基于YOLO的路况监测系统设计与实现

杨国旭
河南师范大学 计算机科学与技术

2022/7/21

目录

- YOLO
- 行人车辆检测
- 车牌检测
- 路面损伤检测
- 实验
- 测试

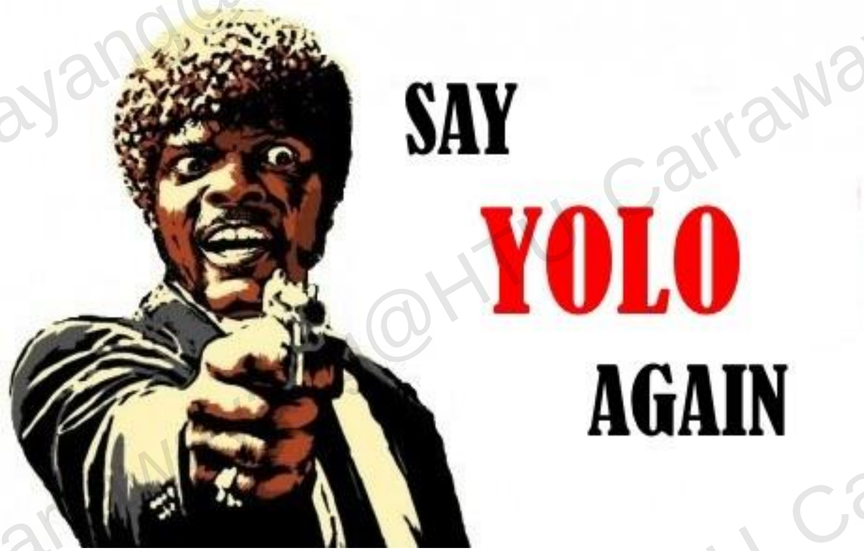
YOLO

YOLO

YOLO

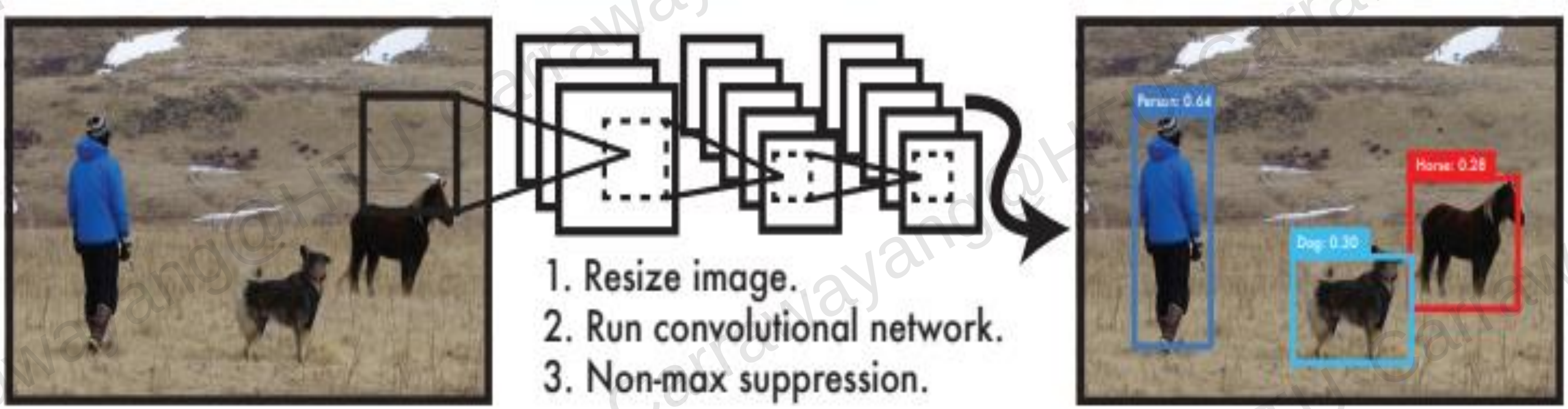
You Only Look Once:
Unified, Real-Time Object Detection

Joseph Redmon



YOLO

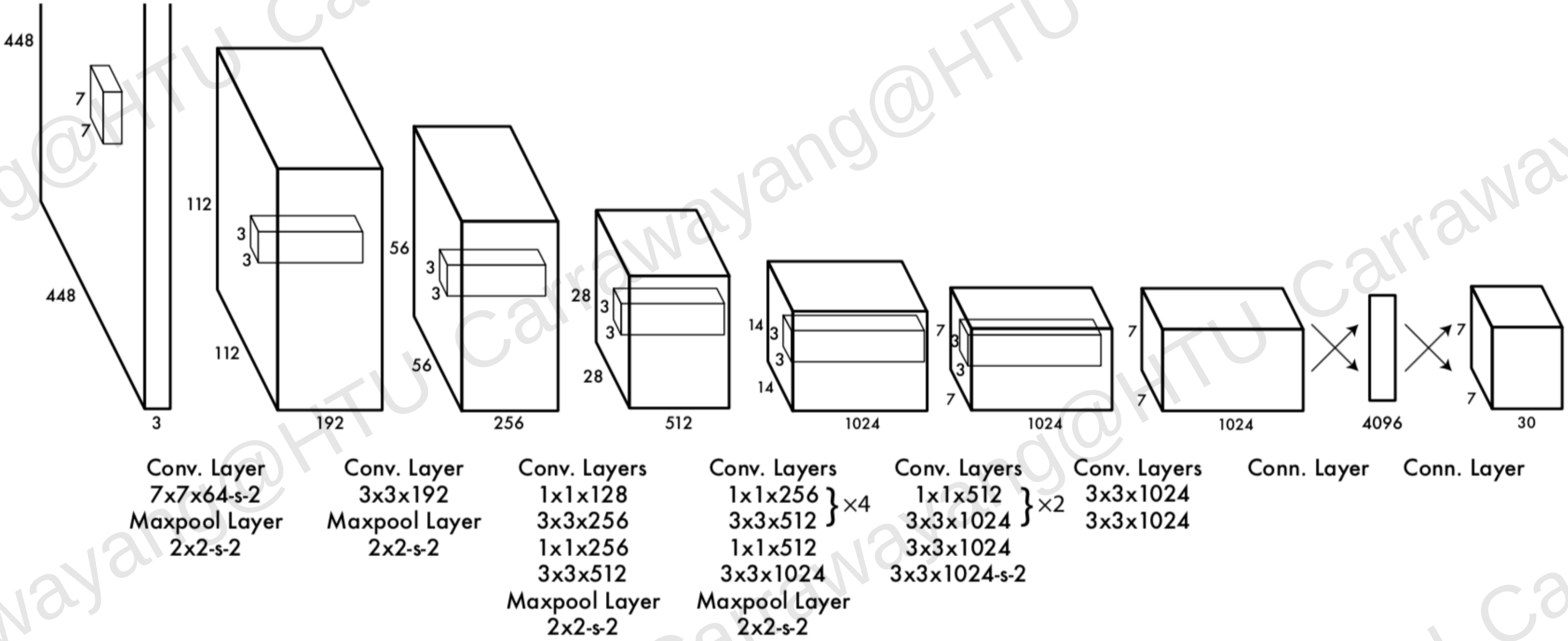
- 将目标检测视为一个回归问题，可以直接进行端到端的处理



1. Resize image.
2. Run convolutional network.
3. Non-max suppression.

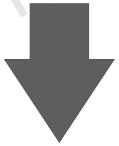
448 X 448

网络推理


448
7
7
448
3
112
3
3
112
192
56
3
3
56
256
28
3
3
28
512
14
3
3
14
1024
7
3
3
7
1024
7
7
1024
4096
7
7
30
Conv. Layer
7x7x64-s-2
Maxpool Layer
2x2-s-2
Conv. Layer
3x3x192
Maxpool Layer
2x2-s-2
Conv. Layers
1x1x128
3x3x256
1x1x256
3x3x512
Maxpool Layer
2x2-s-2
Conv. Layers
1x1x256
3x3x512
×4
1x1x512
3x3x1024
Maxpool Layer
2x2-s-2
Conv. Layers
1x1x512
3x3x1024
×2
3x3x1024
3x3x1024-s-2
Conv. Layers
3x3x1024
3x3x1024
Conn. Layer
Conn. Layer

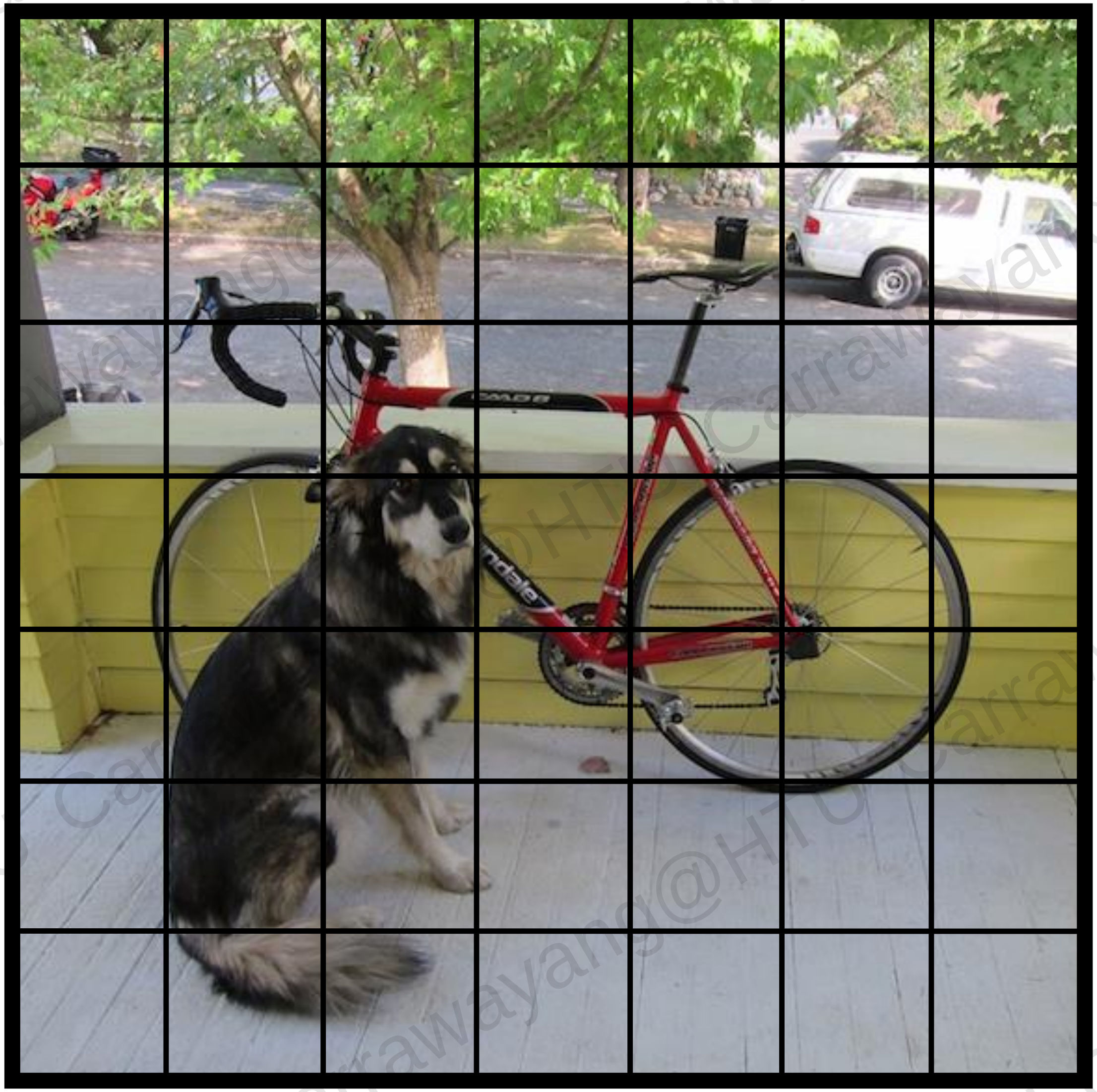
YOLO

将图片划分为7x7网格



每一个网格cell输出
两个bounding box

$$\begin{cases} p_{conf}, x, y, w, h \\ p_{conf}, x, y, w, h \\ p_{c_1}, p_{c_2}, \cdots, p_{c_{20}} \end{cases}$$



YOLO

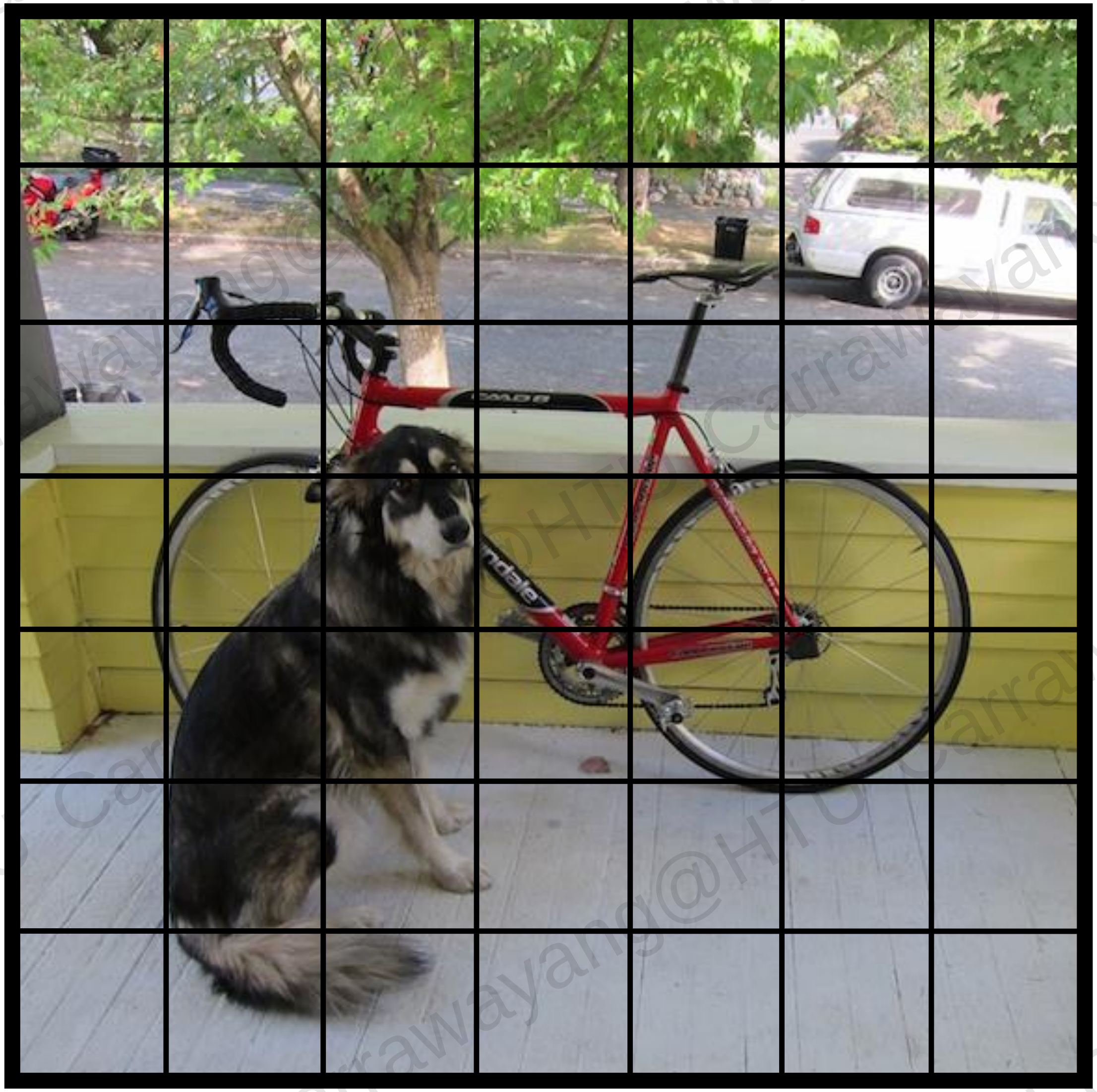
将图片划分为7x7网格

每一个网格cell输出
两个bounding box

$$\left\{\begin{array}{ll}\boxed{p_{conf}, x, y, w, h} & \text{predictor1}\\ p_{conf}, x, y, w, h & \text{predictor2}\\ \boxed{p_{c_1}, p_{c_2}, \cdots, p_{c_{20}}} & \end{array}\right.$$

shared class prob

一个cell对应30个输出



YOLO

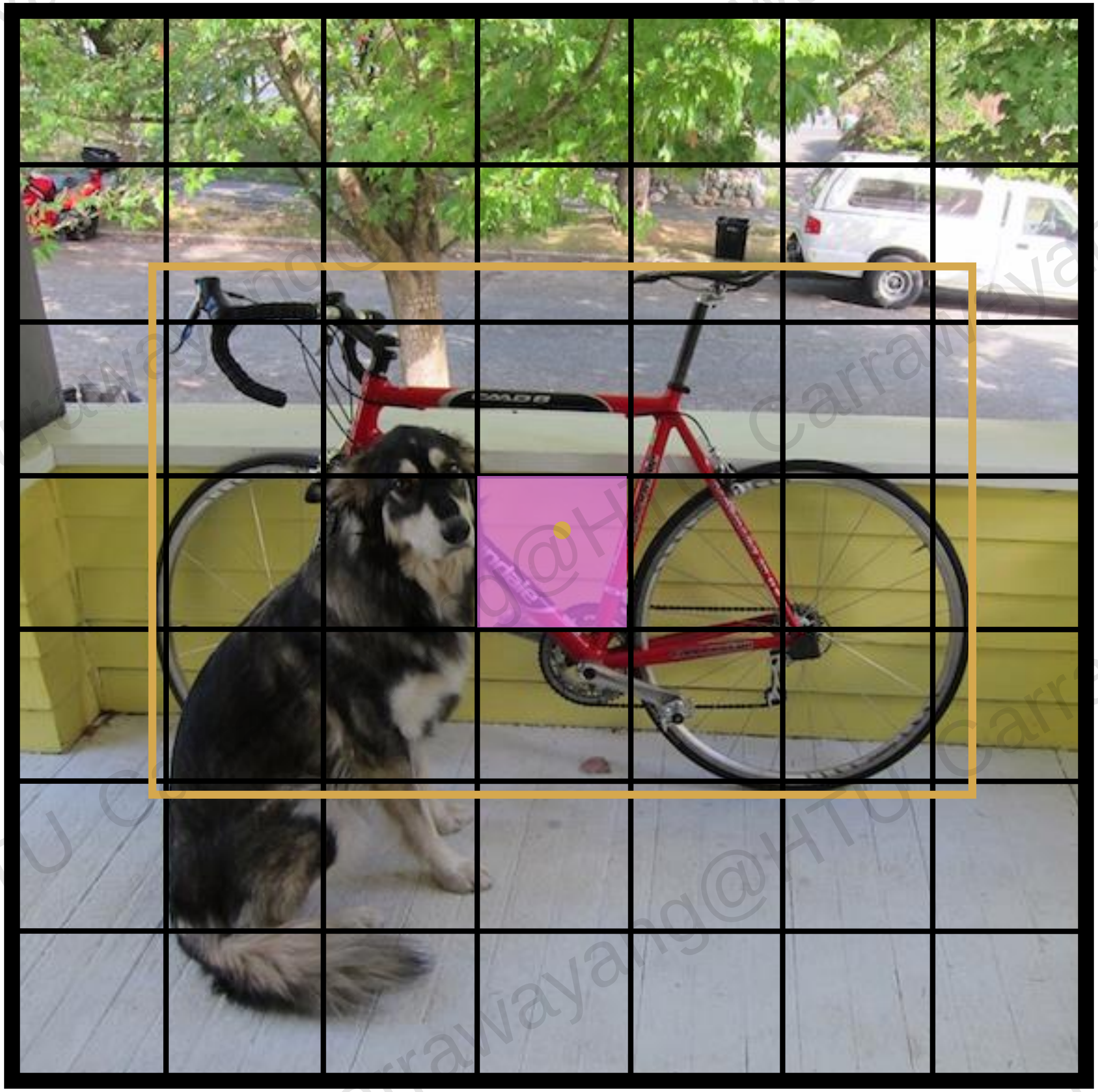
将图片划分为7x7网格

每一个网格cell输出
两个bounding box

$$\begin{cases} p_{conf}, x, y, w, h \\ p_{conf}, x, y, w, h \\ p_{c_1}, p_{c_2}, \cdots, p_{c_{20}} \end{cases}$$

Ground Truth

1.Object属于其center
所在的cell



YOLOv1

将图片划分为7x7网格

↓

每一个网格cell输出
两个bounding box

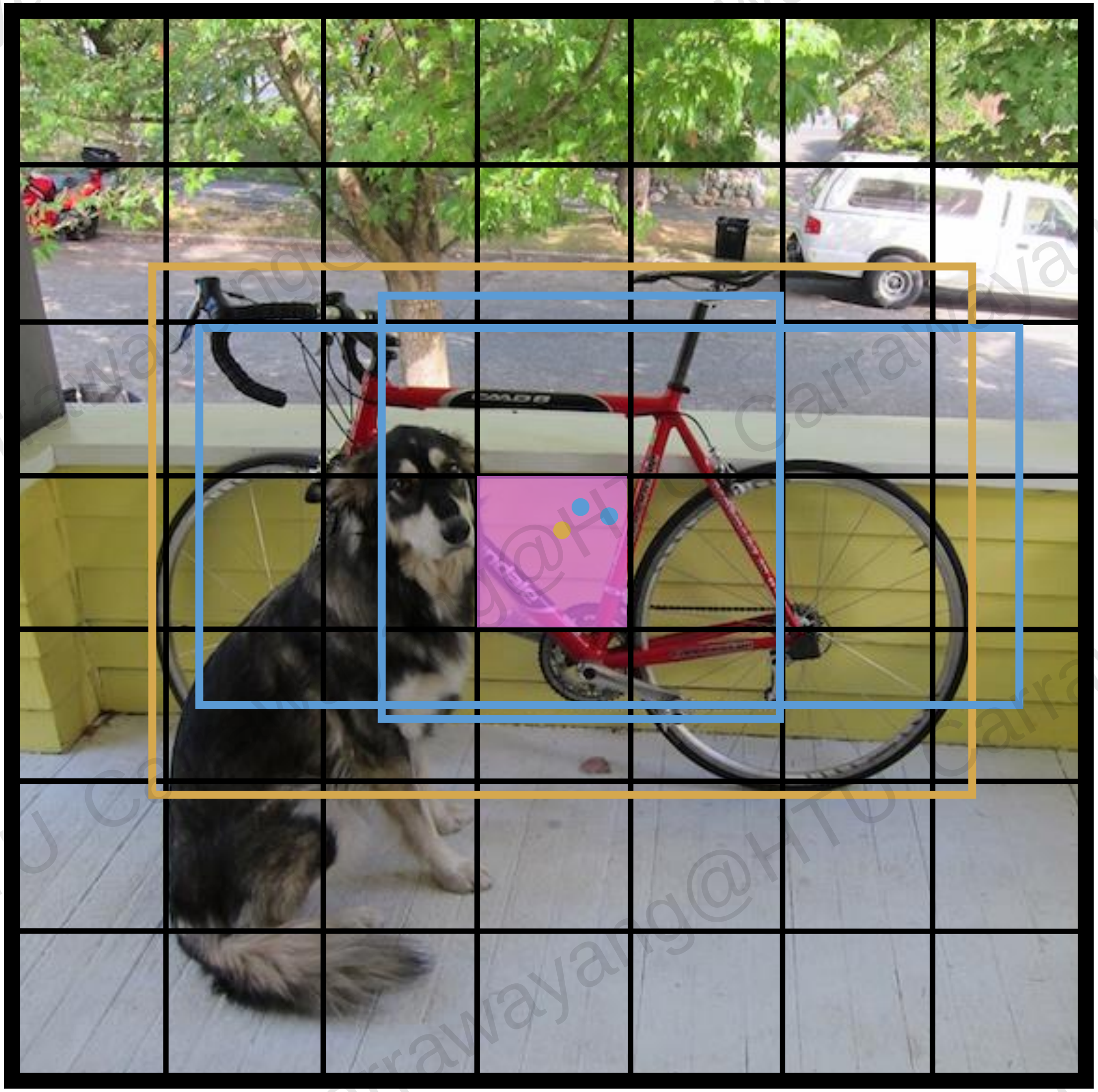
$$\begin{cases} p_{conf}, x, y, w, h \\ p_{conf}, x, y, w, h \\ p_{c_1}, p_{c_2}, \cdots, p_{c_{20}} \end{cases}$$

↓

Ground Truth

1.Object属于其center
所在的cell

2. Object属于与predictor的
预测结果
IOU最大的predictor



YOLOv1

将图片划分为7x7网格

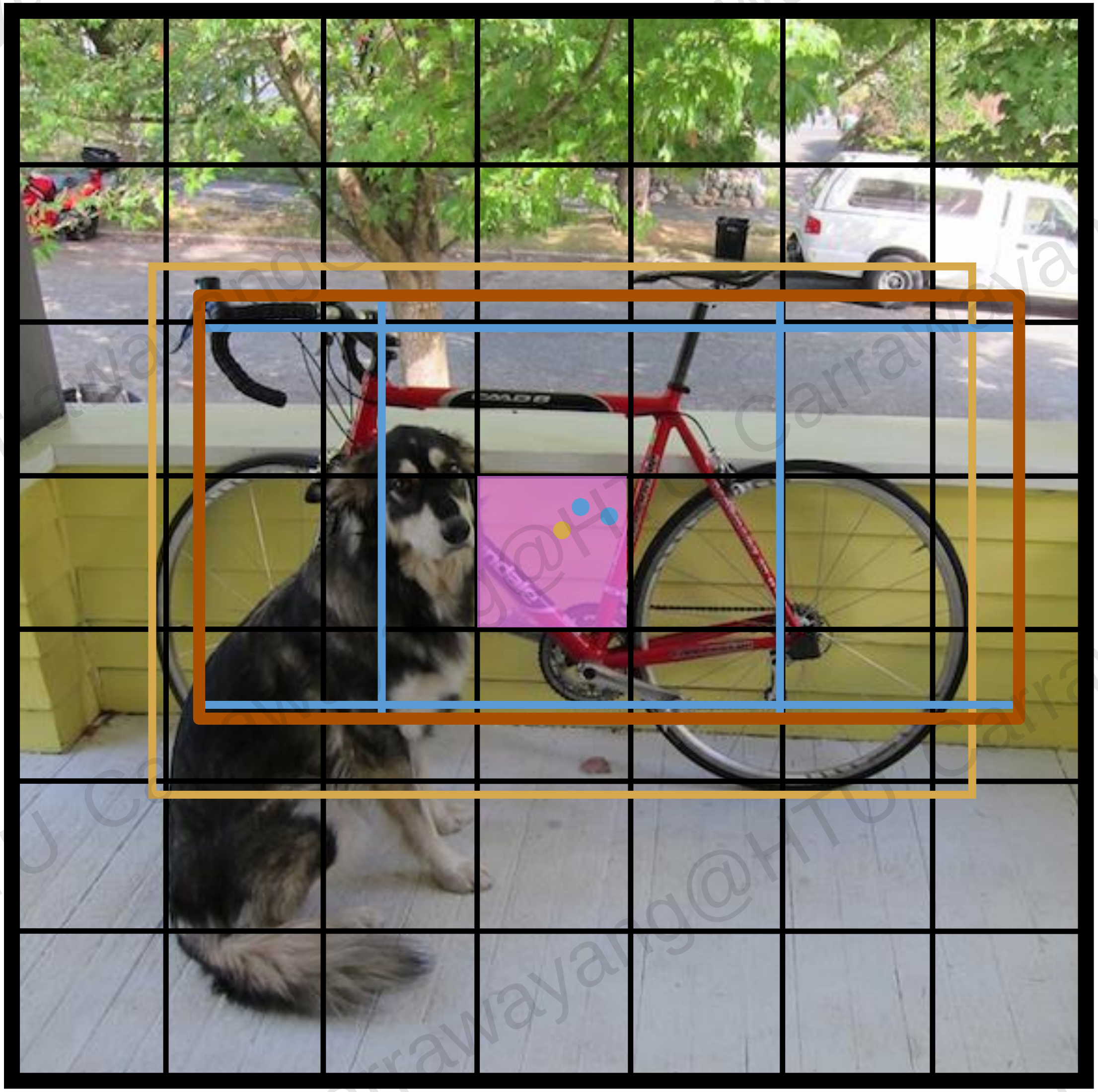
每一个网格cell输出
两个bounding box

$$\begin{cases} p_{conf}, x, y, w, h \\ p_{conf}, x, y, w, h \\ p_{c_1}, p_{c_2}, \cdots, p_{c_{20}} \end{cases}$$

Ground Truth

1.Object属于其center
所在的cell

2. Object属于与predictor的
预测结果
IOU最大的predictor



Loss

位置损失

$$\lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

$$+ \lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

置信度损失

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2$$

$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2$$

类别损失

$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$

$$\mathbb{1}_{ij}^{obj} = \begin{cases} 1 & \text{, the } j\text{th bb in cell } i \text{ is “responsible” for prediction} \\ 0 & \text{, otherwise} \end{cases}$$

位置损失

S=3, B=1

位置损失:
$$\sum_{i=0}^{S^2}\sum_{j=0}^{B}\boxed{\mathbb{1}_{ij}^{obj}}\left[(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2+(w_i-\hat{w}_i)^2+\left(h_i-\hat{h}_i\right)^2\right]$$

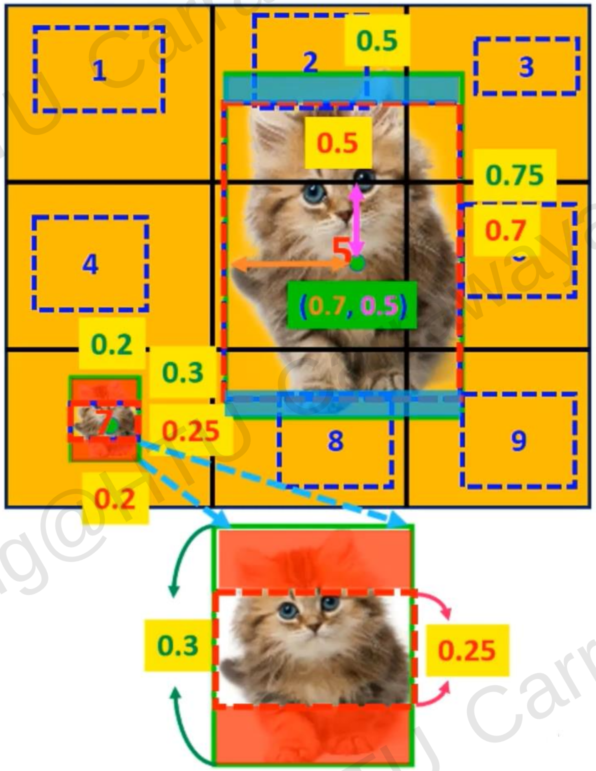
$$\mathbb{1}_{ij}^{obj}=\begin{cases}1 & \text{, the } j\text{th bb in cell } i \text{ is “responsible” for prediction}\\ 0 & \text{, otherwise}\end{cases}$$

$$\mathbb{1}_{51}^{obj}\times((0.7-0.7)^2+(0.5-0.5)^2+(0.5-0.5)^2+\boxed{(0.75-0.7)^2)}$$
$$+\mathbb{1}_{71}^{obj}\times((0.6-0.6)^2+(0.5-0.5)^2+(0.2-0.2)^2+\boxed{(0.3-0.25)^2)}$$
$$=\mathbb{1}_{51}^{obj}\times(0.05)^2+\mathbb{1}_{71}^{obj}\times(0.05)^2$$

$$\lambda_{\textbf{coord}}\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{\text{obj}}\left[(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2\right]$$

$$+\lambda_{\textbf{coord}}\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{\text{obj}}\left[\left(\sqrt{w_i}-\sqrt{\hat{w}_i}\right)^2+\left(\sqrt{h_i}-\sqrt{\hat{h}_i}\right)^2\right]$$

$$\mathbb{1}_{51}^{obj}\times\Big((0.7-0.7)^2+(0.5-0.5)^2+\left(\sqrt{0.5}-\sqrt{0.5}\right)^2+\boxed{\left(\sqrt{0.75}-\sqrt{0.7}\right)^2}$$
$$+\mathbb{1}_{71}^{obj}\times\Big((0.6-0.6)^2+(0.5-0.5)^2+\left(\sqrt{0.2}-\sqrt{0.2}\right)^2+\boxed{\left(\sqrt{0.3}-\sqrt{0.25}\right)^2}$$
$$=\mathbb{1}_{51}^{obj}\times(0.03)^2+\mathbb{1}_{71}^{obj}\times(0.048)^2$$



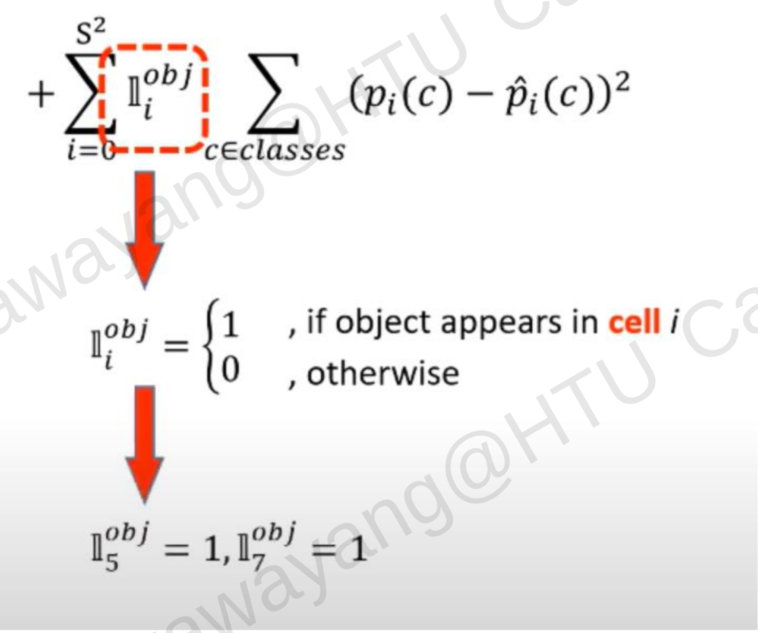
类别损失

$= \mathrm{Pr}(\mathrm{Class}_i|\mathrm{Object})$

类别损失：

$$+\sum_{i=0}^{S^2}\mathbb{1}_i^{\text{obj}}\sum_{c\in\text{classes}}\left(p_i(c)-\boxed{\hat{p}_i(c)}\right)^2$$

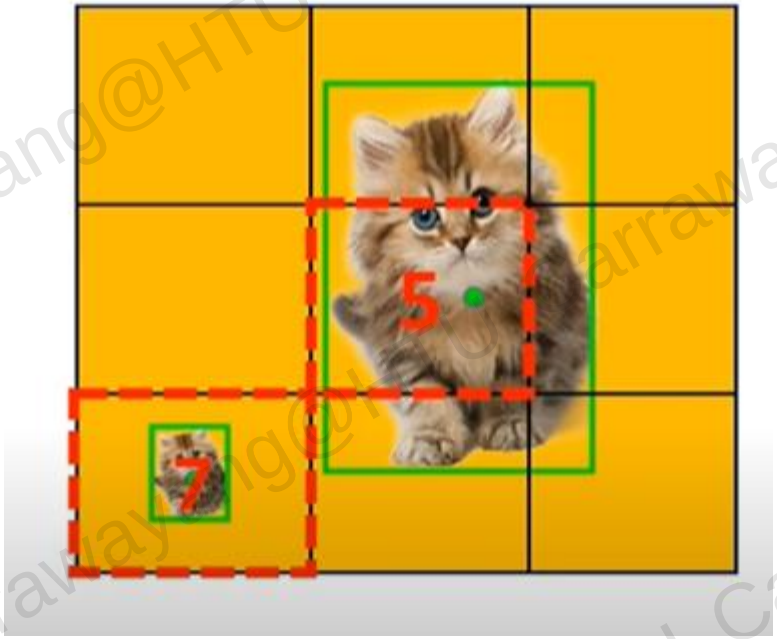
$$\mathrm{Pr}(\mathrm{Class}_i|\mathrm{Object})*\mathrm{Pr}(\mathrm{Object})*\mathrm{IOU}_{\text{pred}}^{\text{truth}}=\mathrm{Pr}(\mathrm{Class}_i)*\mathrm{IOU}_{\text{pred}}^{\text{truth}}$$



$$+\sum_{i=0}^{S^2}\boxed{\mathbb{1}_i^{obj}}\sum_{c\in classes}(p_i(c)-\hat{p}_i(c))^2$$

$$\mathbb{1}_i^{obj}=\begin{cases}1 & \text{, if object appears in cell } i\\ 0 & \text{, otherwise}\end{cases}$$

$$\mathbb{1}_5^{obj}=1,\mathbb{1}_7^{obj}=1$$



置信度损失

$$= \Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}}$$

置信度损失：

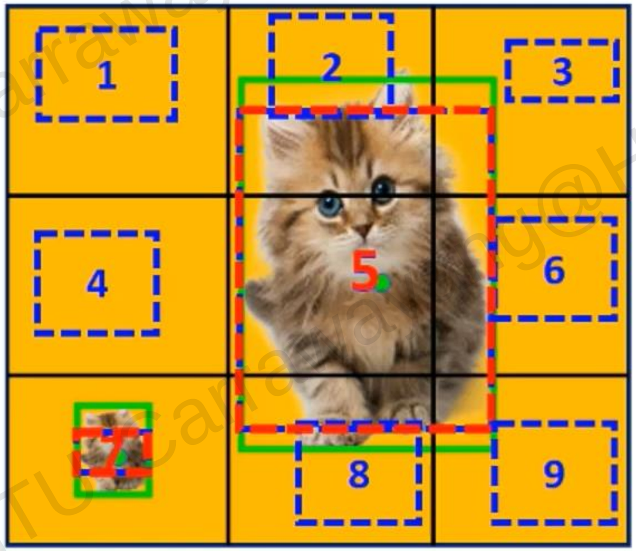
$$+\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{\text{obj}}\left(C_i-\boxed{\hat{C}_i}\right)^2$$

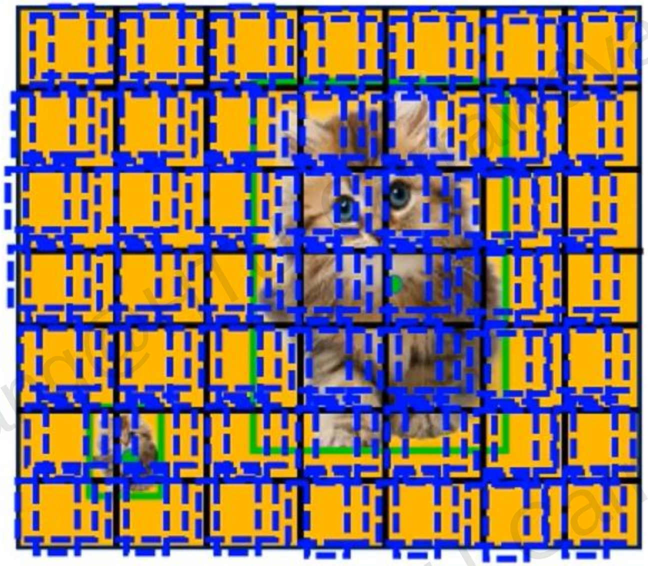
$$+\lambda_{\text{noobj}}\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{\text{noobj}}\left(C_i-\hat{C}_i\right)^2$$

i	1	2	3	4	5	6	7	8	9
C_i	0	0	0	0	1	0	1	0	0
$\hat{C}_i$	0.1	0.1	0.1	0.1	0.6	0.1	0.6	0.1	0.1
$\lvert C_i-\hat{C}_i\rvert$	0.1	0.1	0.1	0.1	0.4	0.1	0.4	0.1	0.1

S=3, B=1 $2\times(0.4)^2+7\times(0.1)^2 = 0.32+0.07$

S=7, B=2 $2\times(0.4)^2+96\times(0.1)^2 = 0.32+0.96$





Loss

位置损失

$$\lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

$$+ \lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

置信度损失

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2$$

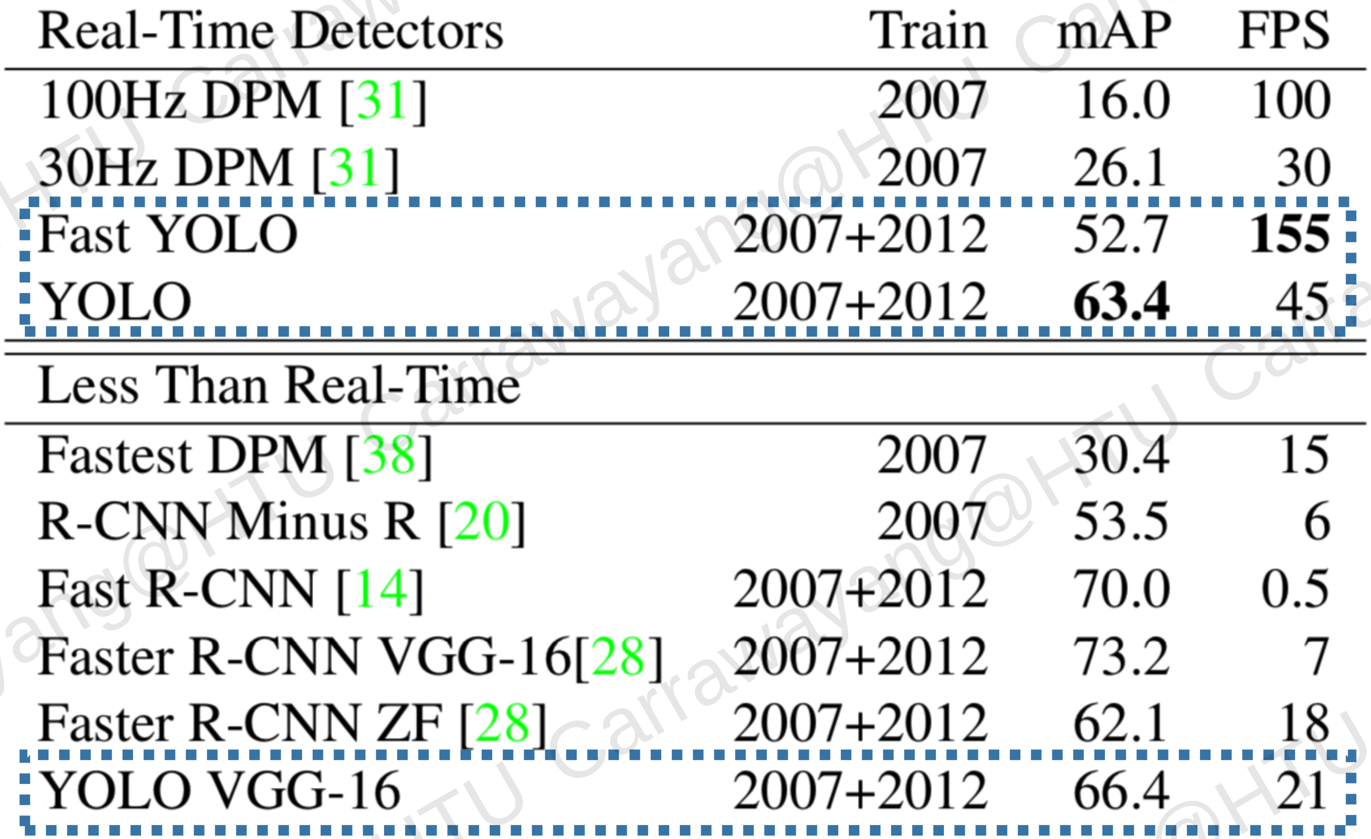
$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2$$

类别损失

$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$

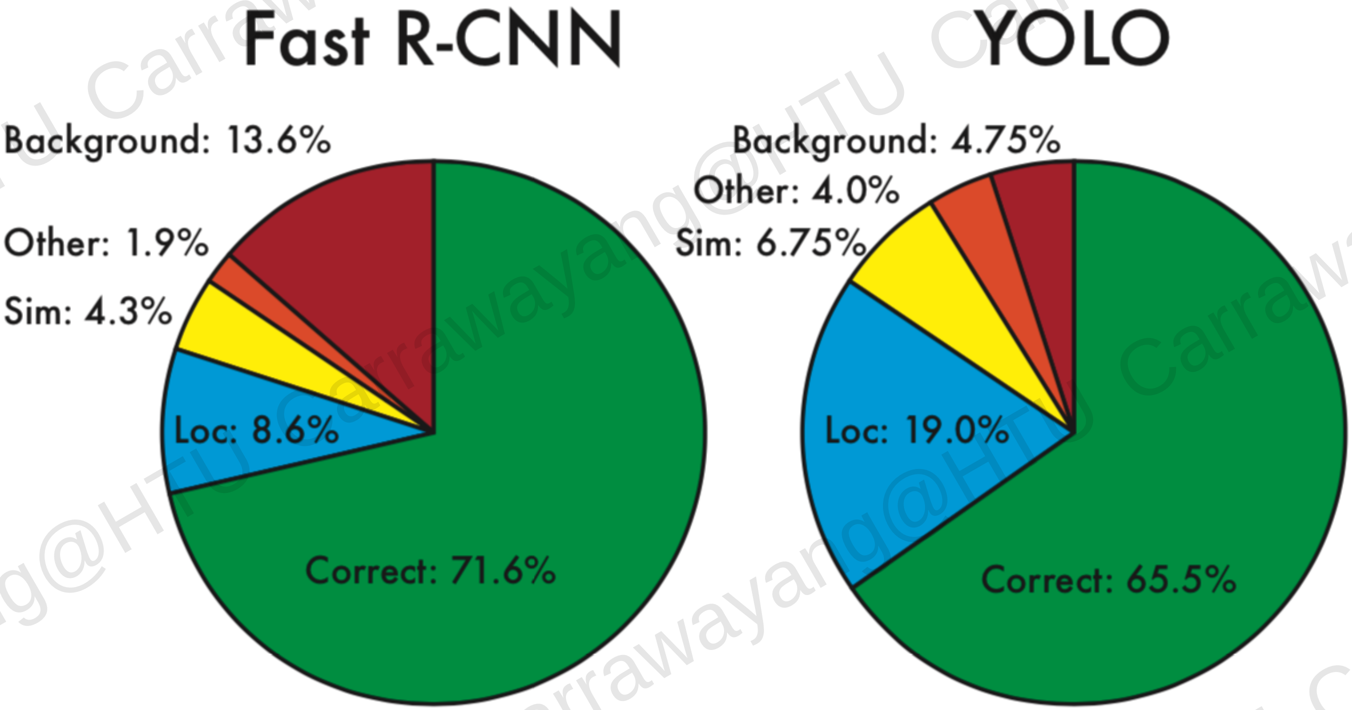
$$\mathbb{1}_{ij}^{obj} = \begin{cases} 1 & \text{, the } j\text{th bb in cell } i \text{ is “responsible” for prediction} \\ 0 & \text{, otherwise} \end{cases}$$

实验



Real-Time Detectors	Train	mAP	FPS
100Hz DPM [31]	2007	16.0	100
30Hz DPM [31]	2007	26.1	30
Fast YOLO	2007+2012	52.7	**155**
YOLO	2007+2012	**63.4**	45
Less Than Real-Time			
Fastest DPM [38]	2007	30.4	15
R-CNN Minus R [20]	2007	53.5	6
Fast R-CNN [14]	2007+2012	70.0	0.5
Faster R-CNN VGG-16[28]	2007+2012	73.2	7
Faster R-CNN ZF [28]	2007+2012	62.1	18
YOLO VGG-16	2007+2012	66.4	21

Error Analysis

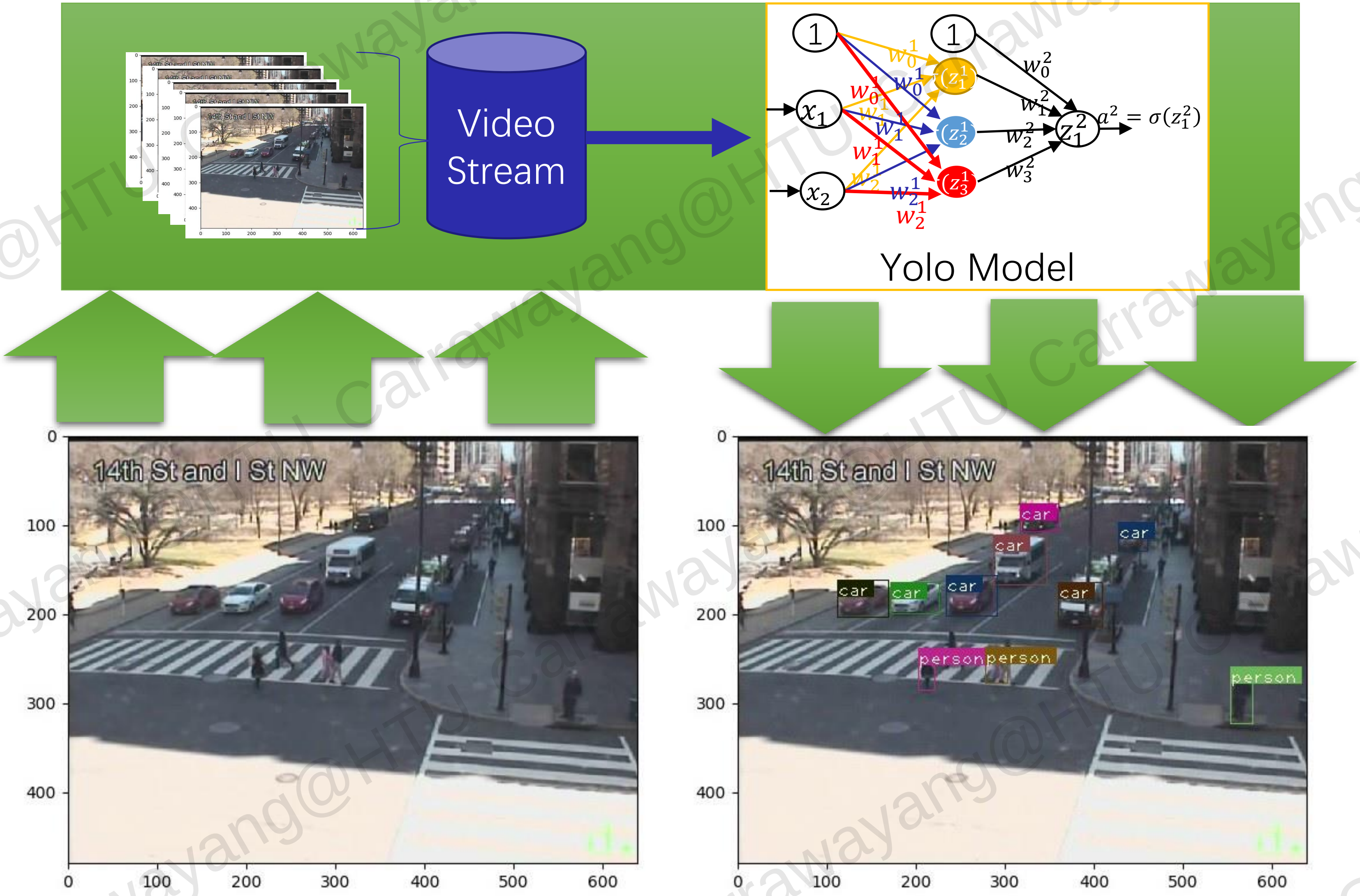


- Correct: correct class and IOU $> .5$
- Localization: correct class, $.1 < \text{IOU} < .5$
- Similar: class is similar, IOU $> .1$
- Other: class is wrong, IOU $> .1$
- Background: IOU $< .1$ for any object

行人车辆检测

- YOLO
- **行人车辆检测**
- 车牌检测
- 路面损伤检测
- 实验
- 测试

行人车辆检测



Real-time video Sequence

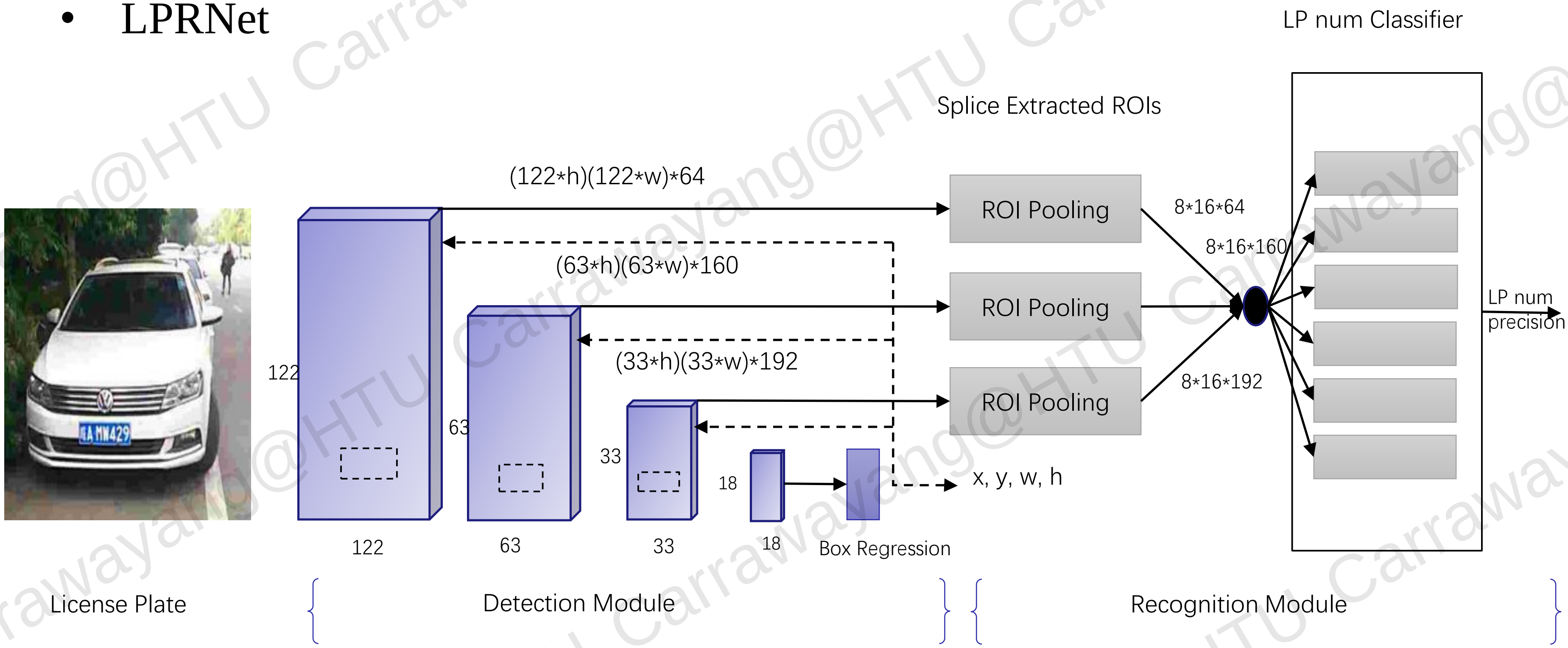
Person-Vehicle Labeling

车牌检测

- YOLO
- 行人车辆检测
- **车牌检测**
- 路面损伤检测
- 实验
- 测试

车牌检测

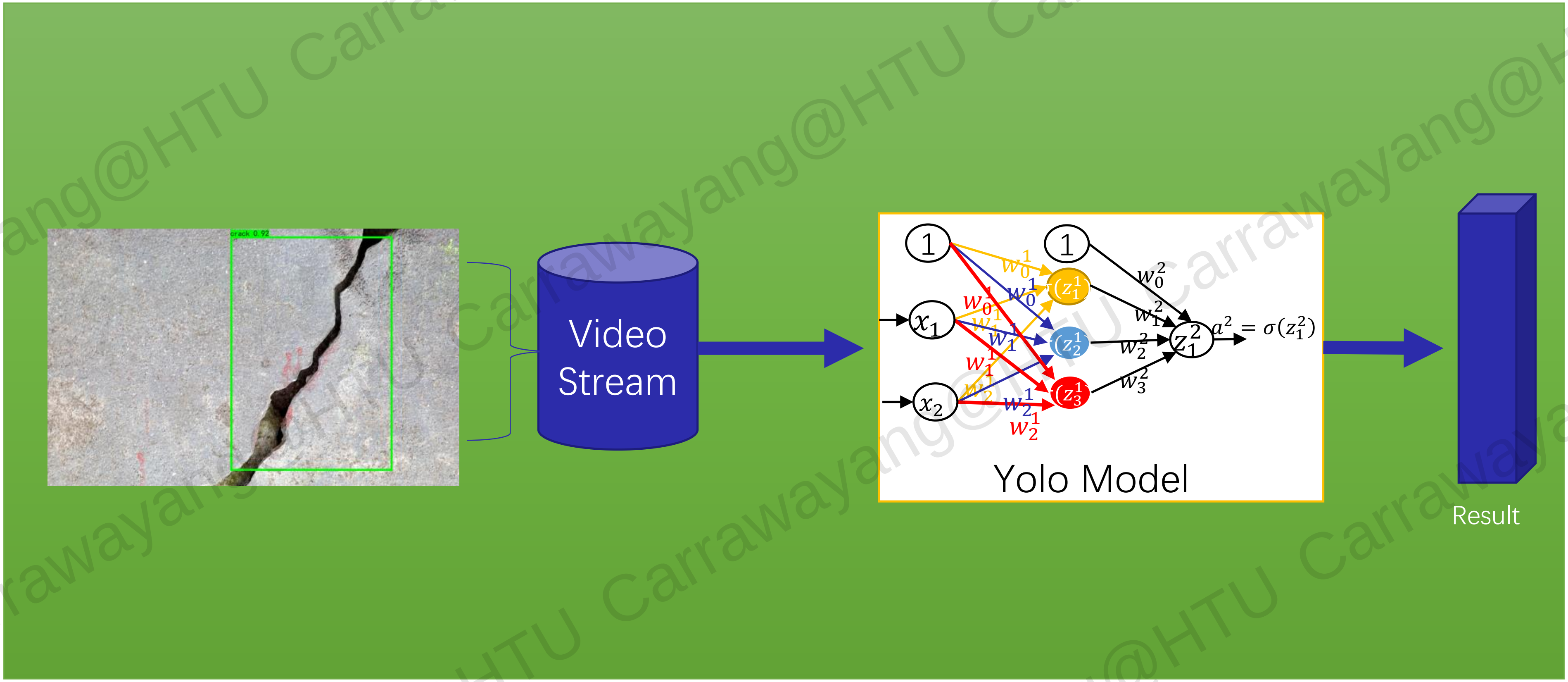
- LPRNet


LP num Classifier
Splice Extracted ROIs
(122*h)(122*w)*64
(63*h)(63*w)*160
(33*h)(33*w)*192
ROI Pooling
ROI Pooling
ROI Pooling
8*16*64
8*16*160
8*16*192
LP num
precision
122
122
63
63
33
33
18
18
x, y, w, h
Box Regression
License Plate
Detection Module
Recognition Module

路面损伤检测

- YOLO
- 行人车辆检测
- 车牌检测
- **路面损伤检测**
- 实验
- 测试

路面损伤检测

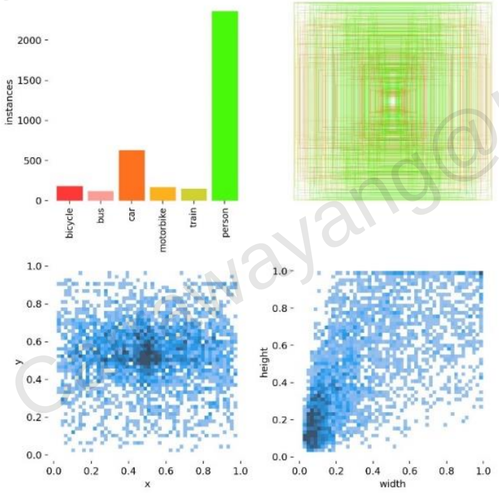

crack 0.92
Video Stream
1
1
x_1
x_2
w_0^1
w_1^1
w_2^1
$\sigma(z_1^1)$
$\sigma(z_2^1)$
$\sigma(z_3^1)$
w_0^2
w_1^2
w_2^2
w_3^2
z_1^2
$a^2 = \sigma(z_1^2)$
Yolo Model
Result

实验

- YOLO
- 行人车辆检测
- 车牌检测
- 路面损伤检测
- **实验**
- 测试

Sub-Pascal VOC2007

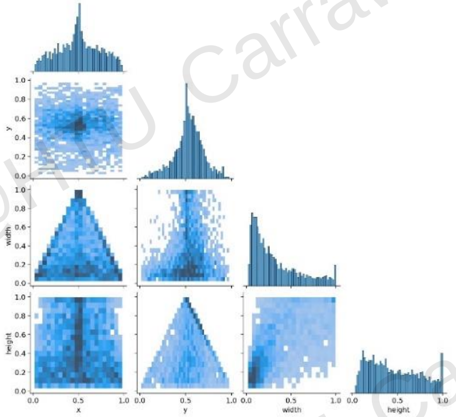
- 设备1：Windows10; Intel(R) I7-8750H; Nvidia-GTX-1050Ti
- 设备2：Ubuntu18.04; Tesla K80



Pascal VOC2007

类别	样例
人	person
动物	bird, cat, cow, dog, horse, sheep
交通工具	aeroplane, bicycle, boat, bus, car, motorbike, train
室内物品	bottle, chair, dining table, potted plant, sofa, tv/monitor

去除数据集中无关数据

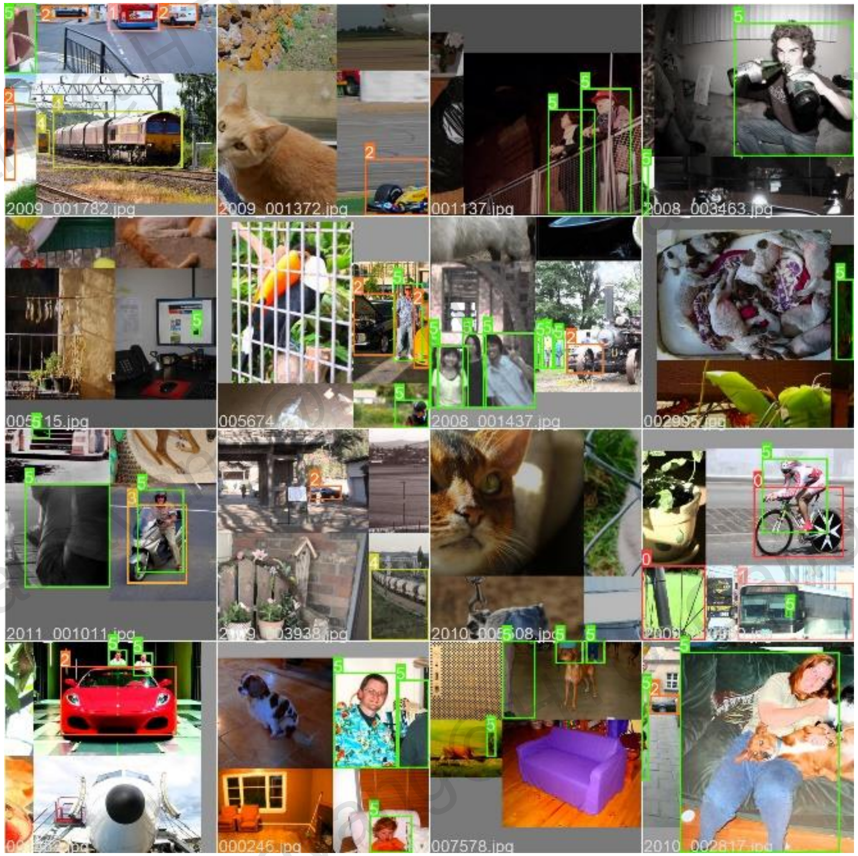


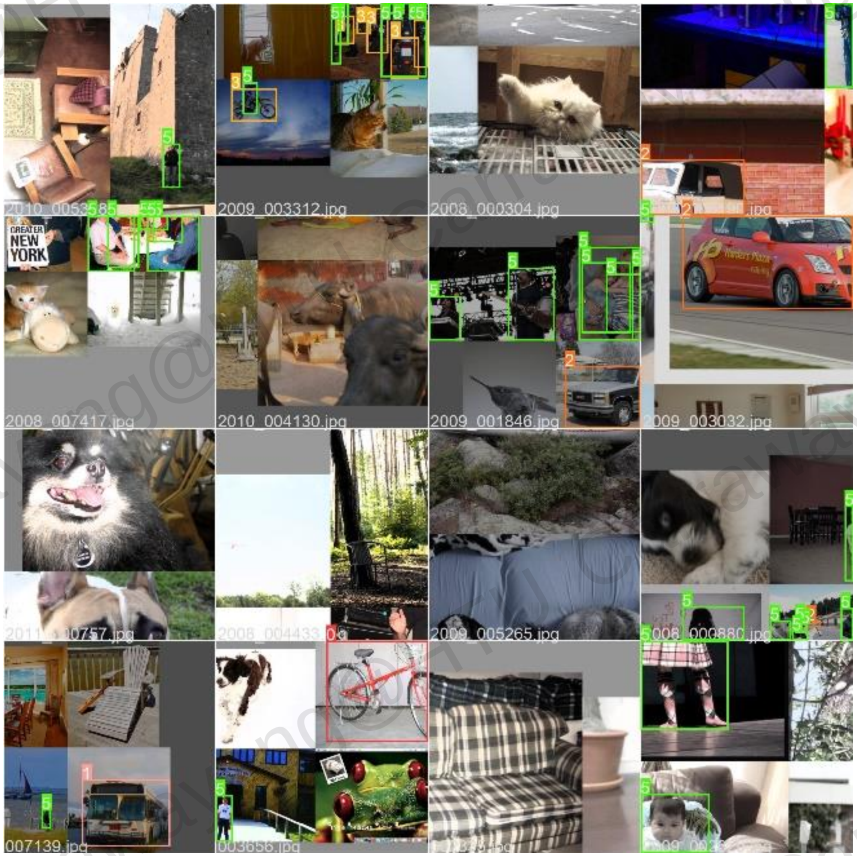
Sub-Pascal VOC2007

类别	样例
人	person
交通工具	bicycle, bus, car, motorbike, train

数据增强

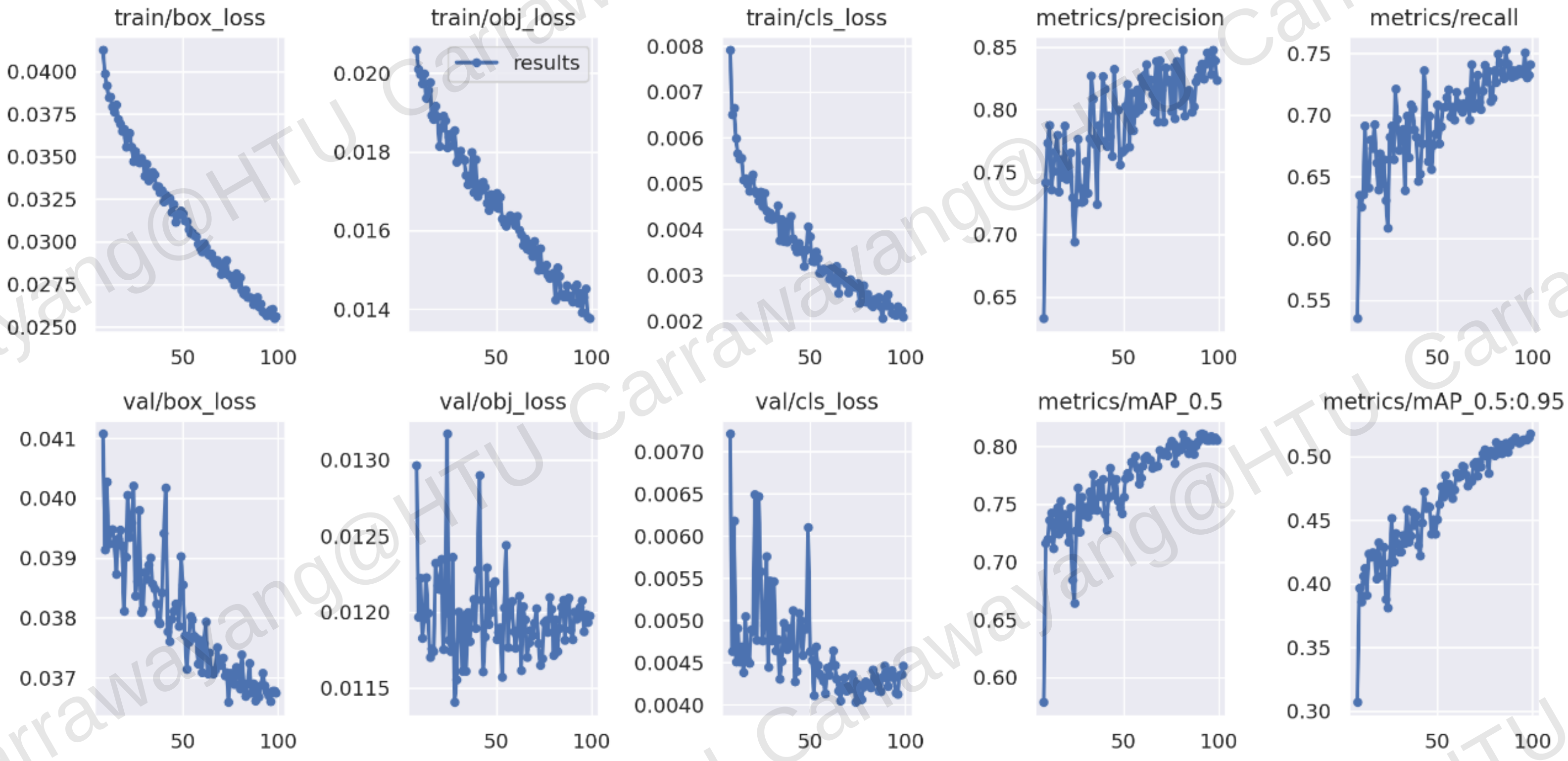
- cutout数据增强后一个Mini Batch



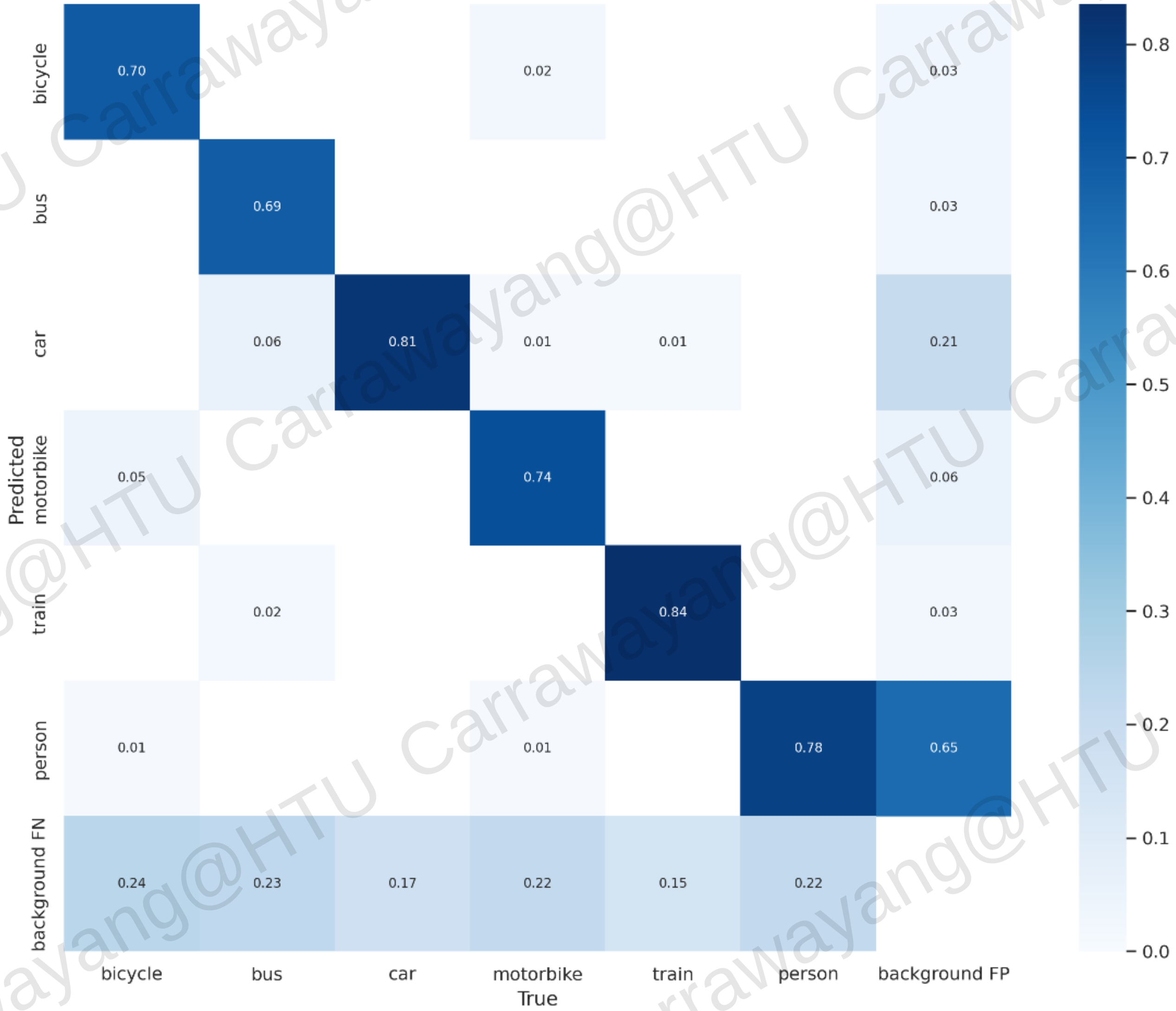


训练

Class	Images	Labels	P	R	mAP@.5	mAP@.5:.95
All	2510	3572	0.78	0.741	0.787	0.473
Bicycle	2510	177	0.783	0.754	0.793	0.459
Bus	2510	114	0.731	0.702	0.745	0.495
Car	2510	625	0.805	0.776	0.824	0.544
Motorbike	2510	172	0.751	0.703	0.75	0.418
Train	2510	152	0.828	0.763	0.811	0.489
Person	2510	2332	0.783	0.749	0.798	0.434

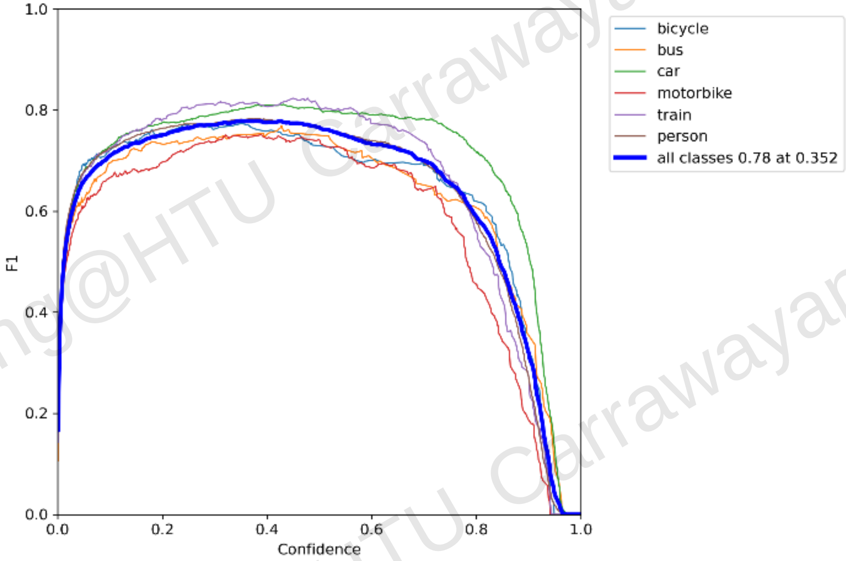


训练

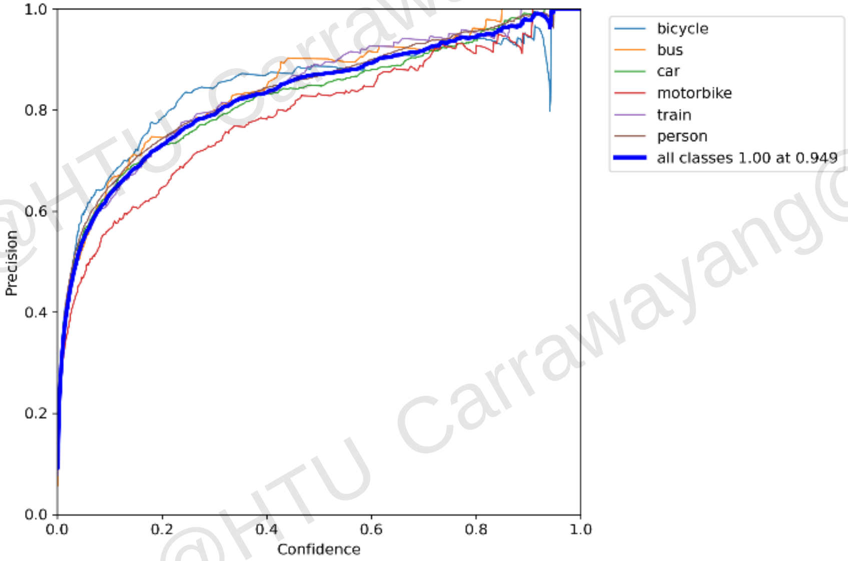


Model Summary: 213 layers, 1767283 parameters, 0 gradients, 4.2 GFLOPs

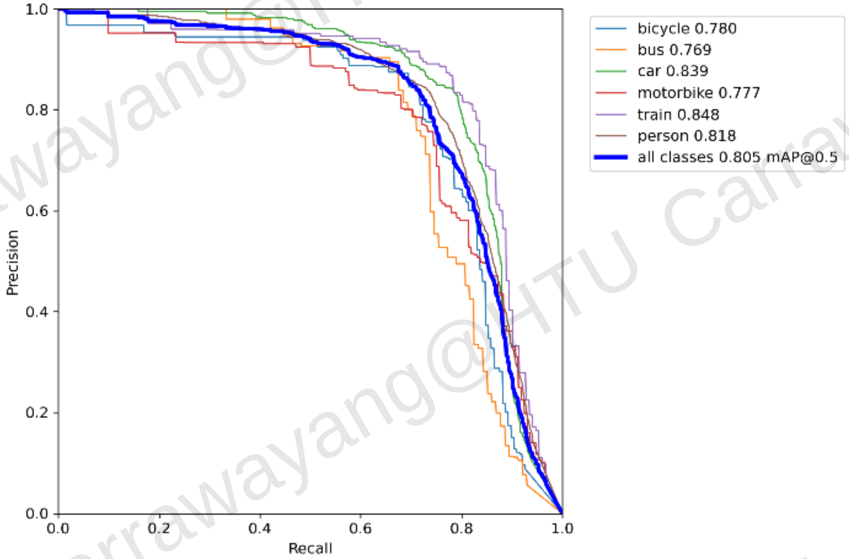
训练



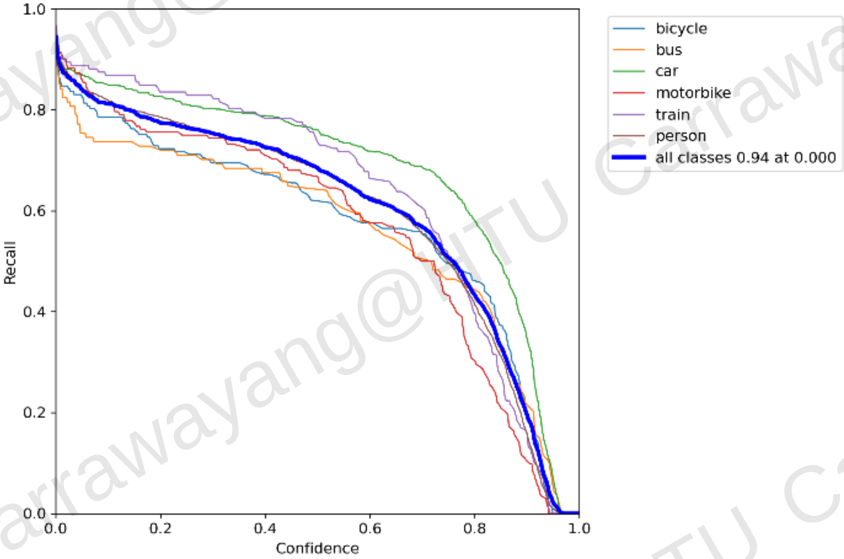
F1
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
bicycle
bus
car
motorbike
train
person
all classes 0.78 at 0.352



Precision
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
bicycle
bus
car
motorbike
train
person
all classes 1.00 at 0.949

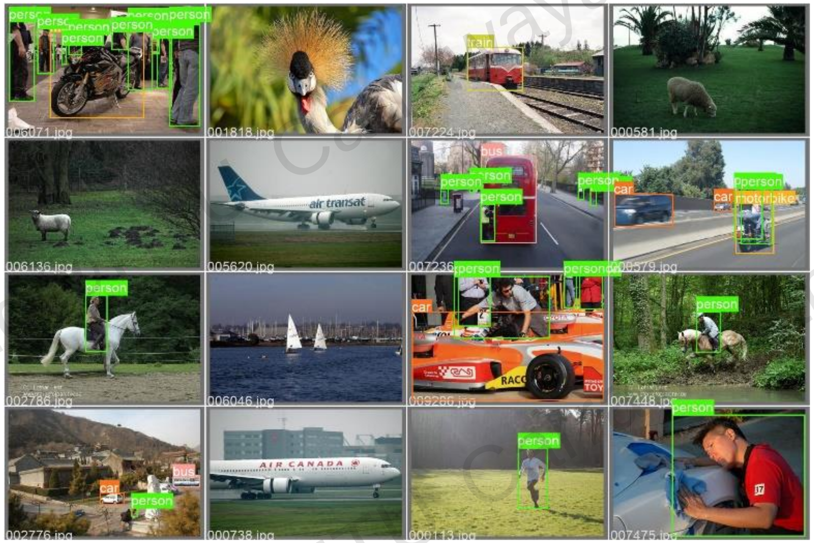


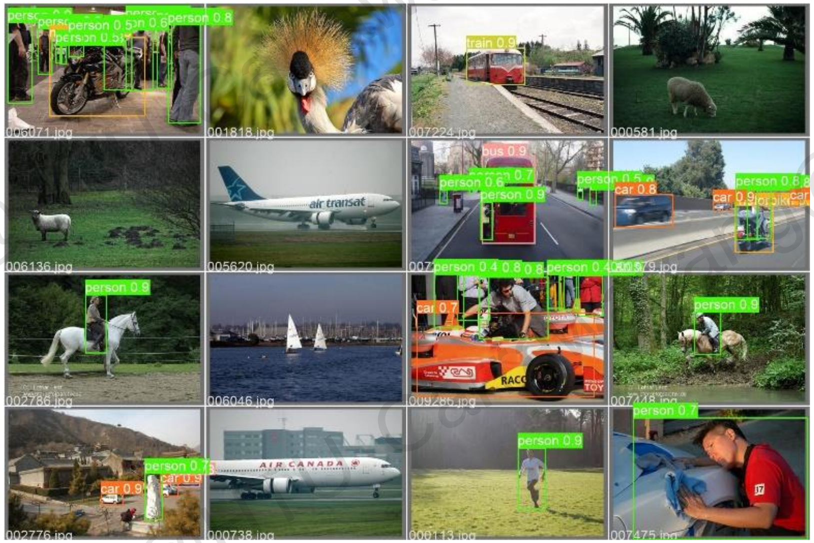
Precision
Recall
0.0
0.2
0.4
0.6
0.8
1.0
bicycle 0.780
bus 0.769
car 0.839
motorbike 0.777
train 0.848
person 0.818
all classes 0.805 mAP@0.5

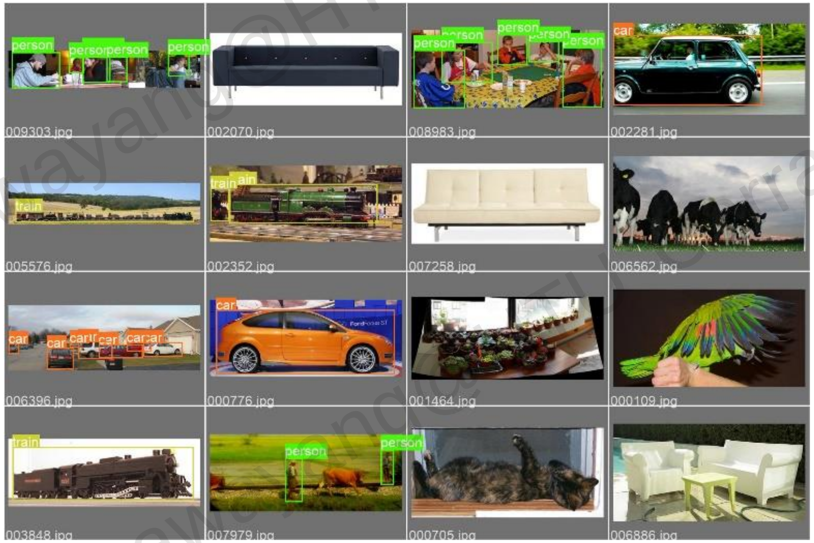


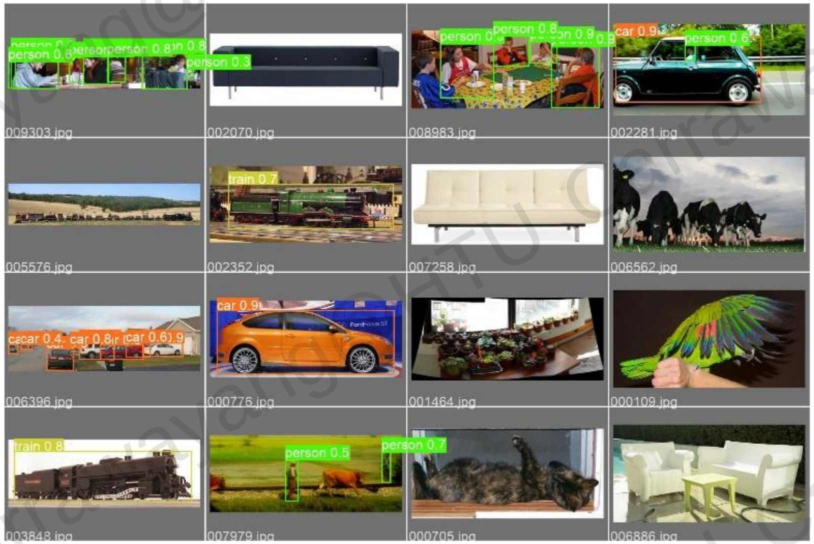
Recall
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
bicycle
bus
car
motorbike
train
person
all classes 0.94 at 0.000

训练

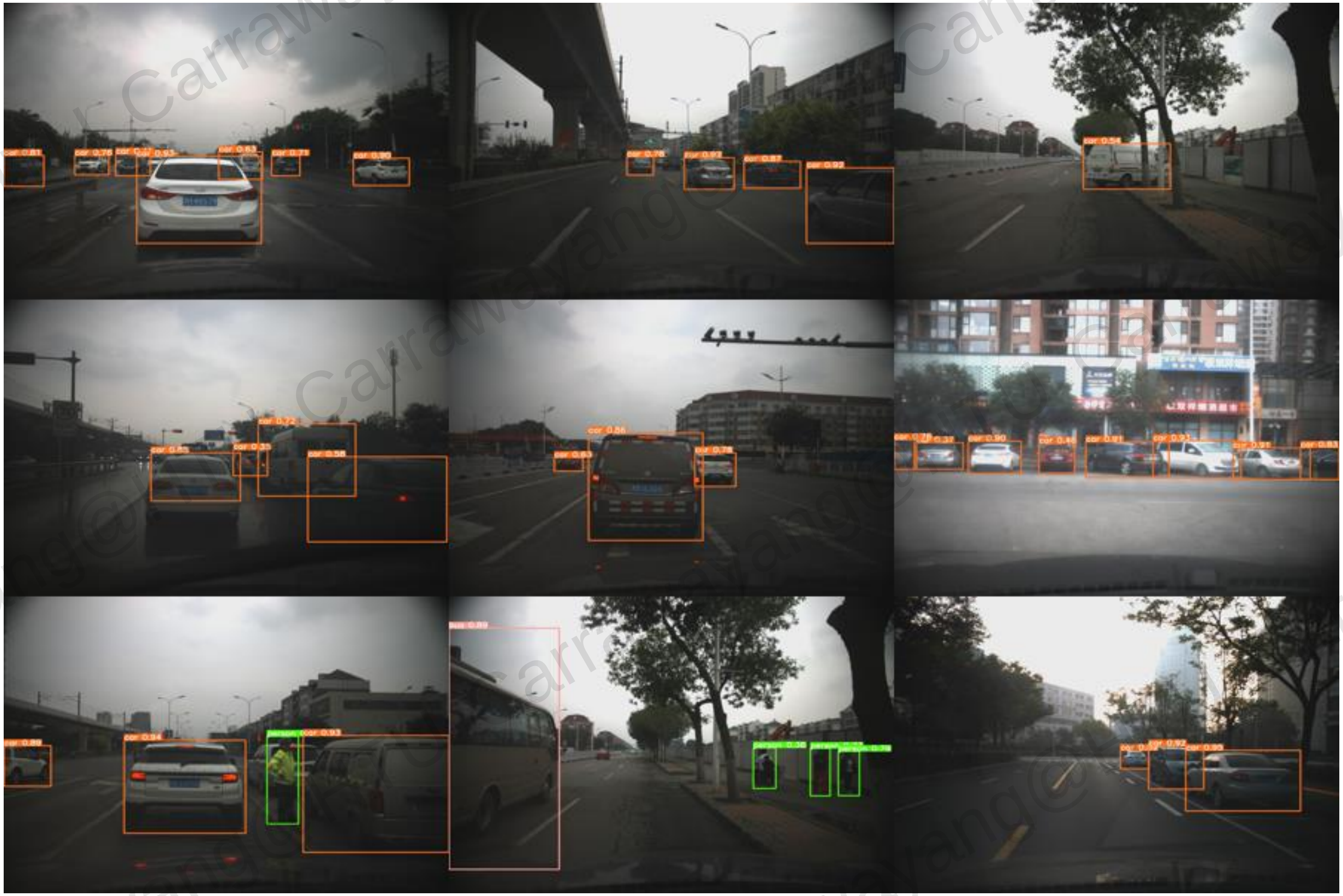

person
train
bus
car
motorbike
006071.jpg
001818.jpg
007224.jpg
000581.jpg
006136.jpg
005620.jpg
007236.jpg
008579.jpg
002786.jpg
006046.jpg
009286.jpg
007448.jpg
002776.jpg
000738.jpg
000113.jpg
007475.jpg


person 0.8
train 0.9
bus 0.9
person 0.6
car 0.8
person 0.9
car 0.7
person 0.7
car 0.9
006071.jpg
001818.jpg
007224.jpg
000581.jpg
006136.jpg
005620.jpg
007236.jpg
008579.jpg
002786.jpg
006046.jpg
009286.jpg
007448.jpg
002776.jpg
000738.jpg
000113.jpg
007475.jpg

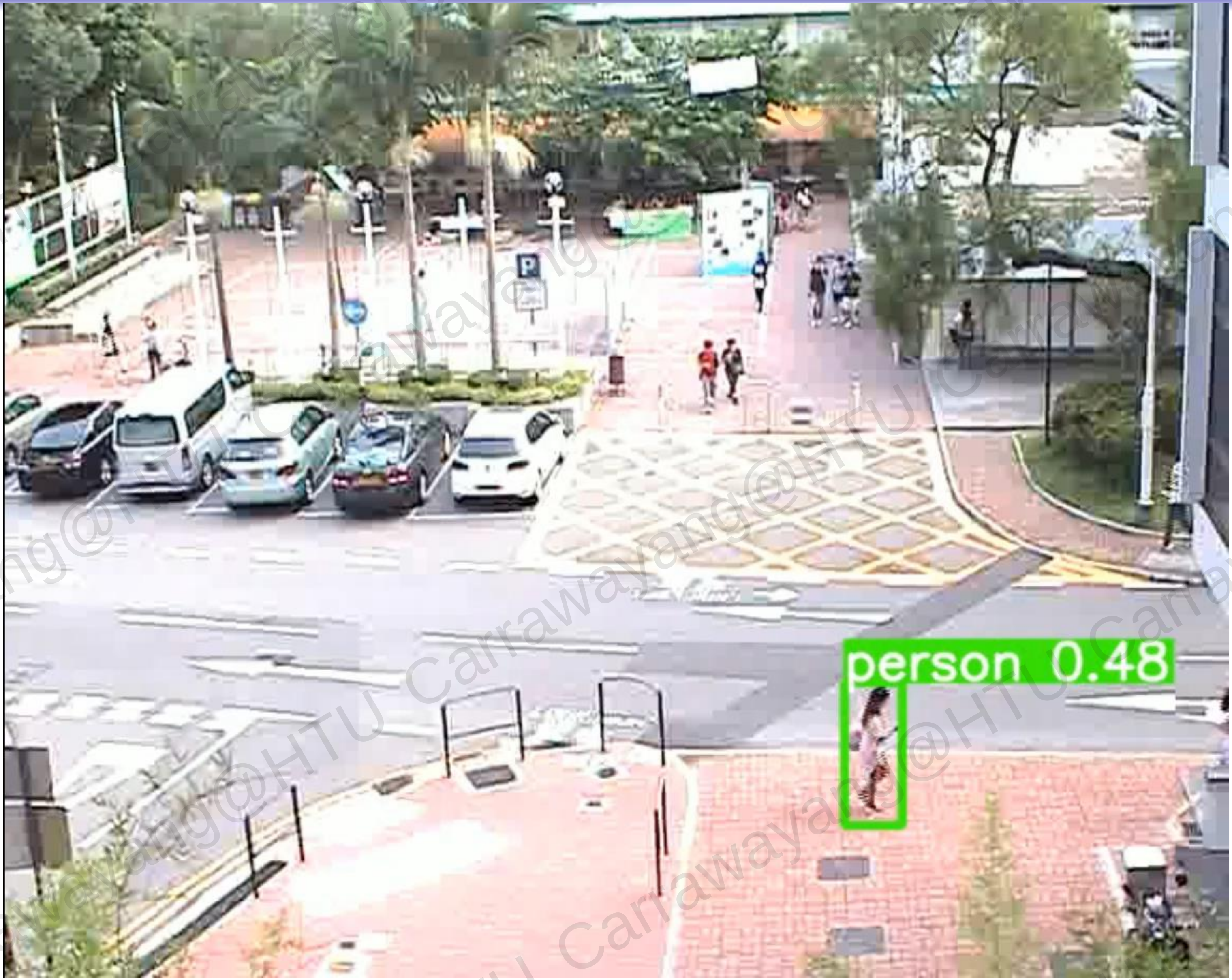

person
car
train
009303.jpg
002070.jpg
008983.jpg
002281.jpg
005576.jpg
002352.jpg
007258.jpg
006562.jpg
006396.jpg
000776.jpg
001464.jpg
000109.jpg
003848.jpg
007979.jpg
000705.jpg
006886.jpg


person 0.8
person 0.3
car 0.9
person 0.6
train 0.7
car 0.8
train 0.8
person 0.5
person 0.7
009303.jpg
002070.jpg
008983.jpg
002281.jpg
005576.jpg
002352.jpg
007258.jpg
006562.jpg
006396.jpg
000776.jpg
001464.jpg
000109.jpg
003848.jpg
007979.jpg
000705.jpg
006886.jpg

测试



测试

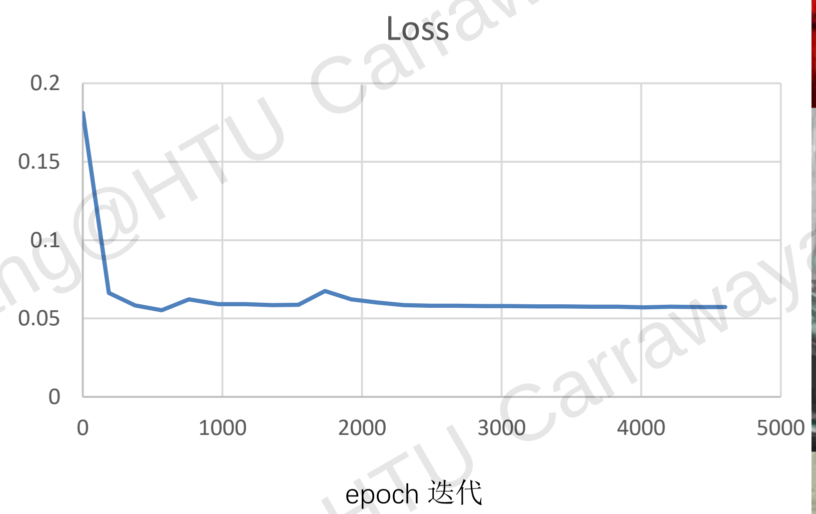


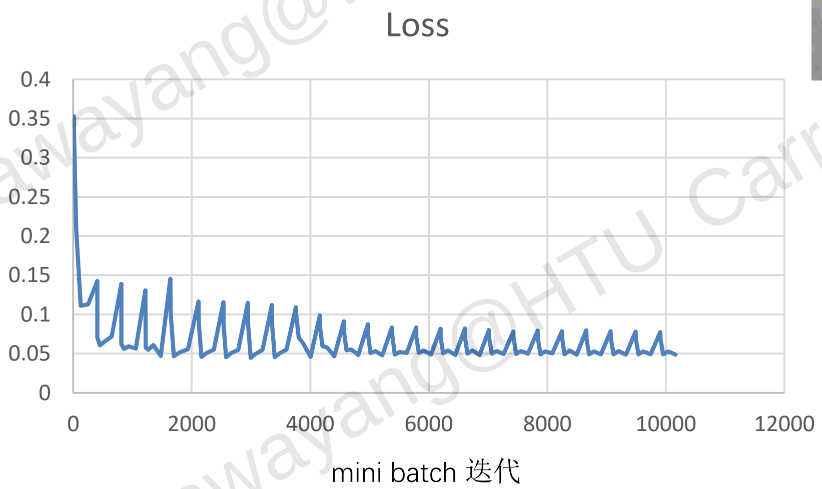
CCPD

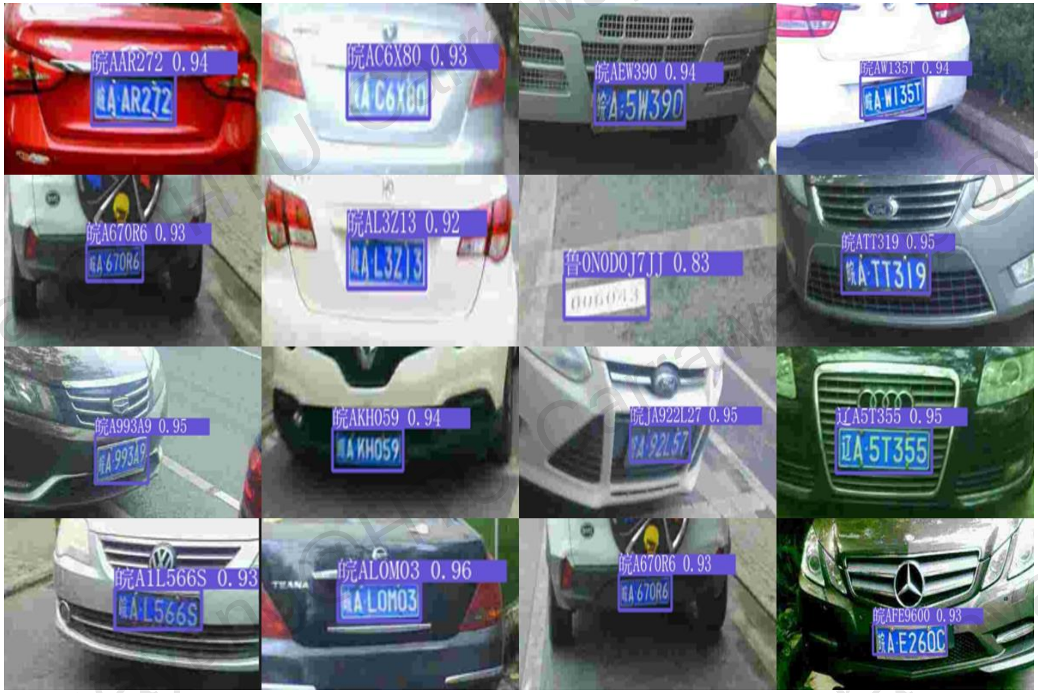
- CCPD(Chinese City Parking Dataset, ECCV)，是一个用于车牌识别的大型国内停车场车牌数据集，该数据在合肥市的停车场采集得来，采集时间为早上7：30到晚上10：00。停车场采集人员手持Android Pos机对停车场的车辆拍照并手工标注车牌位置。拍摄的车牌照片涉及多种复杂环境，包括模糊、倾斜、阴雨天、雪天等等。CCPD数据集一共包括进30万张图片，每张图片大小720*1160*3[6]。一共包含8项，具体如下：

CCPD类型	图片数量	说明
base	199998	正常车牌
challenge	10006	比较具有挑战性的车牌
db	20001	光线较暗或较亮
fn	19999	距离摄像头较远或较近
np	3036	没上牌的新车
rotate	9998	水平倾斜20-50°，垂直倾斜-10-10°
tilt	10000	水平倾斜15-45°，垂直倾斜15-45°
weather	9999	雨天、雪天或者雾天的车牌
总共：283037张车牌图像		

训练&测试



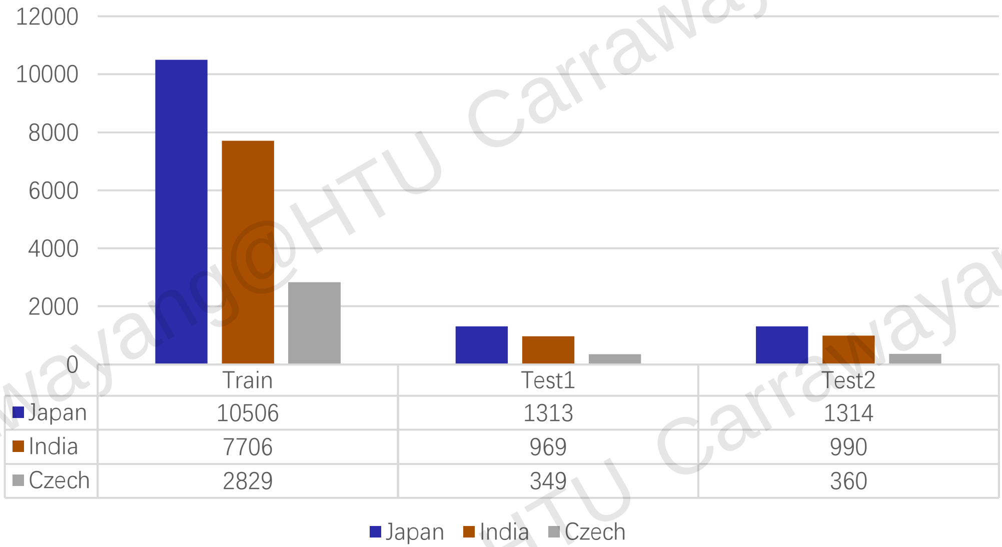




测试速度为每张图片85ms，车牌识别平均置信度为0.93，平均识别率为96%，在数据集中的少量样本中存在非车牌字符区域被识别

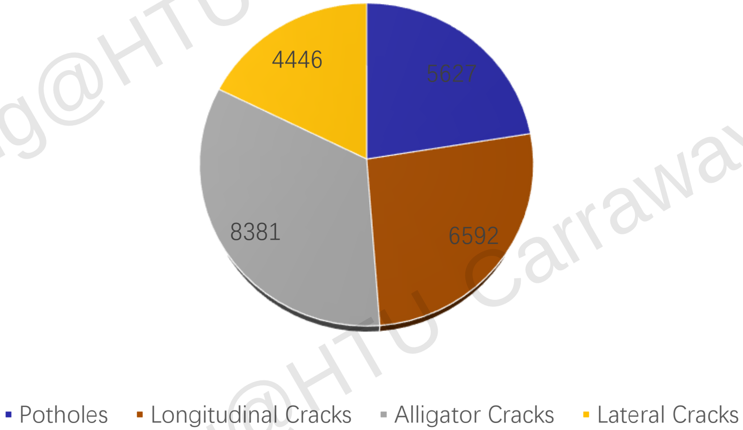
GRDDC2020

- GRDDC2020数据集(Golbal Road Damage Detection Challenge)， 是从印度、日本、捷克收集的道路图像。包括三个部分：Train， Test1， Test2， 训练数据集包括带有PASCAL VOC格式XML文件标注的道路图像，Train中没有标注，Test1和Test2中没有标注



	Train	Test1	Test2
■Japan	10506	1313	1314
■India	7706	969	990
■Czech	2829	349	360

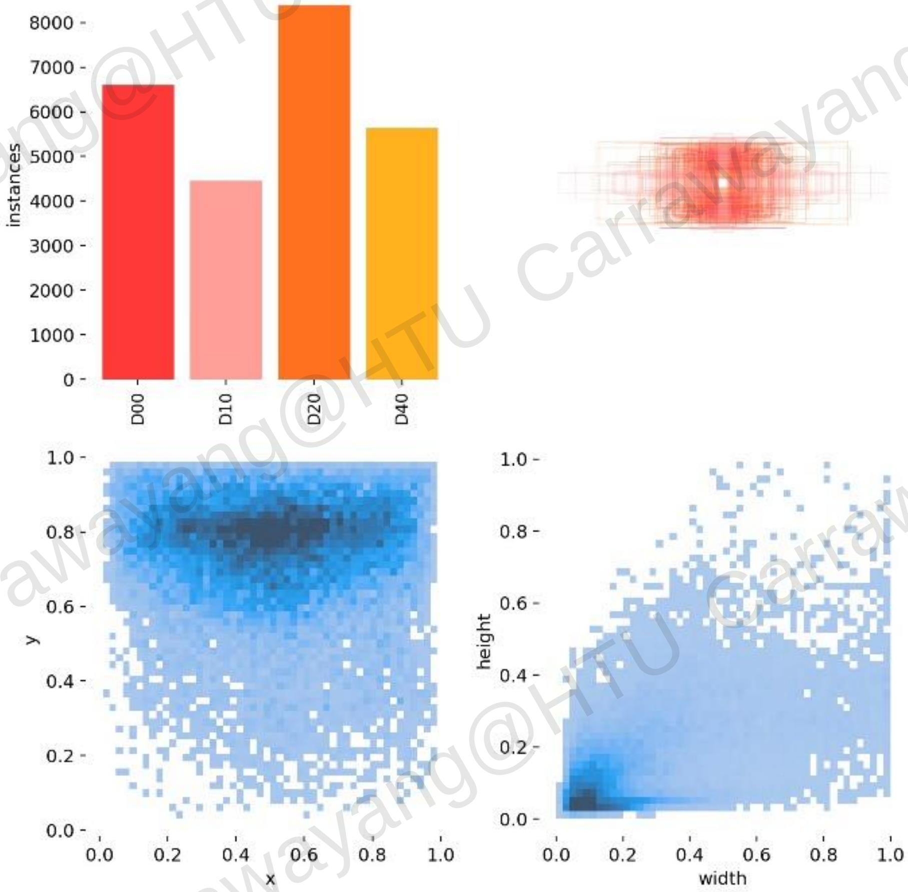
GRDDC Dataset



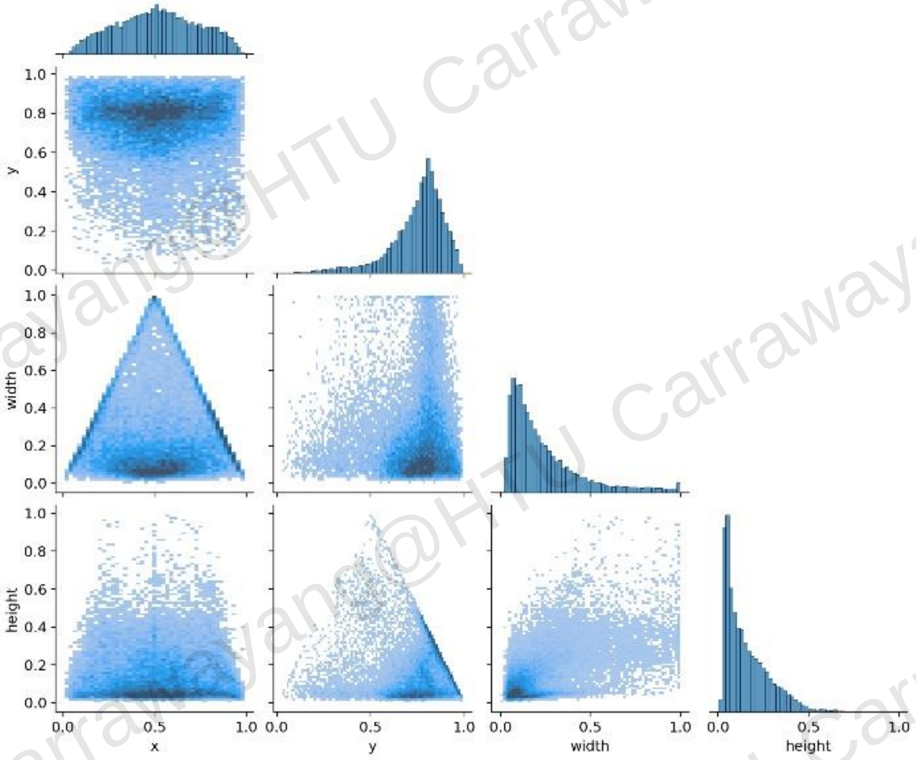
训练数据中每种损伤类型的实例数

训练

- 训练参数



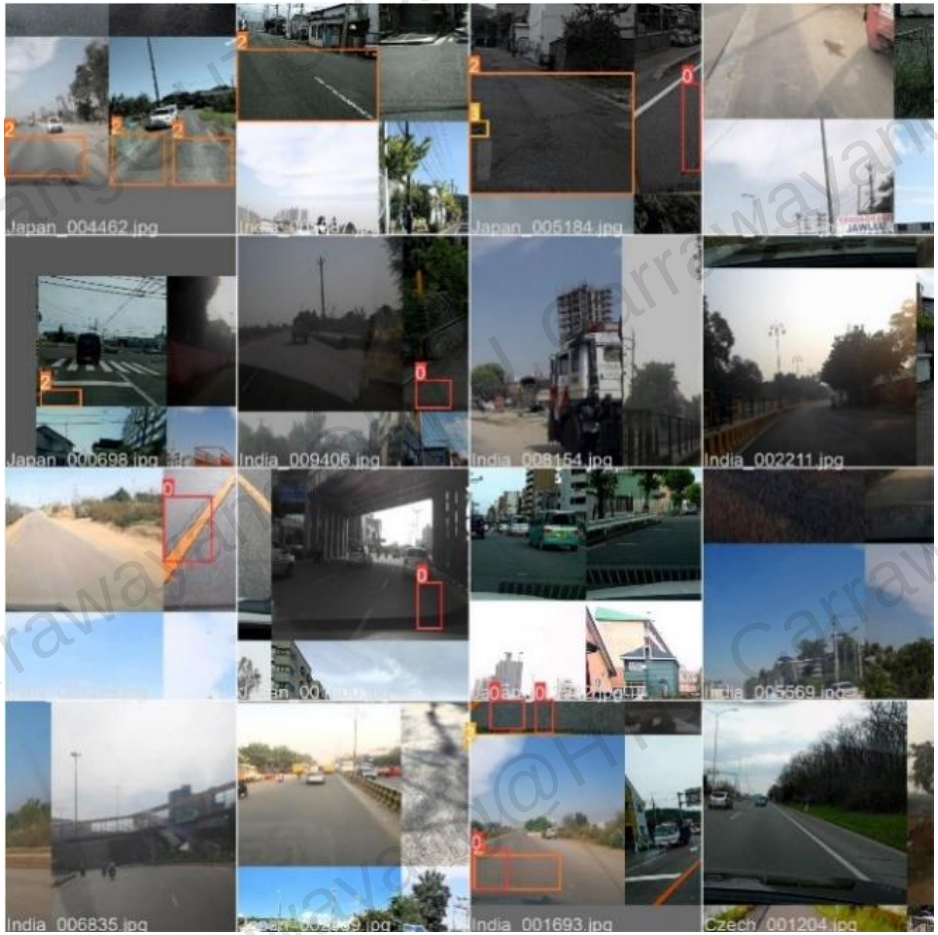
8000
7000
6000
5000
4000
3000
2000
1000
0
instances
D00
D10
D20
D40
1.0
0.8
0.6
0.4
0.2
0.0
y
0.0
0.2
0.4
0.6
0.8
1.0
x
1.0
0.8
0.6
0.4
0.2
0.0
height
0.0
0.2
0.4
0.6
0.8
1.0
width

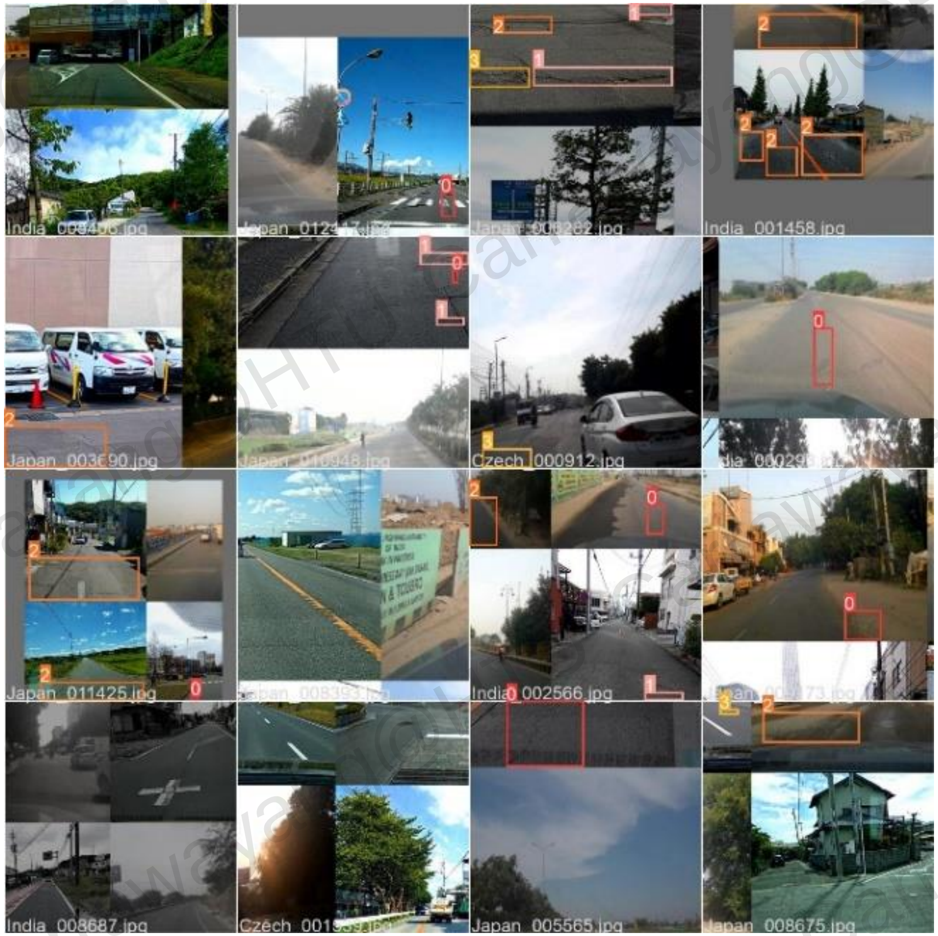


1.0
0.8
0.6
0.4
0.2
0.0
y
1.0
0.8
0.6
0.4
0.2
0.0
width
1.0
0.8
0.6
0.4
0.2
0.0
height
0.0
0.5
1.0
x
0.0
0.5
1.0
y
0.0
0.5
1.0
width
0.0
0.5
1.0
height

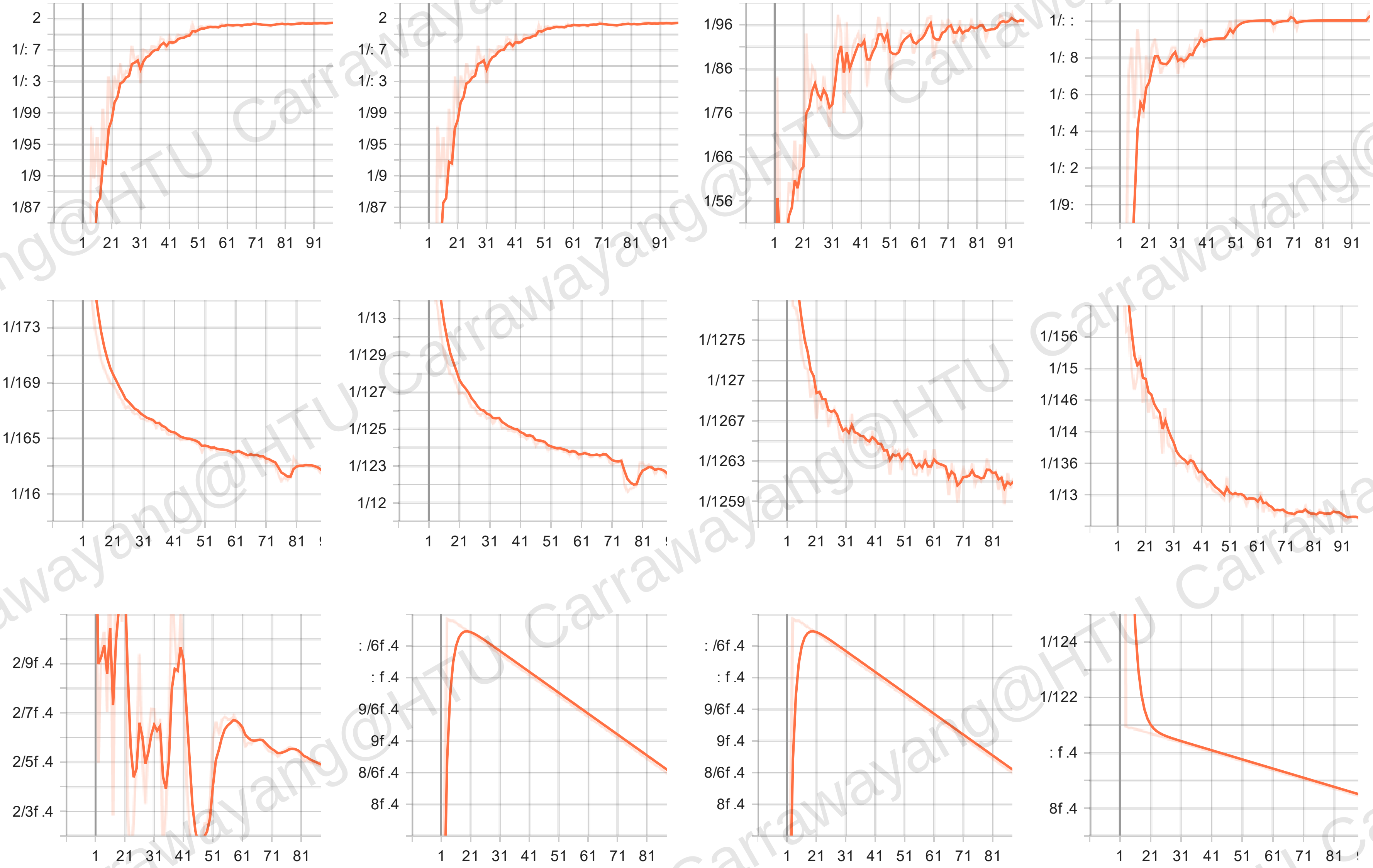
训练

cutout数据增强后一个Mini Batch

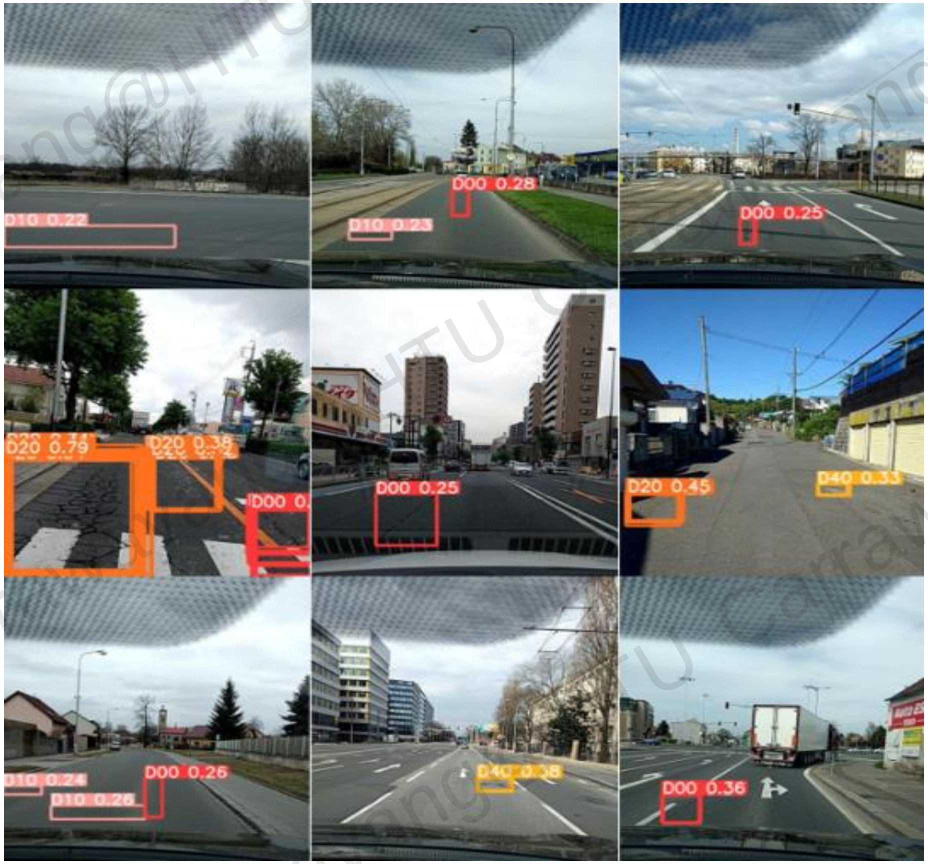


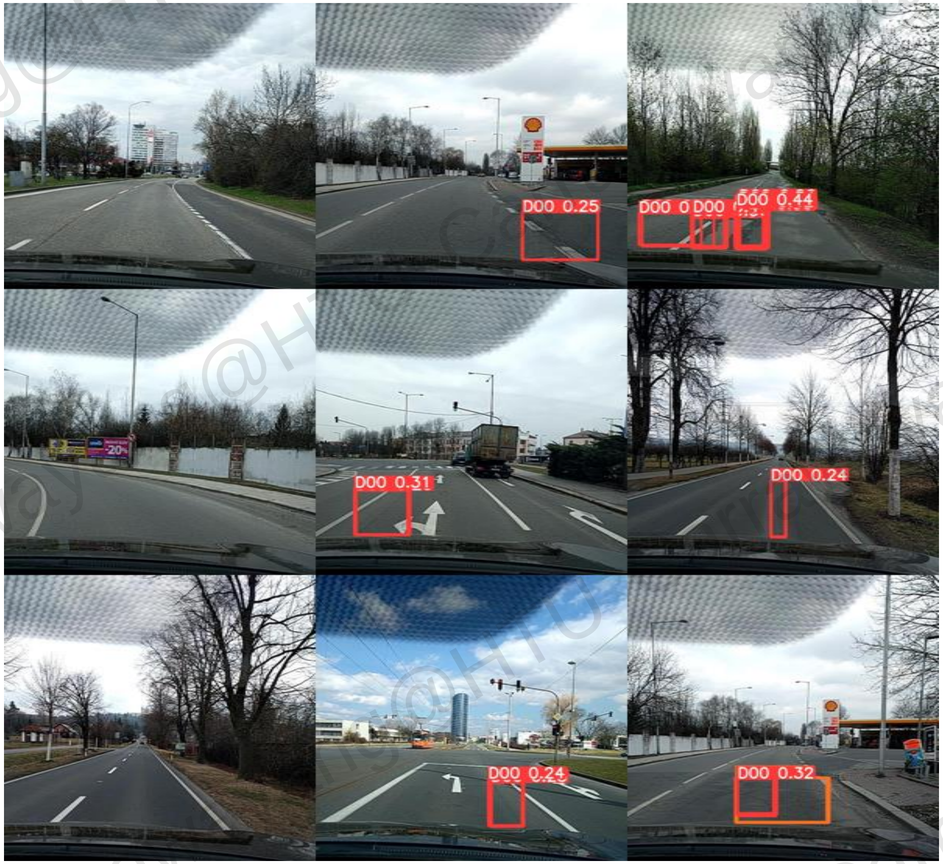


训练


2
1/: 7
1/: 3
1/99
1/95
1/9
1/87
1 21 31 41 51 61 71 81 91
2
1/: 7
1/: 3
1/99
1/95
1/9
1/87
1 21 31 41 51 61 71 81 91
1/96
1/86
1/76
1/66
1/56
1 21 31 41 51 61 71 81 91
1/: :
1/: 8
1/: 6
1/: 4
1/: 2
1/9:
1 21 31 41 51 61 71 81 91
1/173
1/169
1/165
1/16
1 21 31 41 51 61 71 81
1/13
1/129
1/127
1/125
1/123
1/12
1 21 31 41 51 61 71 81
1/1275
1/127
1/1267
1/1263
1/1259
1 21 31 41 51 61 71 81
1/156
1/15
1/146
1/14
1/136
1/13
1 21 31 41 51 61 71 81 91
2/9f .4
2/7f .4
2/5f .4
2/3f .4
1 21 31 41 51 61 71 81
: /6f .4
: f .4
9/6f .4
9f .4
8/6f .4
8f .4
1 21 31 41 51 61 71 81
: /6f .4
: f .4
9/6f .4
9f .4
8/6f .4
8f .4
1 21 31 41 51 61 71 81
1/124
1/122
: f .4
8f .4
1 21 31 41 51 61 71 81

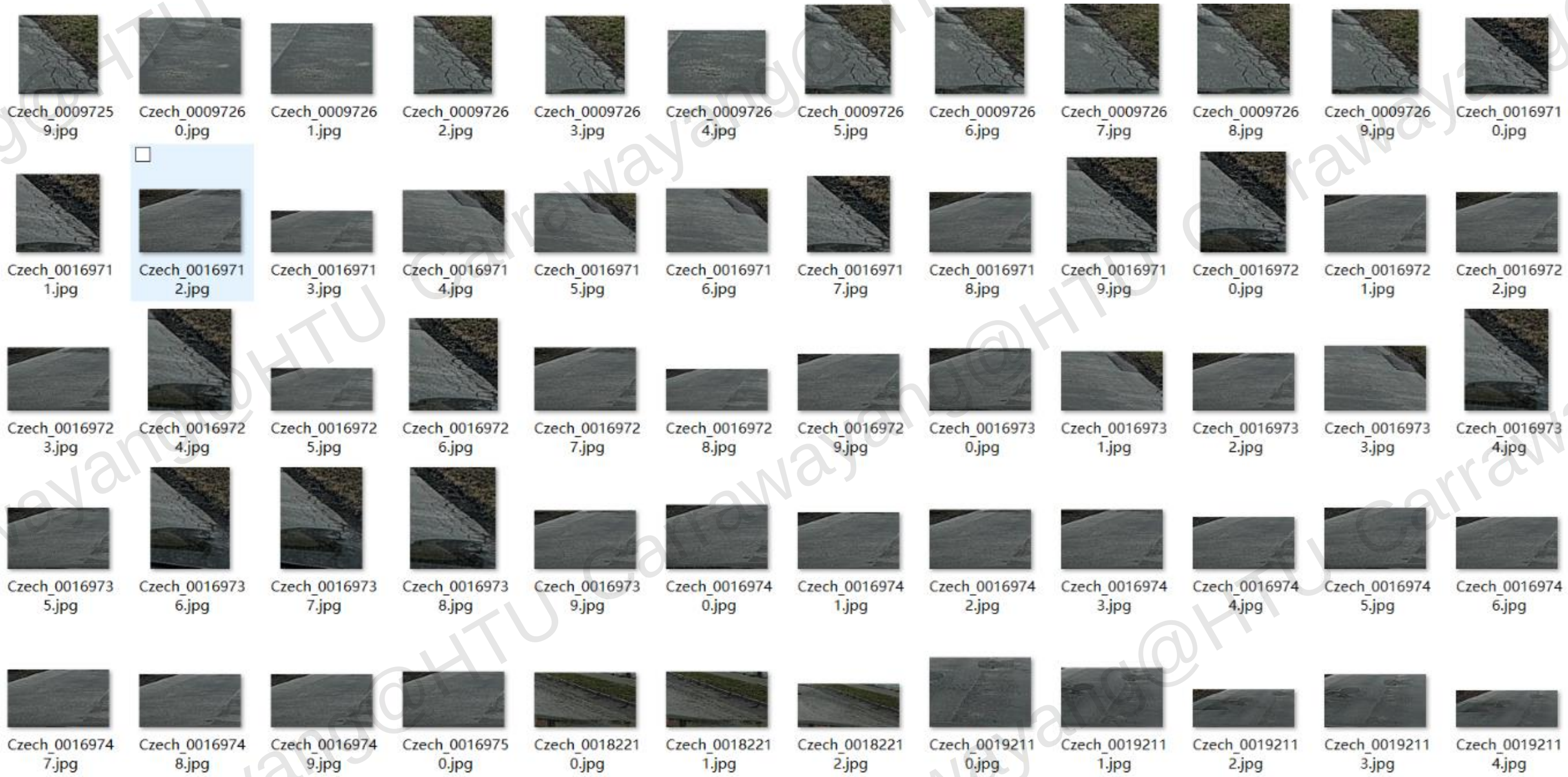
测试


D10 0.22
D00 0.28
D10 0.23
D00 0.25
D20 0.79
D20 0.38
D00 0.
D00 0.25
D20 0.45
D40 0.33
D10 0.24
D00 0.26
D10 0.26
D40 0.38
D00 0.36


D00 0.25
D00 0
D00
D00 0.44
D00 0.31
D00 0.24
D00 0.24
D00 0.32

测试

- 损伤截面



Czech_00097259.jpg
Czech_00097260.jpg
Czech_00097261.jpg
Czech_00097262.jpg
Czech_00097263.jpg
Czech_00097264.jpg
Czech_00097265.jpg
Czech_00097266.jpg
Czech_00097267.jpg
Czech_00097268.jpg
Czech_00097269.jpg
Czech_00169710.jpg

Czech_00169711.jpg
Czech_00169712.jpg
Czech_00169713.jpg
Czech_00169714.jpg
Czech_00169715.jpg
Czech_00169716.jpg
Czech_00169717.jpg
Czech_00169718.jpg
Czech_00169719.jpg
Czech_00169720.jpg
Czech_00169721.jpg
Czech_00169722.jpg

Czech_00169723.jpg
Czech_00169724.jpg
Czech_00169725.jpg
Czech_00169726.jpg
Czech_00169727.jpg
Czech_00169728.jpg
Czech_00169729.jpg
Czech_00169730.jpg
Czech_00169731.jpg
Czech_00169732.jpg
Czech_00169733.jpg
Czech_00169734.jpg

Czech_00169735.jpg
Czech_00169736.jpg
Czech_00169737.jpg
Czech_00169738.jpg
Czech_00169739.jpg
Czech_00169740.jpg
Czech_00169741.jpg
Czech_00169742.jpg
Czech_00169743.jpg
Czech_00169744.jpg
Czech_00169745.jpg
Czech_00169746.jpg

Czech_00169747.jpg
Czech_00169748.jpg
Czech_00169749.jpg
Czech_00169750.jpg
Czech_00182210.jpg
Czech_00182211.jpg
Czech_00182212.jpg
Czech_00192110.jpg
Czech_00192111.jpg
Czech_00192112.jpg
Czech_00192113.jpg
Czech_00192114.jpg

目录

- YOLO
- 行人车辆检测
- 车牌检测
- 路面损伤检测
- 实验
- **测试**

Welcome

欢迎批评指正！