DRONE-BASED VEHICLE FLOW ESTIMATION AND ITS APPLICATION TO TRAFFIC CONFLICT HOTSPOT DETECTION AT INTERSECTIONS

[1]*Ping-Yang Chen,* [2]*Jun-Wei Hsieh,* [3]*Munkhjargal Gochoo,* [4]*Ming-Ching Chang,* [5]*Chien-Yao Wang,* [1]*Yong-Sheng Chen, and* [5]*Hong-Yuan Mark Liao*

[1]Department of Computer Science, National Chiao Tung University, HsinChu, Taiwan.
[2]College of Artificial Intelligence and Green Energy, National Chiao Tung University, HsinChu, Taiwan.
[3]College of Information Technology, United Arab Emirates University, Al-Ain, United Arab Emirates
[4]Department of Computer Science, University at Albany, State University of New York, NY, USA
[5]Institute of Information Science, Academia Sinica, Taiwan.

ABSTRACT

Drones can provide a wider field of view, high mobility and flexibility for monitoring and analyzing traffic flows and safety conditions. In case of a perpendicular viewing angle to the ground, there will be a very less occlusion that can occur and make vehicle tracking be easier. Thus, a drone-based solution will be better for traffic conflict hotspot detection at an interaction. However, due to its observation far from the ground, limited battery time, and bandwidth, this solution should be edge-based and have a good recognition rate in small object detection. However, current edge-based SoTA (state-of-the-art) methods are weak in a small object detection. We propose CoBiF net (Concatenated Bi-Fusion feature pyramid network), a one-stage object detection model for a real-time small object detection, which consists of SPP (spatial pyramid pooling), FE (Feature Extractor), CF (Concatenated Feature) block, and BFM (Bottom-up Fusion Module). CoBiF net is memory-and-bandwidth saving for the most edge devices. Extensive experiments on UAVDT benchmark show the proposed method achieved the SoTA results for the small object detection task in terms of accuracy and efficiency.

Index Terms— Small object detection, traffic flow estimation, traffic conflict hot spot detection, edge computing

1. INTRODUCTION

Traffic conflict hotspot detection at intersection [1]-[3] is an important task in an intelligent transportation system. The analysis results can provide helpful information for traffic safety and control tasks such as sign design, and further prevent potential traffic accidents. A pre-requisite for enabling this analysis is to accurately locate vehicles in video images so that vehicle attributes such as speeds, types, and can be extracted, tracked, and then counted. Roadside cameras or Lidar sensors are not suitable for this analysis due to their limited field of views and the occlusions between vehicles. In addition, using multiple cameras needed in an interaction leads to other synchronization and calibration problems between cameras. The drone camera can shoot vehicles from a bird-view perpendicular to the ground, which can overcome the problem of vehicle perspective distortion. From a top-down view, the same car is always consistent in different frames and can be easily tracked and counted. Although considerable concerns and limitations still exist, such as limited battery time, safety concerns, etc., its mobility and flexibility [1], [3] make it be widely used in the transportation field to analyze traffic flow and safety conditions. However, due to its observation far from the ground and limited battery time and bandwidth, the drone-based solutions [4] should be edge-based and have a good accuracy in small object detection.

Recently, the accuracy of object detection models have been improved by a large margin with various state-of-the-art (SoTA) models like FPN[5], YOLOv3[6], and SSD[7]. To increase the accuracies of object classification and detection, a very deep CNN architecture is often adopted and usually brings up a lot of computation cost. However, using such deep architectures cannot satisfy the requirement of short inference time on mobile devices.

To improve the accuracy on small object detection, a feature pyramid (FP) structure is commonly adopted in the SoTA detectors due to its multi-scale structure. With this structure, abundant spatial information can be extracted from the last few layers of the network backbone. There are few common types of FPs employed in object detection models, i.e., pyramidal feature hierarchy (bottom-up), hourglass (bottom-up and top-down), SPP (spatial pyramid pooling), SPP + multi-scale fusion, which are adopted in SSD[7], FPN [5], PFPN [12], and SPP [13], respectively. Hourglass FPs are generated by fusing last three layers of a backbone. On the other hand, SPP-based FPs [12][13] are generated from the last layer of a backbone. Thus, hourglass FPs contain richer multi-scaled features than SPP-based FPs, and lead to a higher accuracy in small object detection. However, the Hourglass-based method adopts a top-down path to generate

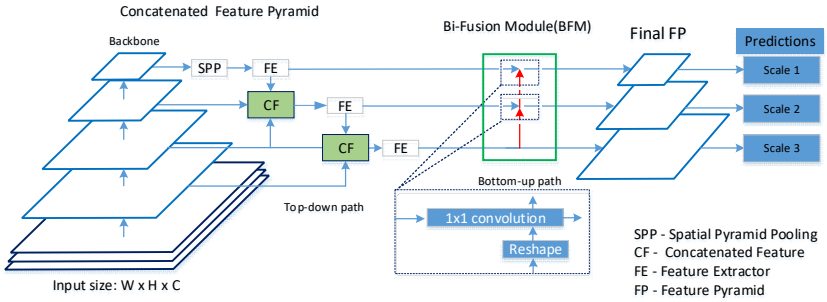

Fig. 1. Proposed concatenated feature Bi-Fusion pyramid network (CoBiF net).

a three-scale FP for object prediction by summing features from the deeper layers to the shallower layers of the backbone. This one-directional path will prohibit the networks from detecting small objects.

Fig. 2. Traffic scene captured from a bird-view camera mounted under a drone. The traffic conflict hotspots were detected and shown with colors.

This paper proposes a novel deep CNN architecture to detect smaller vehicles for real-time traffic flow monitoring installed at intersections from a bird-view camera on an embedded device. To accurately estimate the traffic flow, the key task is to well detect smaller objects such as pedestrians, cars or motorcycles. Rather than using pooling operations, concatenating operation is adopted in this paper to generate concatenated FPNs (CFPNs) from which smaller objects (even <15×15 pixels) can be detected. This paper proposes CoBiF net (Concatenated Bi-Fusion feature pyramid network), a one-stage object detection model for real-time object detection, which consists of SPP (spatial pyramid pooling), FE (Feature Extractor), CF (Concatenated Feature) block and BFM (Bottom-up Fusion Module). The concatenated BiFPNs are generated not only by up-sampling features from deeper layers but also by reorganizing features from shallow layers. This concatenated BiFPN structure can hold the spatial information of a smaller object at the end of network but also increase the efficiency when running on an embedded system. More importantly, our new model can accurately detect smaller vehicles even with significant distortions. Its performance on TX2 is up to 22 fps. On the UAV Dataset [22], its accuracy also outperforms other state-of-the-art methods. Major contributions of this work are noted as follows:

- A new CoBiF net is proposed for estimating traffic flows from a bird-view camera. Its feature preserving property is very suitable for detecting small objects even from a drone;
- Spatial information of extremely smaller objects (<15×15 pixels) can be extracted at the end of network;
- First on-board drone-based traffic flow estimation system which can detect smaller objects and traffic conflict hotspots is implemented.

2. RELATED WORKS

In recent years, deep learning has leaded dramatic improvements for the object detection. In the literature, YOLO [8] achieved the state-of-the-art performance by integrating bounding box proposal and subsequent feature resampling as one stage. Next, SSD [7] employed in-network multiple feature maps for detecting objects with varying shapes and sizes, and this feature makes SSD more robust than YOLO. For better detection of small objects, FPN [5] is developed using a feature pyramid (FP) structure and it achieves a higher detection accuracy on small objects. Later, the state-of-the-art YOLOv3 [6] was developed by adopting the concept of FPN. Similarly, RetinaNet [15], a combination of FPN and ResNet as a backbone, proposes the use of focal loss to significantly reduce false positives in one-stage detectors by dynamically adjusting the weights of each anchor box.

The above methods for improving the accuracy often come at a cost: deepening networks requiring high computational resources and thus failing to detect objects in real time on many mobile or embedded applications. In [17], Howard, *et al.* proposed a MobileNet by using point-wise group convolutions to reduce computation complexity of 1×1 convolution. In [18], Zhang *et al.* proposed a ShuffleNet which utilizes two new operations, a pointwise group convolution and a channel shuffle, to greatly reduce computation cost while decreasing the accuracy of the smaller object detection. Most lighting architectures try to use depth-wise operations to replace pixel-wise operations. This way can improve its efficiency but lead to the loss of detection accuracy.

3. PROPOSED METHOD

FPN is a top-down method to bring semantically robust features from the last layer to discriminate objects from the background. However, FPN cannot preserve object's accurate positions due to the effect of pooling and quantization. To solve this problem, we propose CoBiF net (Concatenated Bi-Fusion feature pyramid network), a one-stage object detection model for real-time object detection, which consists of SPP (spatial pyramid pooling), FE (Feature Extractor), CF (Concatenated Feature) block and BFM (Bottom-up Fusion Module). Fig. 1. illustrates the architecture of CoBiF net. All parts of CoBiF net is elaborated in details as follows.

3.1 CF Block

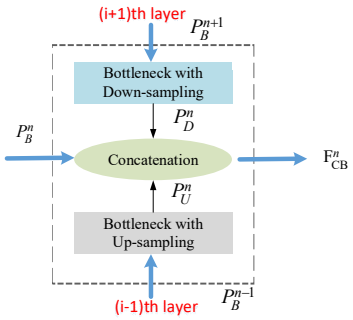

Fig. 3. Proposed concatenated feature block (CF block).

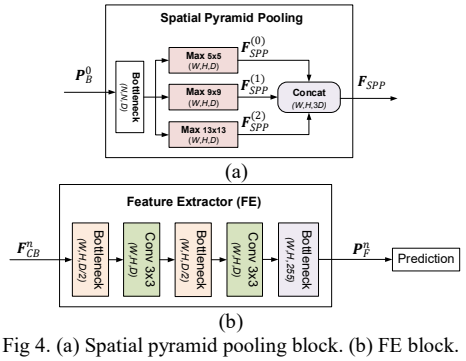

Fig 4. (a) Spatial pyramid pooling block. (b) FE block.

CF block has one concatenation and two 1×1 convolutions as a bottleneck layer to adjust the channels of feature maps from different layers. Before the concatenation of three feature maps, a deeper feature map is up-sampled and a shallow feature map is down-sampled as operations for different sizes of the feature map, *i.e.*, max pooling and bilinear up-sample. Unlike concatenation methods in SoTA methods, i.e., FPN [2] or CFPN [23], the proposed CF block concatenates contextual features of not only adjacent layers but also even deeper (*n*-1)th layer. In other words, CF block fuses features from 3 adjacent scales (shallow P_D^n, current P_B^n, and deep P_U^n) of a backbone to richen the features for better detection. As shown in Fig. 3, the CF block, concatenating current scale features directly from a backbone and outputs of the bottleneck layers from a deeper and shallow scales, can be defined as follows:

$$F_{CB}^n = \begin{cases} [P_B^n, P_U^n, P_D^n], & if\ n > 0, \\ [F_{SPP}^0], & if\ n = 0. \end{cases} \quad (1)$$

To increase the accuracy of the classification in the backbone of deeper scales, the SPP module generates robust semantic features in the deep final layer without high-computational operations. Fig. 4(a) shows SPP that consists of a bottleneck layer, three max-pooling layers with kernel sizes of (5×5), (9×9), and (13×13), and a concatenation. The number of feature channels is reduced to half with the bottleneck layer, then three groups of max pooled features maps with the same dimension, F_{SPP}^0, F_{SPP}^1 and F_{SPP}^2, are generated. For concatenation purposes, all three max-pooling operations employ a zero-padding and a stride of 1 to create the same sized output feature maps. The concatenated max pooled features, which will become the output of SPP, can be calculated as follows:

$$F_{SPP} = [F_{SPP}^{(0)}, F_{SPP}^{(1)}, F_{SPP}^{(2)}]. \quad (2)$$

Fig. 4(b) illustrates the flowchart of the FE block. It is put after the CF block and SPP to extract more contextual and semantic features from fused features of 4 adjacent scales. The FE block consists of 2 consecutive parts of feature extraction where each part includes one bottleneck layer and a 3x3 convolutional layer. The former is employed to reduce the number of channels from D to D/2. The latter is used to extract contextual features. The output of the second bottleneck layer is fed to CP at the shallower scale for fusion.

3.2 Bi-Fusion Module

As described before, SoTA hourglass-based methods adopt a top-down fusion path to generate a three-scale FP for object prediction by bringing semantic features from the deepest layer to other shallow layers. To circulate semantic and localization information from a bottom-up pathway, current bidirectional methods adopt a memory-and-bandwidth consuming way to create new feature maps from shallow layers for feature fusion to predict object candidates with better accuracy. Fig. 2. shows the architecture to construct a bi-fusion feature pyramid. The output of the (*i*-1)th CF and FP is the input of the *i*th CF module to generate more semantic contexts. This "re-using" mechanism of features allows the model to be memory-and-bandwidth efficient and suitable for embedded applications. Circulating semantic and localization information bi-directionally from deep and shallow layer also significantly improves the accuracy of small object detection and also conveys extra localization information for better localization of large objects.

3.3 Vehicle Tracking

For vehicle tracking, we adopt an overlap IOU (intersection-over union) to propose a real-time MOST (Multiple Object

Tracking System) without using objects' visual features. The overall complexity of the method is very low compared to other state-of-the-art object trackers, such as Kalman filter and Hungarian algorithm. Since no visual information is used, it can even run on embedded systems efficiently with frame rates exceeding 100kfps.

4. EXPERIMENTAL RESULTS

4.1. Data preparation and model training

Model evaluations are conducted on the Unmanned Aerial Vehicle Dataset benchmark [22] using a machine with NVIDIA V100. CoBiF net is compared with the latest state-of-the-art one-stage object detectors in terms of accuracy and efficiency. The metric adopted for performance evaluation is Average Precision (AP). Experiment follow the protocol provided by [22] and evaluate the performance using the PASCAL style AP and evaluate on NVIDIA Titan X according to the GPU capacity.

4.2. Evaluation

Table I shows a comparison with several existing single and two-stage detectors on the UAVDT benchmark [22]. In case of Faster-RCNN [24], R-FCN [26], SSD [7], RON [25], and LRF[28], their results are taken from [22]. Our detector outperforms LRF with an AP score of 60.98 with the same backbone VGG-16. Moreover, the backbone of PeleeNet [27] was adopted for our CoBiF net for real time object detection directly on a drone with NVIDIA Jetson TX2. It outperforms VGG-16 with an AP score of 63.16 in effectiveness and twice the speed in efficiency. The evaluation comparisons on the UAVDT benchmark can prove the superiority of our method on small object detection.

Table I. Comparisons on UAVDT benchmark

Methods	backbone	input size	AP	FPS
Fster-RCNN[24]	VGG-16	1024x540	22.32	2.8
R-FCN[26]	ResNet-50	1024x540	34.35	4.7
SSD[7]	VGG-16	512x512	33.62	120
RON[25]	VGG-16	512x512	21.59	11.1
RetinaNet[15]	ResNet-101-FPN	512x512	33.95	25.0
LRF[28]	VGG-16	512x512	37.81	91.0
Ours	VGG-16	512x512	**60.98**	66.00
Ours	Pelee[27]	512x512	**63.16**	126.74

Ablation Studies

Table II tabulates the ablation studies to show the advantages of CF, BFM, CoBiF. For the performance evaluations on edge devices, the "Pelee" backbone [27] is adopted. It is noticed that the frame rate difference between before/after using the BFM or CF module is minor. If only the CF or BFM module is adopted, CF makes more significant improvements on a shallower backbone (VGG 16) than BFM. However, when a deeper backbone is adopted, the BFM module makes better improvements than the case only using CF. From the table, we can see that CoBiF net outperforms on all categories.

Table II. Ablation studies of CF, BFM, and CoBiF.

Models		Input size				FPS	AP
backbone	**CoBiF** [our]	416	512	with BFM	with CF		
VGG-16		✓				86.35	50.25
		✓		✓		86.35	50.31
		✓			✓	86.28	53.44
	✓	✓		✓	✓	79.74	**56.88**
			✓			70.07	53.98
			✓	✓		69.15	54.42
			✓		✓	69.12	56.01
	✓		✓	✓	✓	66.00	**60.98**
Pelee [27]		✓				151.74	55.13
		✓		✓		139.08	58.74
		✓			✓	142.54	57.58
	✓	✓		✓	✓	129.89	**60.97**
			✓			139.27	59.98
			✓	✓		132.10	62.31
			✓		✓	134.22	61.27
	✓		✓	✓	✓	126.74	**63.16**

Due to the limited bandwidth, the drone-based solutions should be edge-based and good in small object detection. The uploading speed for 4G and 5G are 6.19Mbps and 59.83Mbps, respectively. Table III shows the ablation studies of object detection on edge devices or on cloud servers. The edge device is TX2 and the GPU for cloud server is NVIDIA GeForce RTX 2080ti. The up-loading time will increase the latency time for object detection. Although the GPU on the cloud server is much more powerful than the edge device on a drone, the cloud-based solution cannot detect objects if the input frame is with a 4K dimension.

Table III. Ablation studies of object detection with/without data transmission.

Resolution	4G	5G	CoBiF
SD	1.5fps	28fps	26.7fps
HD	x	15fps	22.1fps
Full HD	x	6fps	20fps
4K	x	x	15.2fps

Traffic conflict hot spot detection

After vehicle detection and tracking, different vehicle trajectories can be detected and recorded. Then, an open source "SSAM (Surrogate Safety Assessment Model)" can be used to find different traffic conflict. Due to the limited paper space, the result of "vehicle-pedestrian" conflict hotspots detected from a drone is only shown in Fig. 1.

5. DISCUSSIONS AND CONCLUSSIONS

Our proposed CoBiF net outperforms the SoTA models on UAVDT benchmark [22] to prove its capability for a real-time traffic estimation on a drone. The effectiveness and efficiency of CF and BFM modules were proven and CoBiF can be generalized to different backbones. The drone-based solution can provide a good mobility and flexibility for traffic conflict hotspot detection. In the near future, if the problem of limited battery can be solved, this solution can be widely used in ITS.

REFERENCES

[1] T.-Z. Xiang, G.-S. Xia, and L.P. Zhang, "Mini-Unmanned Aerial Vehicle-Based Remote Sensing: Techniques, applications, and prospects," *IEEE Geoscience and Remote Sensing Magazine,* vol. 7, no.3, pp.29-63, 2019.

[2] K. Kanistras, G. Martins, M. J. Rutherford, and K. P. Valavanis, "Survey of unmanned aerial vehicles (UAVs) for traffic monitoring," In Handbook of Unmanned Aerial Vehicles; Springer: Dordrecht, The Netherlands, pp. 2643-2666, 2015.

[3] M. A. Khna, et al., "Unmanned Aerial Vehicle-Based Traffic Analysis: A Case Study for Shockwave Identification and Flow Parameters Estimation at Signalized Intersections," Remote Sensing, vol.10, no.3, 458, 2018.

[4] J.-J. Wang, et al., "Bandwidth-Efficient Live Video Analytics for Drones Via Edge Computing," *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, 2018.

[5] T.-Y. Lin, et al., "Feature pyramid networks for object detection," *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117-2125, 2017.

[6] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv:1804.02767*, 2018.

[7] W Liu, et al., "SSD: Single shot multibox detector," *The IEEE Conference on Computer Vision*, pp. 21-37, 2016.

[8] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263-7271, 2017.

[9] K. He, X. Y. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.

[10] S.-W. Kim, et al. "Parallel Feature Pyramid Network for Object Detection," In *ECCV*, 2018.

[11] B. Bosquet, et al., "STDnet: A ConvNet for Small Target Detection," In *BMVC*, 2018.

[12] S.-W. Kim, et al., "Parallel Feature Pyramid Network for Object Detection," In *ECCV*, 2018.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," In *ECCV*, 2014.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," In *CVPR*, 2016.

[15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," In ICCV, 2017.

[16] H. Law and J. Deng, "CornerNet: Detecting Objects as Paired Keypoints," In *ECCV*, 2018.

[17] A. G. Howard, et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv:1704.04861, 2017.

[18] X. Zhang, et al., "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," In *ICCV*, 2017.

[19] S. Zhang, et al., "Single-Shot Refinement Neural Network for Object Detection," arXiv preprint arXiv:1711.06897, 2017.

[20] A. Van Etten, "You Only Look Twice: Rapid Multi-Scale Object Detection in Satellite Imagery," arXiv preprint arXiv:1805.09512, 2018.

[21] E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information," Proc. IEEE Int. Workshop Traffic Street 51 Surveill. Safety Secur. (AVSS), pp. 1-6, Sep. 2017.

[22] D. W. Du, *et al*., "The unmanned aerial vehicle benchmark: Object detection and tracking," In ECCV, 2018.

[23] P.-Y. Chen, *et al*., "Smaller object detection for real-time embedded traffic flow estimation using fish-eye cameras," ICIP 2019.

[24] S. Ren, *et al.*, "Faster r-cnn: Towards real-time object detection with region proposal networks," In: Advances in neural information processing systems, pp.91-99, 2015.

[25] T. Kong, *et al*., "Ron: Reverse connection with objectness prior networks for object detection," In CVPR, 2017.

[26] J. F. Dai, Y. Li, K. M. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," In NIPS, 2016.

[27] R. J. Wang, X. Li, and C. X. Ling, "Pelee: A Real-Time Object Detection System on Mobile Devices," In NIPS, 2018.

[28] T. Wang, *et al.*, "Learning Rich Features at High-Speed for Single-Shot Object Detection," In ICCV 2019.

[29] L. Pu, L. and R. Joshi, "Surrogate safety assessment model (SSAM): software user manual," In: Federal Highway Administration Report FHWA-HRT-08-050.