DEEP REAL-TIME HAND DETECTOIN USING CFPN ON EMBEDDED SYSTEMS

[1]*Pirdiansyah Hendri*, [2]*Jun-Wei Hsieh*, [3]*Ping-Yang Chen*, [4]*Munkhjargal Gochoo, and* [3]*Yong-Sheng Chen*

[1]Depart of Com. Scie. Eng. ,National Taiwan Ocean University, Taiwan., [2]College of AI, National Chiao Tung University, Taiwan,

[2]College of C. S., National Chiao Tung University, Taiwan., [4]College of I. T., United Arab Emirates University, UAE.

Introduction

Small object (<30×30 pixels) detection is still a challenging task for existing methods. Thus, we propose a shallow model named **Concatenated Feature Pyramid Network** (CFPN) which has concatenated block (CB) that reduces the number of convolutional layers and employs concatenations instead of time-consuming algebraic operations. The superiority of CFPN is confirmed on the COCO and an in-house HandFlow datasets on Nvidia TX2.

Concatenated Feature Pyramid Network

We propose inter-scale **Concatenation Block** (CB), shown in Fig. 4., consisting of few number of convolutional layers and concatenation operations for the replacement of addition and convolutional operations.

The major contributions:

- Instead of using pooling operations, concatenating operation is adopted to generate CFPNs from which even less than 15 15 small hands can be well detected.
- The new CFPNs can hold the spatial structure of small objects at the end of network but also increases the efficiency of hand detection in real time.
- A real time small hand detection system is proposed even under an embedded system.

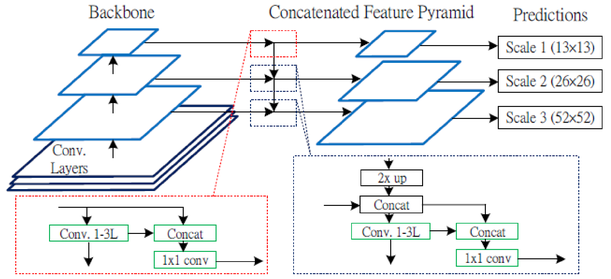

Fig. 3. Proposed Concatenated Feature Pyramid Network (CFPN).

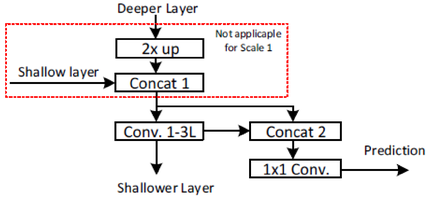

Fig. 4. Proposed Concatenation Block (CB), consisting of two concatenations, 2x upscaling operation, and 2-4 convolutional layers. (Conv. 1-3L) refers to 1-3 convolutional layers.

TABLE I. PERFORMANCE COMPARIONS ON 416×416 RESOLUTION HANDFLOW DATASET ON TX2 EMBEDDED DEVICE.

Model	Backbone	FP	CB	mAP:0.5	FPS
YOLOv3 [37]	darknet53	3	-	84.0	5
YOLOv2 [36]	darknet19	-	-	78.5	6
YOLOv3-tiny [37]	tiny15	2	-	71.8	15
YOLO v2 SpeedUP[36]	darknet19	-	-	68.0	13
YOLT[47]	darknet19	-	-	85.6	5.5
FPN-tiny	tiny15	3	-	90.3	18
CFPN-1	~tiny15	2	1	91.3	**33**
CFPN-3	~tiny15	3	3	**95.6**	**33**

TABLE II. PERFORMANCE COMPARIONS AMONG DIFFERENT METHODS ON COCO DATASET.

Methods	Backbone	Train set	AP50	APS
Faster R-CNN [31]	VGGNet-16	trainval	42.7	-
R-FCN [30]	ResNet-101	trainval	51.9	10.8
YOLOv2 [36]	DarkNet-19	trainval35k	44	5
YOLOv3-608[37]	DarkNet-53	trainval35k	50.2	16.9
SSD512 [32]	VGGNet-16	trainval35k	48.5	10.9
RefineDet320 [45]	ResNet-101	trainval35k	51.4	10.5
RefineDet512 [45]	VGGNet-16	trainval35k	54.5	16.3
PFPNet-S512 [34]	VGGNet-16	trainval35k	54.8	16.3
Proposed CFPN	**Darknet-53**	**trainval35k**	**54.8**	**18.4**

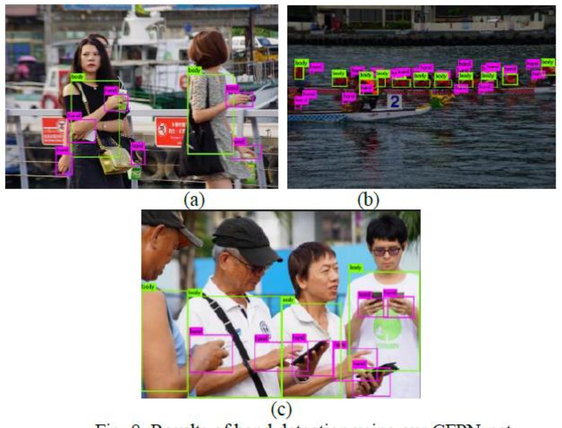

Fig. 8. Results of hand detection using our CFPN-net.

Fig. 9. Results of hand detection using our CFPN-net.

Results & Conclusions

Our proposed CFPN model outperforms the existing state-of-the-art models on HandFlow and COCO dataset to prove its capability for hand detection on TX2 embedded device. Furthermore, the existing methods (YOLO3, SSD, RefineDet) cannot perform real-time on TX2 embedded device since their FPS < 3.