributions of this paper are as follows:

- Some key techniques are explored to optimize the final fusion module when designing the proposal body network.
- The multi-level alignment method is proposed for the first time. It is very effective to preserve exact spatial locations from hierarchical features and address scale variation problem in object detection.
- Several adjustments for efficiency optimization are proposed during the combination of the two strategies.
- Our FMLA-CNN overtakes several other state-of-the-art object detection models both in accuracy and efficiency on KITTI benchmark.

II. FMLA-CNN Framework

Our FMLA-CNN framework is composed of two modules: feature fusion proposal body network and refined detection head network. In the following part, we would respectively develop two different multi-scale detection strategies and exploit how to efficiently combine feature fusion model in the body network and MLA strategy in the head network.

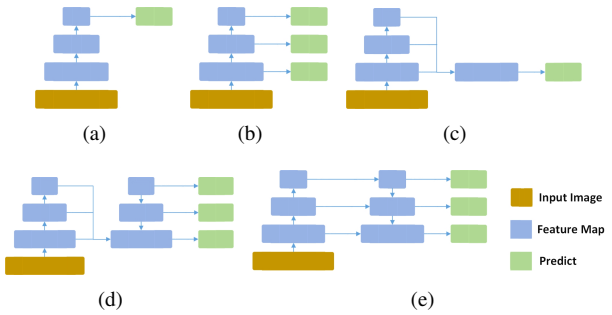

Fig. 2. The first kind of strategies for multi-scale object detection applied in single-stage detectors or proposal generation network in two-stage detectors: **(a) single RPN feature; (b) pyramidal feature hierarchy; (c) lightweight feature fusion; (d) fusion1: combination of b and c; (e) fusion2: top-down pyramidal fusion.**

A. Feature Fusion Proposal Network

In this subsection, we mainly explore two feature fusion methods on VGG16 variant. When we design the feature fusion proposal network, several factors should be taken into consideration such as context or semantics mergence, aliasing effect alleviation and redundancy reduction. We would investigate these key points in the following two fusion models. Furthermore, the performance of two fusion models will be compared with baseline1 model like Fig.2(a) and baseline2 model like Fig.2(b).

Baseline1 and Baseline2. Our backbone network is a VGG16 variant in which we discard the original architectures after *pool5* and attach *pool5* with *conv6* and *pool6*. In our baseline1 model, *conv5_3* feature map is chosen as standard RPN layer with nine kinds of anchor boxes to generate proposals. On the other hand, our baseline2 model is also built on the same VGG16 variant backbone but leverage on *conv4_3*, *conv5_3*, *conv6_2* and *pool6* as pyramidal feature hierarchy to produce proposals. The reason why we do not choose *conv3_3* is that coarse feature maps would aggressively hurt the detection performance. Because the baseline2 model already has four hierarchical feature maps to cope with scale variation challenge, we only apply two scales of anchor boxes to each proposal generation branch to reduce redundant computation. Specially, the fusion methods in the following part all adopt the efficient structure with four proposal generation branches and two-scale anchor box strategy for each branch.

Fusion1: Lightweight feature fusion on baseline2. As illustrated in Fig.2(d), the first feature fusion model is introduced in a one-stage detector FSSD [8] to implement a lightweight feature fusion by concatenation and then generate pyramidal feature hierarchy which is used to predict object only from the new fused feature. We remodify that architecture based on our baseline2 model and make it available for proposal generation in a two-stage detector. Specifically, we upsample *conv5_3* by a factor of 2 and *conv6_2* by a factor of 4 with deconvolution operation. In the next step, *conv4_3*, *conv5_3_2x* and *conv6_2_4x* are all appended by a 1×1 convolutional layer and an L2 normalization layer

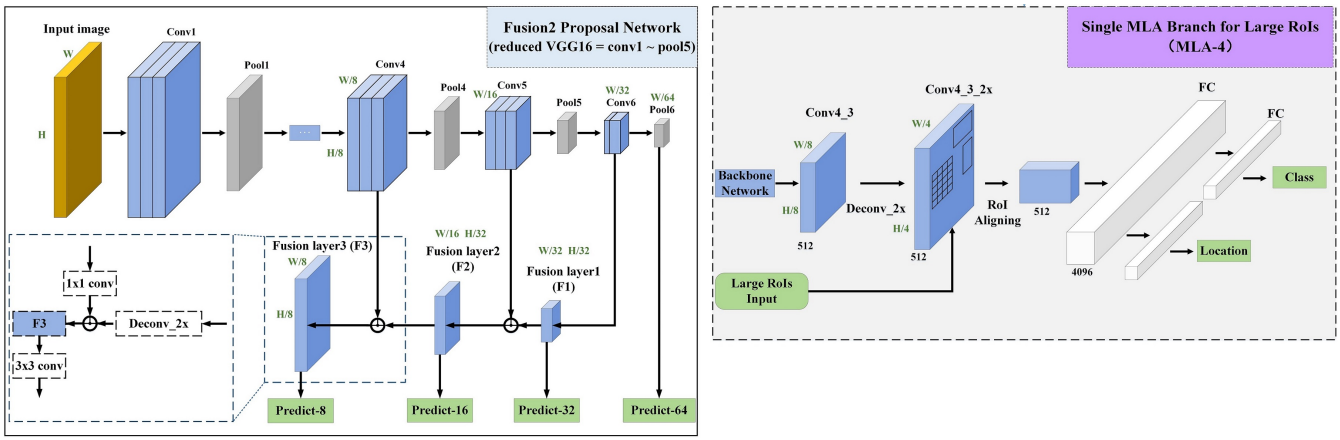

Fig. 3. The left architecture is our feature fusion proposal network in FMLA-CNN which finally adopts fusion2 model. The right structure is one MLA branch subnetwork for large RoIs. Backbone layers are shared with the left proposal network. W and H refer to the width and height of the input image.

[13] before they are catenated together. The function of 1×1 convolutional layer is to decrease channel dimensions to 256. Concatenating features without normalizing individual layers would cause inferior performance since the feature maps with high resolution in the shallow layers always dominate the ones with low resolution in the deeper layers. Finally, three groups of *conv-pool* layer are successively added and the four new feature maps respectively have the identical spatial sizes to the corresponding original pyramidal features in baseline2.

Fusion2: Top-down pathway fusion on baseline2. Different from previous four-level Feature Pyramid Network (FPN) based on ResNet, we redesign a top-down pathway fusion on baseline2. The specific structure is shown in the left graph in Fig.3 and our modification is using VGG16 variant in place of ResNet. This modification is suitable for driving scenes with simple classification task and real-time requirement. Although the baseline2 model exists four branches for proposal prediction, our final version of fusion2 model (which is applied in the FMLA-CNN) only merge three feature layers in the top-down pathway. To begin with, the upsampled feature (*conv6_2_2x*) is fused with its corresponding bottom feature map (*conv5_3*) by element-wise addition. While there is no need to reduce the channels of the bottom-up maps {*conv4_3*, *conv5_3*, *conv6_2*} since they have the same channels, it is still necessary to append 1×1 convolution to the bottom-up maps as lateral connection. This process is repeated until the highest resolution map is produced. It is also crucial to attach 3×3 convolution on each fused map to generate the new feature map for prediction. 1×1 and 3×3 convolution can both benefit on alleviating the aliasing effect generated by deconvolution. Note that the first and coarsest layer (*conv6_2*) is also connected with 1×1 convolution and then predict proposals. Thus, the final set of pyramidal feature hierarchy is composed of {F1, F2, F3, pool6} as illustrated in Fig.3.

Training for feature fusion proposal network. As hard negative mining strategy can ensure fast optimization and stable training, we adopt that method to balance the distribution of the positive and negative training examples (respectively notated in N^+ and N^-). The strategy is collecting top N^- negative anchor boxes to make $|N^-| = \gamma|N^+|$ on each feature layer. Since negatives are far more than positives in each branch, we add the cross-entropy terms of the positive anchors and negative anchors to the classification loss. Referring to the classical multi-task loss function [5], an objective function for an image in each fused branch is defined by us as follows

$$L\left(p,u,b,\hat{b}\right) = L_{cls}\left(p,u\right) + \lambda\left[u \geq 1\right] L_{loc}\left(b,\hat{b}\right) \tag{1}$$

where $L\left(p,u,b,\hat{b}\right)$ is used to jointly learn for classification and bounding box regression. p, b represent the predicted class probability and predicted location coordinates respectively, whereas u, $\hat{b}$ are the ground-truth label for class and location respectively.

$$\begin{aligned} L_{cls}\left(p,u\right) = & \frac{1}{1+\gamma}\frac{1}{\left|N_{cls}^{+}\right|}\sum_{i\in N_{cls}^{+}} -\log\left(p_i^j\right) \\ & + \frac{\gamma}{1+\gamma}\frac{1}{\left|N_{cls}^{-}\right|}\sum_{i\in N_{cls}^{-}} -\log\left(p_i^0\right) \end{aligned} \tag{2}$$

where L_{cls} is a softmax loss over class u with cross-entropy terms. p_i^j is a predicted probability for matching the i-th anchor box to the j-th ground truth box, whereas p_i^0 means probability for the i-th anchor box being negative.

$$L_{loc}\left(b,\hat{b}\right) = \frac{\sum_{i\in N_{loc}^{+}}\sum_{k\in\{x,y,w,h\}} smooth_{L1}\left(b_i^k - \hat{b}_i^k\right)}{4\left|N_{loc}^{+}\right|} \tag{3}$$

where L_{loc} is a $smooth_{L1}$ loss. b_i^k and $\hat{b}_i^k$ means k parameterized coordinates of the i-th predicted box and the i-th ground truth box respectively.

The feature fusion proposal network is initialized with a reduced VGG16 which is pre-trained on the ImageNet dataset. Then gradients are back-propagated from four branches to update corresponding convolutional filters for

prediction during fine-tuning. Different from [2], our first-stage model can be regard as an independent detector.

B. Refined Detection Network

In this subsection, similarly for scale variation challenge, we would investigate another line work on the object detection network which intends to leverage on RoI proposals. It is noted that the detection network of our two baselines is a standard Fast R-CNN architecture with RoI pooling.

RoI Aligning. When RoIPool [5] extracts fixed features (e.g., 7×7) for each RoI, misalignments between the original RoIs and the final fixed RoI features are caused by quantization in two steps. Quantization is firstly conducted when dividing each continuous coordinate by the scale ratio of the input image and the RoIPool feature map. Similarly, another quantization occurs when splitting the RoIPool feature map into 7×7 spatial grids before max pooling. RoIAlign [14] leverages bilinear interpolation on RoIPool layer to avoid the harsh quantization of the RoI boundaries. The specific method is that four regular locations around each RoI grid are interpolated with values respectively and then RoI pooling is performed to extract fixed features.

Multi-level Alignment. As illustrated in the Fig.3, our contribution is that multi-level alignment (MLA) strategy is first introduced to effectively reserve exact spatial locations from multiple layers and tackle multi-scale object detection problem. The proposed MLA method is different from the feature pyramid strategies shown in Fig.3 but can be combined with those feature pyramid strategies applied in proposal generation networks to efficiently solve the scale variation problem together. Our MLA strategy categorizes all RoI proposals into three scale levels (based on the height pixels) and respectively assigns proposals in each level to corresponding feature layer within a backbone ConvNet to implement RoI aligning operation. Therefore, how to design and select corresponding RoI feature layer is a critical component in our strategy. Given that deeper convolutional layers within a ConvNet perform weakly for small object location, it would provide limited information for 7×7 RoI aligning. Therefore, as presented in Fig.3, we upsample a high-resolution layer (eg. *conv4_3*) by deconvolution operation in order to generate a new RoI feature map. In contrast to input image unsampling, that operation can reduce extra cost for computation and memory footprint. The later experiments show that upsampling *conv4_3* by a factor 2 can achieve best trade-off between accuracy and efficiency for small object detections in our framework rather than upsampling *conv4_3* by a factor 4. We also do not select *conv3_3* which has the same resolution as deconvoluted *conv4_3* since a lower layer of the backbone (eg. *conv3_3*) can not provide enough semantic context.

As for specific multi-level alignment application in baseline2, our model build three branches (MLA-4, MLA-8 and MLA-16) after *conv4_3_2x*, *conv4_3* and *conv5_3*. 'n' in 'MLA-n' means the scale ratio of input image to the corresponding RoIAlign feature map. Firstly, if an object proposal has a height smaller than 80 pixels, we allocate these small RoIs to deconvolutional *conv4_3_2x* layer (MLA-4). Middle RoIs with a height between 80 and 160 pixels are distributed to *conv4_3* layer (MLA-8) and the rest proposals the size of which exceeds 160 pixels are assigned to *conv5_3* layer (MLA-16). Next, for instance in MLA-4, a 190×190 RoI proposal is mapped into a 47.5×47.5 patch from the *conv4_3_2x* layer before 7×7 max pooling, which is namely the quantization-free RoI aligning.

The predictor heads in Fast-RCNN network, namely the architectures after RoIPool layer, adopt two successive 4096-*fc* layers with *ReLU* activations and dropout layers for specific class classifiers and bounding box regressors. In our refined detection network, in order to enhance computational speed, each MLA is connected by only one fully-connected layer with 4096 channels. Nevertheless, it does no harm to accuracy due to the thin feature maps (with 512 channels) before RoI aligning and relative light prediction burden (less RoIs) in each MLA line. In particular, the three predictor heads connected to MLAs all share their parameters regardless of their RoI scale levels.

C. *FMLA-CNN* Framework and Efficiency Optimization

When we apply the best fusion module to the refined detection network with MLAs, as shown in Fig.1, finer multi-level strides would be achieved to refine location and high-level semantic feature maps at all scales would be built to promote classification. The training for feature fusion proposal network has been explained in Section.II-A. The first-stage resulting model is utilized to initialize the shared layers in head detection network. During fine-tuning in the second stage, input RoI proposals are firstly split into group {small, middle, large} according to height and then assigned to corresponding RoIAlign feature layer {*conv4_3_2x*, F2, F3}. Again, the three predictor heads share parameters regardless of their levels. However, it is natural to produce information and computation redundancy after the combination. Apart from sharing features for feature fusion proposal network and refined detection network, we replace RoIAlign layer with RoIPool layer in MLA-8 and MLA-16 since the two layers have similar results in that two MLA lines while RoIAlign costs more computation.

Apart from architecture adjustments to optimize efficiency, we also implement the following data strategy. Substantial background areas which are of no avail for training occupy a large amount of memory. Also, traffic images contains different resolutions and object scales (especially for numerous small objects in the distant field) compared with nature images. Thus, each training image is processed as follows:

- Reshape the original input image by 1.5 times. (e.g., reshape 1241×376 to 1920×576)
- Randomly crop and sample a patch around objects from the whole image.
- The maximum crop size is 768×576 and the minimum jaccard overlap for the objects is 0.1, 0.4 and 0.8.

This data strategy is conducted in all models for ablation experiments on validation set, guaranteeing that all improvements are only introduced by the architecture adjustments.

Fig. 4. Baseline2 vs FMLA-CNN in ablation experiments. The top row presents the validation results from the baseline2 model and the bottom row shows the outcomes from the FMLA-CNN model.

III. Experiments and Analysis

A. Datasets and Parameter Settings

The experimental dataset consists of 7481 labled images and 7518 testing images both obtained from the KITTI object detection benchmark [11], which contains a total of 40 labeled on-road objects. Given that the KITTI benchmark does not provide ground truth labels for testing set, we follow the data split in [15] to implement our ablation experiments by dividing the 7481 labeled training/validation images into training set with 3712 images and validation sets with 3769 images. For cars it requires a high overlap of 70%, whereas for pedestrians it requires an overlap of 50% for detection. Three-level difficulties including 'Easy', 'Moderate' and 'Hard' are defined in [11]. Note that **Average Precision (AP) in moderate difficulty** is the decisive evaluation indicator especially when we compare our proposed models with the two baselines or other state-of-the-art methods in the KITTI ankings.

As elaborated in Section.II-C, input images are reshaped to 1920×576 and jointly trained in two stages. The iterations of two training stage are 1.5×10^4 and 3×10^4 respectively. We use the 5×10^{-4} learning rate for 1.5×10^4 iterations in the first stage, and then fine-tune the fist-stage model using SGD strategy with initial learning rate 5×10^{-4} which is dropped by a factor of 10 after each 1.5×10^4 iterations. In addition, the training batch size is 4 corresponding to the 1920×576 input with one GPU (Nvidia GTX Titan X).

B. Ablation Experiments of Our Models on Validation Set

The structures of all models in Table I are elaborated thoroughly in the previous section. First of all, the large improvement (about **14**%) from baseline1 to baseline2 demonstrates that the detectors with pyramidal feature hierarchy always perform better than single RPN feature. The improvement further proves that four proposal generation branches with two-scale anchors for each feature hierarchy can reduce redundant computation but still achieve more excellent accuracy than a single RPN layer with nine anchors. Secondly, when evaluating our two feature fusion methods which are both developed from the baseline2, we observe that fusion2 model (top-down pathway fusion) with three fused levels has the best fusion performance, around **2.4**% enhancement in contrast to non-fused pyramidal feature hierarchy. Fusion1 model (lightweight feature fusion) has inferior performance than fusion2 model especially for hard AP (an indicator influenced largely by small object detection), indicating that the most reasonable way to associate hierarchical feature maps is to insert high-level semantics to all pyramidal feature hierarchy with top-down pathway fusion. Fusion2 model can integrate more semantic context into high-resolution features in shallow layers, thus detecting small object more effectively than fusion1 model. Based on the results of our experiments, different from the conventional FPN with four-level, the fusion with three layers performs slightly better than four layers since our backbone VGG16 is far shallower than original ResNet.

The third row from the bottom in Table I presents the results for devoting MLAs model to baseline2 model, which is elaborated in Section.II-B. It proves that the proposed MLA strategy could improve detection accuracy by **2.7**%, which is similar to our best fusion strategy, but the MLAs model spends triple run-time compared with the fusion model. The original combined model enhance the the performance by around **3.7**% with 0.235 seconds per image (s/img), indicating that the two strategies for multi-scale object detection has the repeated contribution to the performance and the simple combination may produce redundant computation. Given the real-time requirement in driving scenario, we make several architecture adjustments which are illustrated in Section.II-C when intergerating the two strategies. The combination model after adjustments are defined as FMLA-CNN (about **89.91**% with **0.175** s/img) and we can observe that the efficiency of FMLA-CNN is greatly improved without damaging the accuracy of original combination model.

TABLE I

BASELINES VS OUR NEW STRATEGIES ON KITTI VALIDATION SET

Model	Car			Time
	Easy	Moderate	Hard	(s/img)
baseline1(Fig.2(a))	83.26	72.64	62.33	0.290
baseline2(Fig.2(b))	87.29	86.26	69.51	0.130
baseline2+fusion1(Fig.2(d))	91.07	87.93	73.64	0.155
baseline2+fusion2(N=3)	92.28	**88.67**	75.86	**0.125**
baseline2+fusion2(N=4)	90.56	88.47	88.47	0.132
baseline2+MLAs	93.60	**88.98**	76.23	**0.390**
baseline2+fusion2+MLAs	93.22	89.96	79.32	0.235
FMLA-CNN	93.48	**89.91**	79.29	**0.175**

TABLE II

THE STATE-OF-THE-ART METHODS VS OUR MODEL ON KITTI TEST SET

Model	Car			Pedestrian			Time
	Moderate	Easy	Hard	Moderate	Easy	Hard	(s/img)
Regionlets [16]	76.56	86.50	59.82	61.16	72.96	55.22	1.00
spLBP [17]	77.39	80.16	60.59				1.50
Faster R-CNN [2]	79.11	87.90	70.19	65.91	78.35	61.19	2.00
RFCN [3]	79.44	88.69	70.06	58.06	74.44	51.14	0.30
FPN [9]	79.48	89.45	69.81				5.00
YOLOv2-3cls [4]	85.65	88.01	74.16	55.43	70.05	51.55	0.05
Mono3D [18]	87.86	90.27	78.09	66.66	77.30	63.44	4.20
MM-MRFC [19]	88.20	90.93	78.02	69.96	82.37	64.76	0.05
3DOP [15]	88.34	90.09	78.79	67.46	82.36	64.71	3.00
SubCNN [20]	88.86	90.75	79.24	71.34	83.17	66.36	2.00
SDP+RPN [12]	89.42	89.90	78.54	70.20	79.98	64.84	0.40
RRC [21]	90.22	90.61	87.44	75.33	84.14	70.39	3.60
our FMLA-CNN	88.83	90.45	77.04	73.75	83.86	68.06	0.17

C. Comparisons with Other Novel Approaches on Test Set

In order to compare with other state-of-the-art approaches, we trained our FMLA-CNN model with 7481 labeled data, and then submitted our car and pedestrian results to the KITTI leaderboards. Our pedestrian AP **ranked 10th** with **73.75**% after submission, whereas our car AP achieves **88.83**% which is just 1.72% lower than the top car result. In addition, the run-time of our model with 1.5x image input and a single GPU reaches **0.17** s/img, which is only **1/10**∼**1/20** of the run-time for top 3 methods in the car leaderboard. Thus, our FMLA-CNN framework achieves outstanding performance on object detection in autonomous driving scenario.

Table II makes the comparison between the FMLA-CNN and other latest methods while only published works or popular methods in the leaderboards are shown in Table II. Faster R-CNN is similar with our baseline1 except for data strategy. FMLA-CNN performs better than Faster R-CNN with a large margin and reaches 10 times speed, once more verifying the effectiveness of our two strategies. In addition, traditional four-level FPN based on ResNet only achieved 79.48% for car AP, around 9.3% lower than our model based on VGG16 variant, which certifies our assumption that sometimes ResNet backbone is too redundant to perform well on the location-oriented tasks. As fusion2 model contributes much to high performance of the FMLA-CNN, that comparison can roughly examine our successful modification on traditional FPN especially for traffic object detection. Furthermore, when it comes to small object detection like pedestrians, FMLA-CNN outperforms the very recent MM-MRFC [19], SDP+RPN [12] and SubCNN [20] with **2.4**% ∼ **3.8**%, and achieves much better performance than 3DOP [15], Mono3D [18] and other prevalent derectors.

IV. CONCLUSION

In conclusion, aiming to efficiently detect multi-scale objects in autonomous driving domain, this paper introduces a new hierarchical ConvNet detector which utilizes a remodified feature pyramid network and a multi-level alignment module respectively in two stages. Experimental results verify the significant benefits of the two strategies to scale variation problem, and also demonstrate that our FMLA-CNN method with high computational efficiency achieves outstanding performance on KITTI benchmark especially for small objects.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

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

[3] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.

[4] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.

[5] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

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

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

[8] Z. Li and F. Zhou, "Fssd: Feature fusion single shot multibox detector," *arXiv preprint arXiv:1712.00960*, 2017.

[9] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," *arXiv preprint arXiv:1612.03144*, 2016.

[10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[11] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Kitti object detection benchmark," 2016.

[12] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2129–2137.

[13] W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," *arXiv preprint arXiv:1506.04579*, 2015.

[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," *arXiv preprint arXiv:1703.06870*, 2017.

[15] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *Advances in Neural Information Processing Systems*, 2015, pp. 424–432.

[16] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for generic object detection," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 17–24.

[17] Q. Hu, S. Paisitkriangkrai, C. Shen, A. van den Hengel, and F. Porikli, "Fast detection of multiple objects in traffic scenes with a common detection framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1002–1014, 2016.

[18] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3d object detection for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2147–2156.

[19] A. Daniel Costea, R. Varga, and S. Nedevschi, "Fast boosting based detection using scale invariant multimodal multiresolution filtered features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6674–6683.

[20] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 924–933.

[21] J. Ren, X. Chen, J. Liu, W. Sun, J. Pang, Q. Yan, Y.-W. Tai, and L. Xu, "Accurate single stage detector using recurrent rolling convolution," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 752–760.