CFENet: An Accurate and Efficient Single-Shot Object Detector

Qijie Zhao
Visual Data Interpreting and Generation Lab
Institute of Computer Science & Technology, Peking University
Homepage: qijiezhao.github.io
Mail: zhaoqijie@pku.edu.cn

- Team: VDIG
- Members: Qijie Zhao, Tao Sheng, Feng Ni, Yongtao Wang
- Lab: VDIG lab, ICST, Peking University
- Ranking: 1st

Qijie Zhao

Third-year M.S candidate student at Peking University

Research Interest: Object Detection, semantic Segmentation, Action recognition.

Tao Sheng

First-year M.S candidate student at Peking University

Research Interest: Object Detection, semantic Segmentation.

Feng Ni

Research Intern at ICST, Peking University.

Research Interst: Object Detection, Recognition, semantic segmentation.

Mentor: Yongtao Wang

Associate researcher at ICST, Peking University

Email: {zhaoqijie, shengtao, nifeng, wyt}@pku.edu.cn

Outline

- VisDrone Challenge Overview
- Dataset Analysis
- Object Detection from Video, VID
- CFENet
- Experiment
- Visualization
- Future work

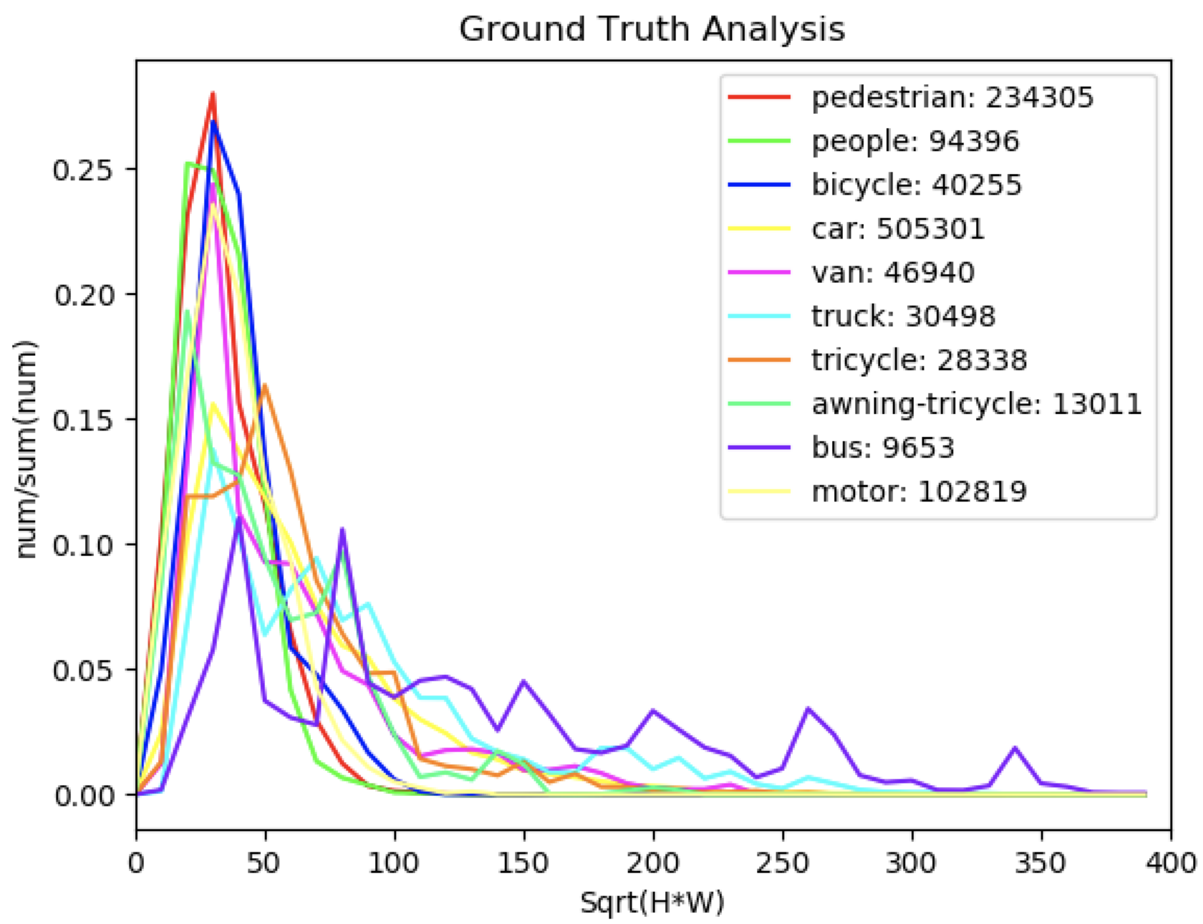
VisDrone Challenge Overview

Object Detection in Videos

- 96 sequences:
  - 56 video sequences for training (24,201 frames)
  - 7 video sequences for validation (2,819 frames)
  - 33 video sequences for testing (12,968 frame): 17 test-dev + 16 test-challenge
- 10 categories:
  - *pedestrian*, *person*, *car*, *van*, *bus*, *truck*, *motor*, *bicycle*, *awning-tricycle*, *tricycle*.
- 2 kinds of useful annotations:
  - occlusion ratio and truncation ratio.

- Evaluation metric:
  - Frame-wise, similar with MS COCO, AP, AP_50,AP_75,…,AR_max=500

Dataset Analysis



From the visualization of box annotations, we learn that:

For object shapes:

- Small objects
- Varying object ratios

For object appearance:

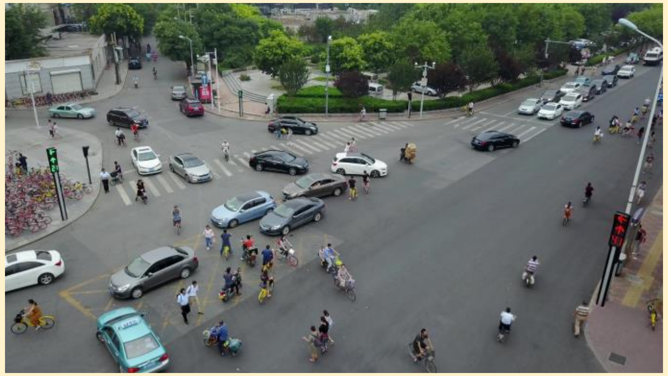
- Little deteriorated objects (motion blur, video defocus, part occlusion and rare poses [1])
- High quality

[1] Xizhou zhu et al. Flow-Guided Feature Aggregation for Video Object Detection. CVPR2017

VisDrone videos vs VID dataset

Videos:

a:
Visdrone video

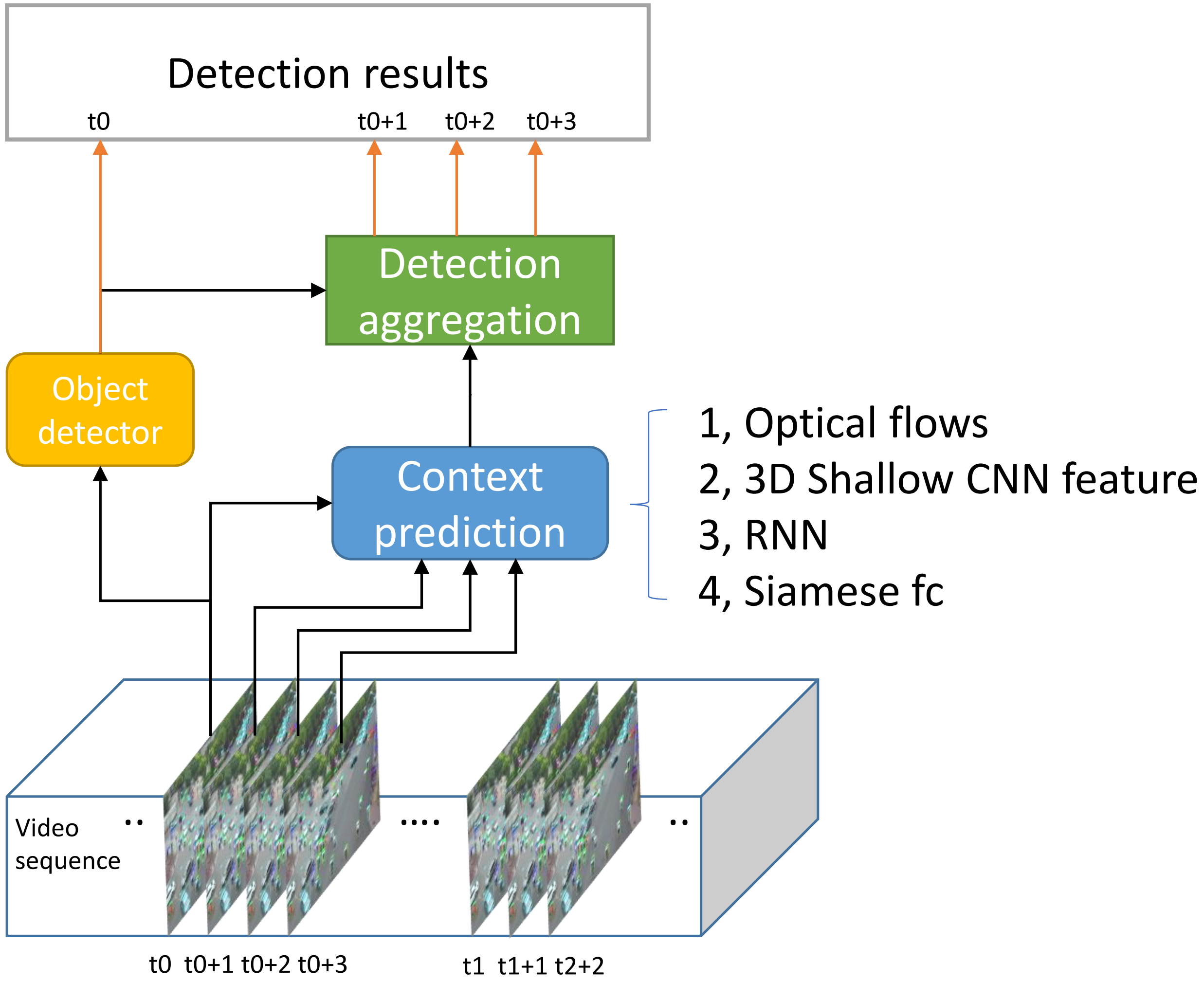


b:
ILSVRC2015 video

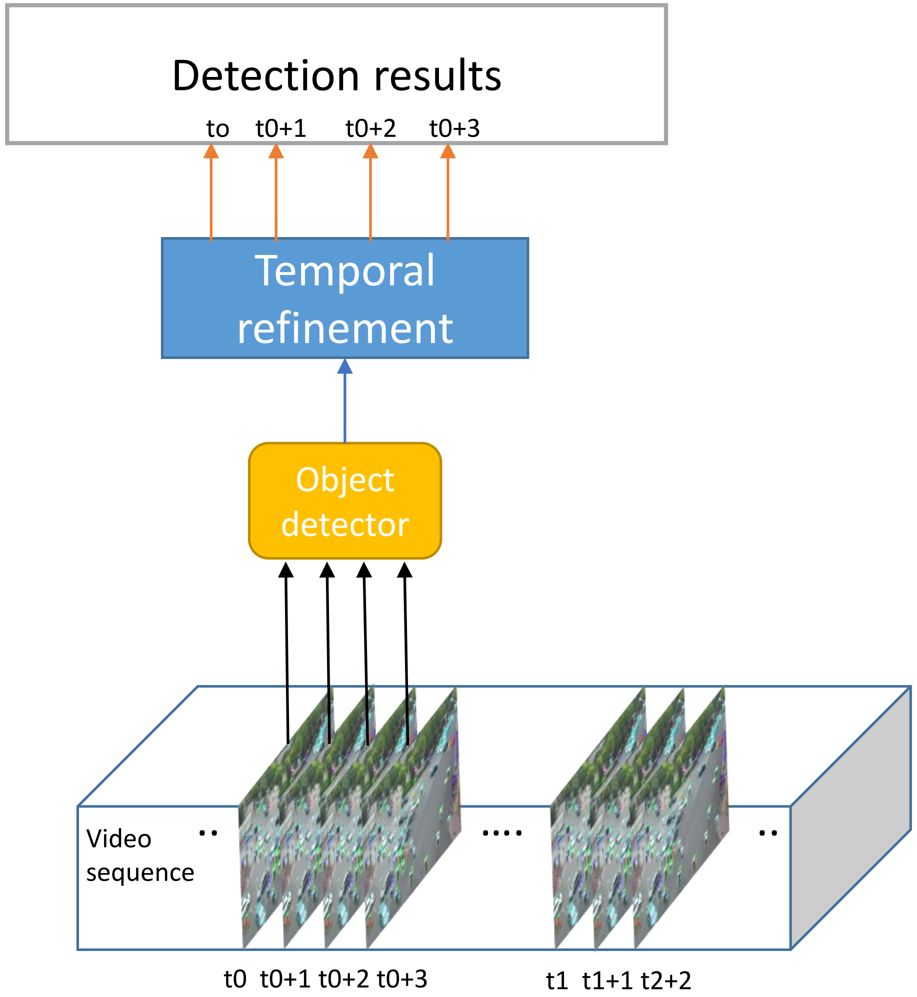
Images (a):

Images (b):

Object detection from videos



(1) clip-level detection

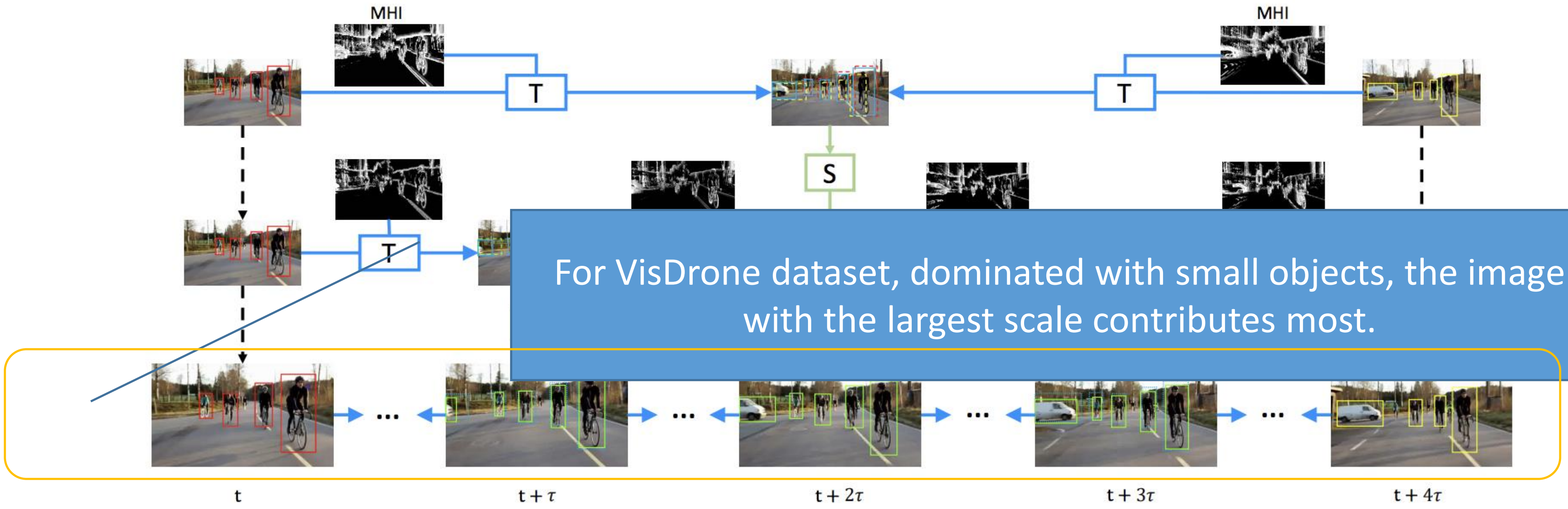


(1) frame-level detection

Object Detection from Videos

(1) clip-level detection:

- Scale-Time lattice[1]



For VisDrone dataset, dominated with small objects, the image with the largest scale contributes most.

Each node represents a **object detection** step; (base detector is Faster R-CNN with ResNet-101)

ue edge represents a **mporal propagation step**;

Green edge represents a **spatial propagation step**

[1] Kai Chen et al. Optimizing Video Object Detection via a Scale-Time Lattice. CVPR2018

Object Detection from Videos

(2) frame-level detection:

- CFENet

Based on the original fast one-stage detector – SSD.
About the inference speed of VGG-SSD300(22ms in original paper, but optimized to be faster):
(a) Mxnet: 15.6 ms [1]
(b) PyTorch: 12.0 ms [2]

We focus on improving it by mainly **enhancing small objects detection** with only **sacrifice little efficiency**. (keep the fast speed)

[1] https://github.com/zhreshold/mxnet-ssd
[2] Songtao Liu et al. Receptive Field Block Net for Accurate and Fast Object Detection. ECCV2018

CFENet

- How to improve SSD / construct a stronger single-shot detector:
  - (1) U-shape modules. E.g., DSSD, RefineDet, RetinaNet.
  - (2) Additional modules. E.g., rfbnet, pyramidbox.
  - (3) loss functions. E.g., IoU loss(Unitbox), RetinaNet(focal loss).

…

CFENet

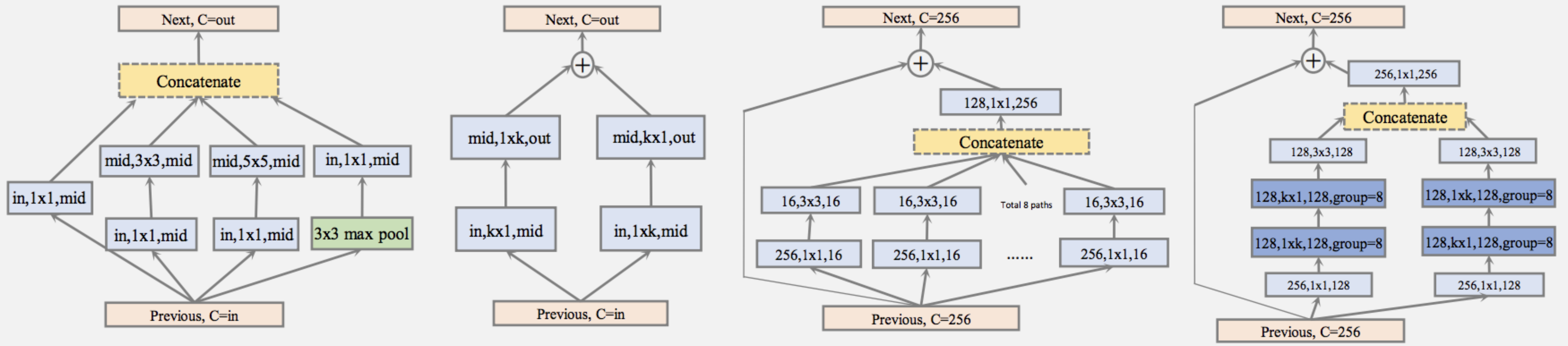
(1) Construct the Comprehensive Feature Enhancement module:

- Receptive field
- 3x3 conv followed by a 1x1 conv & group convolution
- Feature fusion
- Deformable? Non-local?
- Attention?

(2) Assemble CFE modules for SSD

- Where can it benefit mostly? ---- adding modules will definitely **bring accuracy promotion**, but also **increase inference time** and **make training process burdensome**.

CFE module

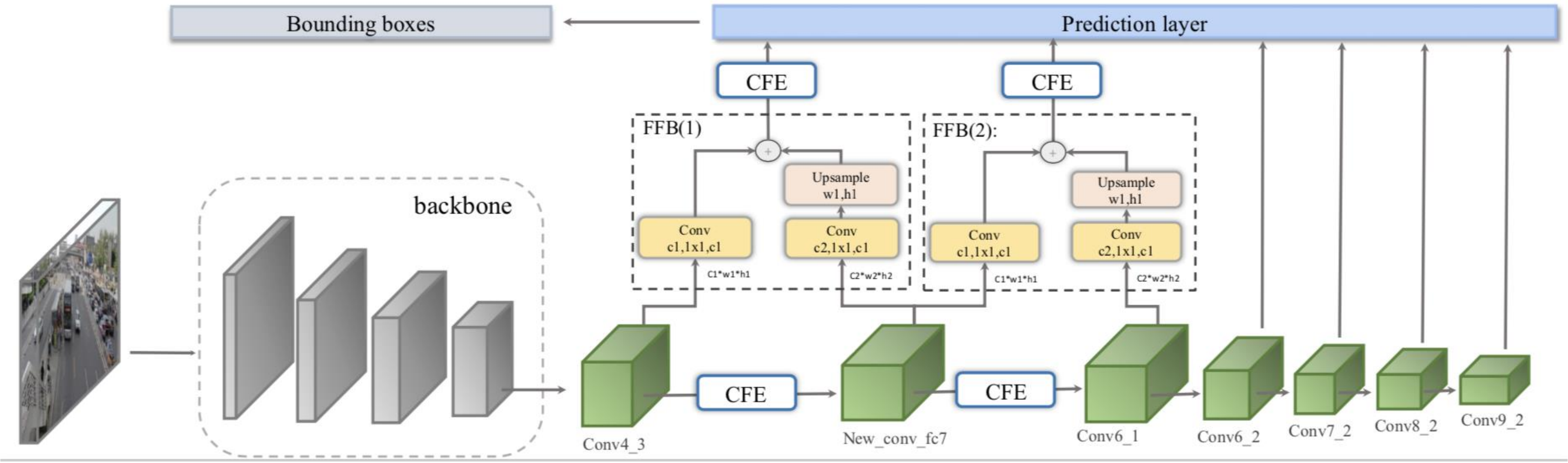


(a) Inception module (b) Large separable module (c) ResNeXt building block (d) Our CFE module

- Receptive field -> Large kernel, i.e., 1xk and kx1, k=7.
- 3x3 conv followed by a 1x1 conv & group convolution -> Group=8.
- Feature fusion -> Residual learning, Symmetric learning.

CFENet

- Assemble CFE modules at detection branches as well as the main path.



Notably, the six feature maps share 2,3,2,2,2,2 CFE modules, respectively.

Experiments

- 4*Titan X, CUDA9.1, cudnn v1.7.5, PyTorch v0.4.0.
- Dataset: MS-COCO
- Baseline: VGG-SSD512

Table 2. Ablation study of CFENet on MSCOCO.

+2 Incep(T)		√			
+2 CFE(T)			√	√	√
+2 CFE(B)				√	√
+2 FFB					√
mAP	28.8	30.3	31.7	33.9	**34.8**

Further, we use resnet101 as backbone and realize a State-of-the-art results on COCO test-dev split.

Experiment

- 4*Titan X, CUDA9.1, cudnn v1.7.5, PyTorch v0.4.0.

- Dataset: VisDrone video, validation set.

model	backbone	Size	Multi-scale	Speed	AP	AP_50	AP_75	AR,M=1	AR,M=10	AR,M=100	AR,M=500
CFENet	VGG	800	False	23 FPS	15.5	34.1	11.8	7.45	19.8	26.1	26.1
CFENet	VGG	800	True	1 FPS	22.3	44.9	18.5	12.0	31.5	42.5	45.2

In addition, we have used Seq-NMS as a temporal refinement step. However, it drops the accuracy a little. For small objects detection in videos, temporal refinement still have a long way to go.

Speed

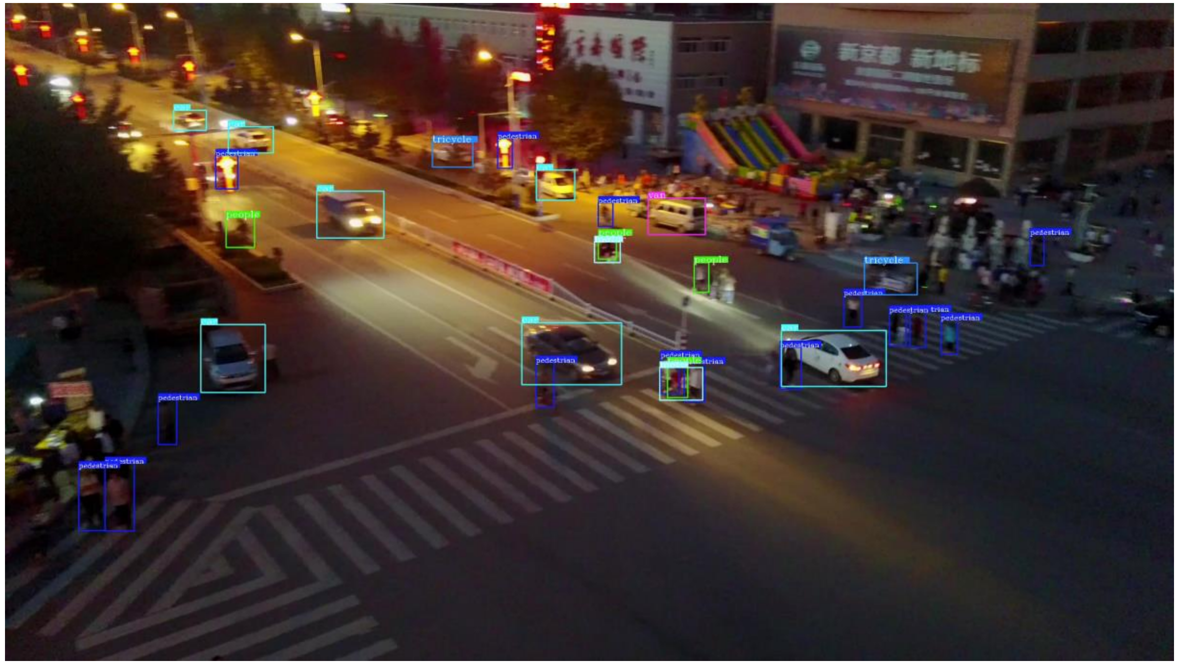
- For one-stage detectors, densely post process (NMS) will cost a lot of time. Tuning score threshold and nms algorithm will help improve efficiency.

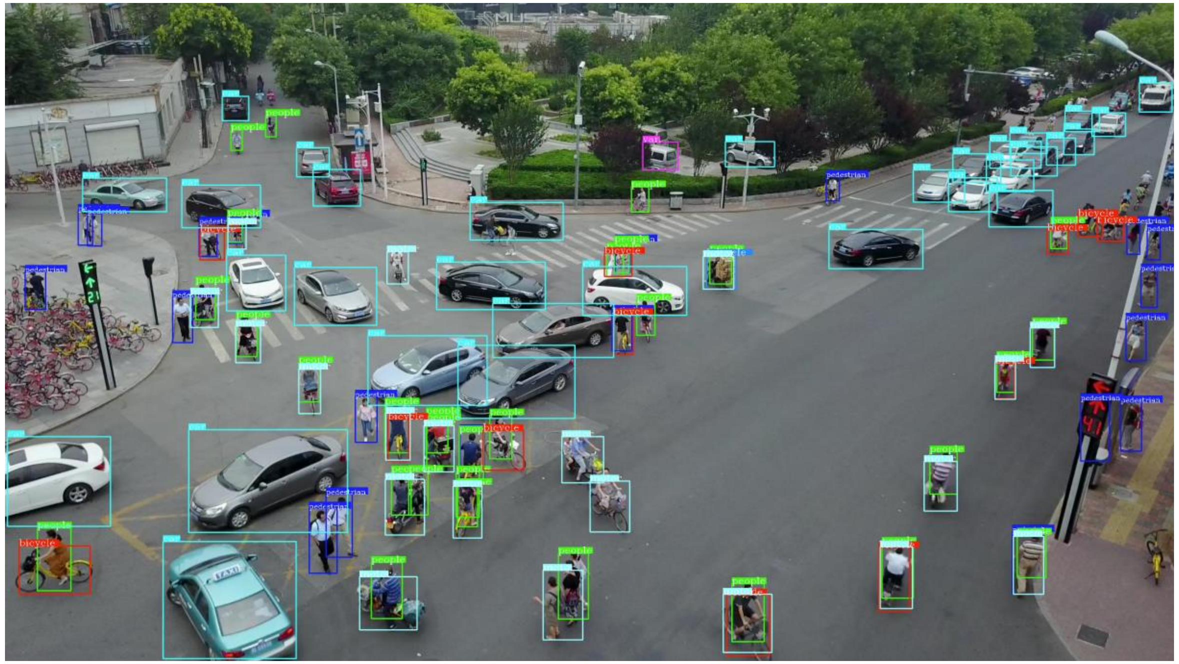
for VGG-CFENet800:

Model	Score	NMS	CNN time	NMS time	AP
CFENet	0.01	Hard	13 ms	7 ms	15.5
CFENet	0.05	Hard	13 ms	3 ms	15.2
CFENet	0.01	Soft, linear	13 ms	11 ms	15.9
CFENet	0.05	Soft, linear	13 ms	4 ms	15.5
CFENet	0.05	Soft, gaussian	13 ms	8 ms	15.6

Including the time of image preprocessing, VGG-CFENet800 can still achieve 23 fps, a real-time speed.

Visualization





Future work

- We will keep focusing on enhancing detection for small objects in videos with temporal assistance in the future. Specifically, introducing temporal propagation into CFENet, replace VGG with ResNet-101. In theory, such a combination operation can not only make it faster, but also more accurate.

The end

Q&A.