18CSC402
DEEP LEARNING
CASE STUDY : FINAL REVIEW

A S Naveena

CB.SC.I5DAS18024

Int. MSc. Data Science

A GLIMPSE OF THE PAPER

- **Paper title:** Real-time license plate detection for non-helmeted motorcyclist using YOLO
- **Authors:** Yonten Jamtsho, Panomkhawn Riyamongkol, Rattapoom Waranusast
- **Published in:** Information & Communications Technology (ICT) Express, (Volume 7, Issue 1) by The Korean Institute of Communications and Information Sciences, on 22 August 2020.
- **Dataset used:** Collected from Naresuan University, Thailand. (Not publicly available) Around 1365 images were annotated with the bounding box information including the class labels for three classes (Person, Helmet, Plate) using LabelImg software.

ARCHITECTURE USED : YOLOV2

19 convolutional layers and 5 max-pooling layers, followed by softmax activation functions to classify the object.

Type	Filters	Size/Stride	Output
Convolutional	32	3×3	224×224
Maxpool		$2 \times 2/2$	112×112
Convolutional	64	3×3	112×112
Maxpool		$2 \times 2/2$	56×56
Convolutional	128	3×3	56×56
Convolutional	64	1×1	56×56
Convolutional	128	3×3	56×56
Maxpool		$2 \times 2/2$	28×28
Convolutional	256	3×3	28×28
Convolutional	128	1×1	28×28
Convolutional	256	3×3	28×28
Maxpool		$2 \times 2/2$	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Maxpool		$2 \times 2/2$	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	1000	1×1	7×7
Avgpool		Global	1000
Softmax			

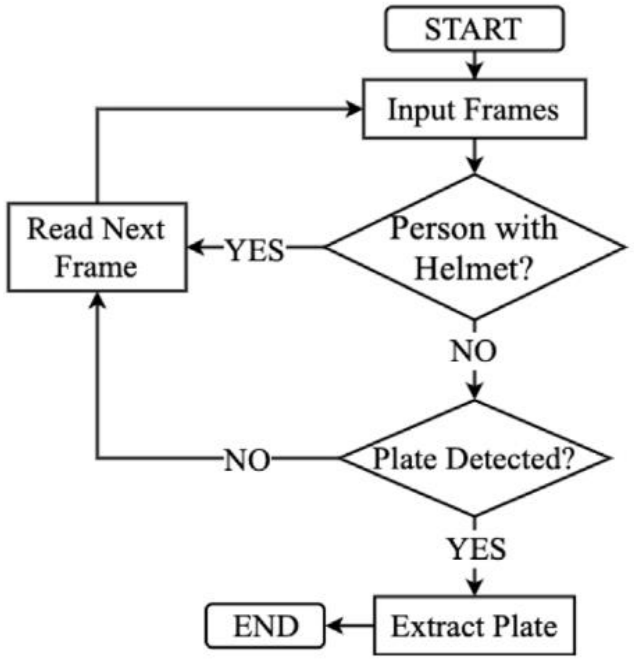


Fig. 3. System overview.

EVALUATION METRICS

Table 1
Tabulation of AP with average IOU of each class.

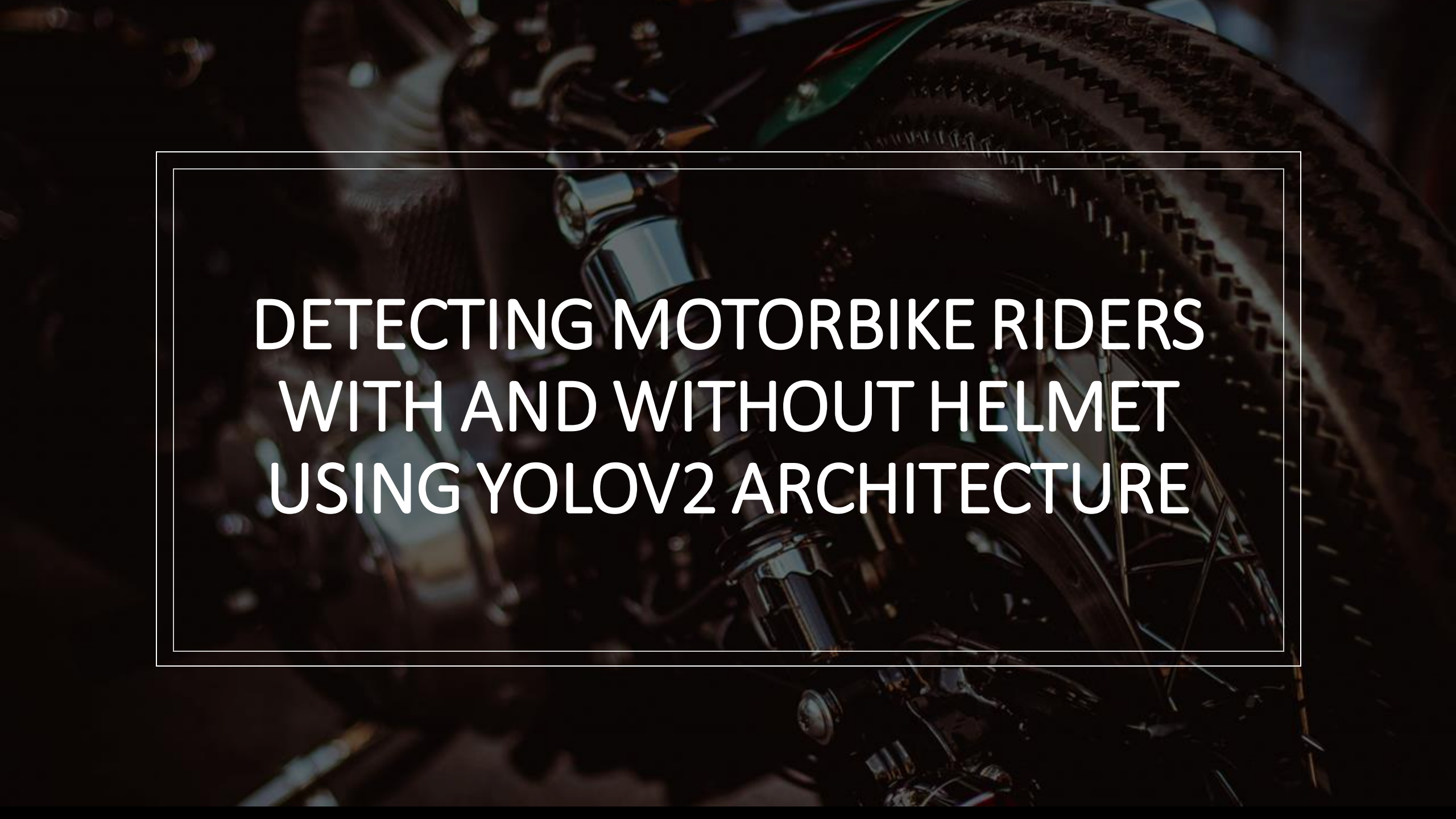
Epochs	Person	Helmet	Plate	Avg IOU
1000	97.78	96.08	98.17	71.58
2000	96.79	97.36	98.53	73.12
3000	96.56	98.19	98.16	74.12
4000	97.15	98.17	98.39	74.59
5000	96.97	98.18	98.24	74.5
6000	97.05	97.86	98.21	74.53
7000	**96.97**	**97.85**	**98.4**	**74.72**
8000	97	97.84	98.41	74.62
9000	97.02	97.85	98.39	74.52
10,000	97.3	97.85	98.42	74.5

Table 2
The Confusion matrix obtained from the experiments.

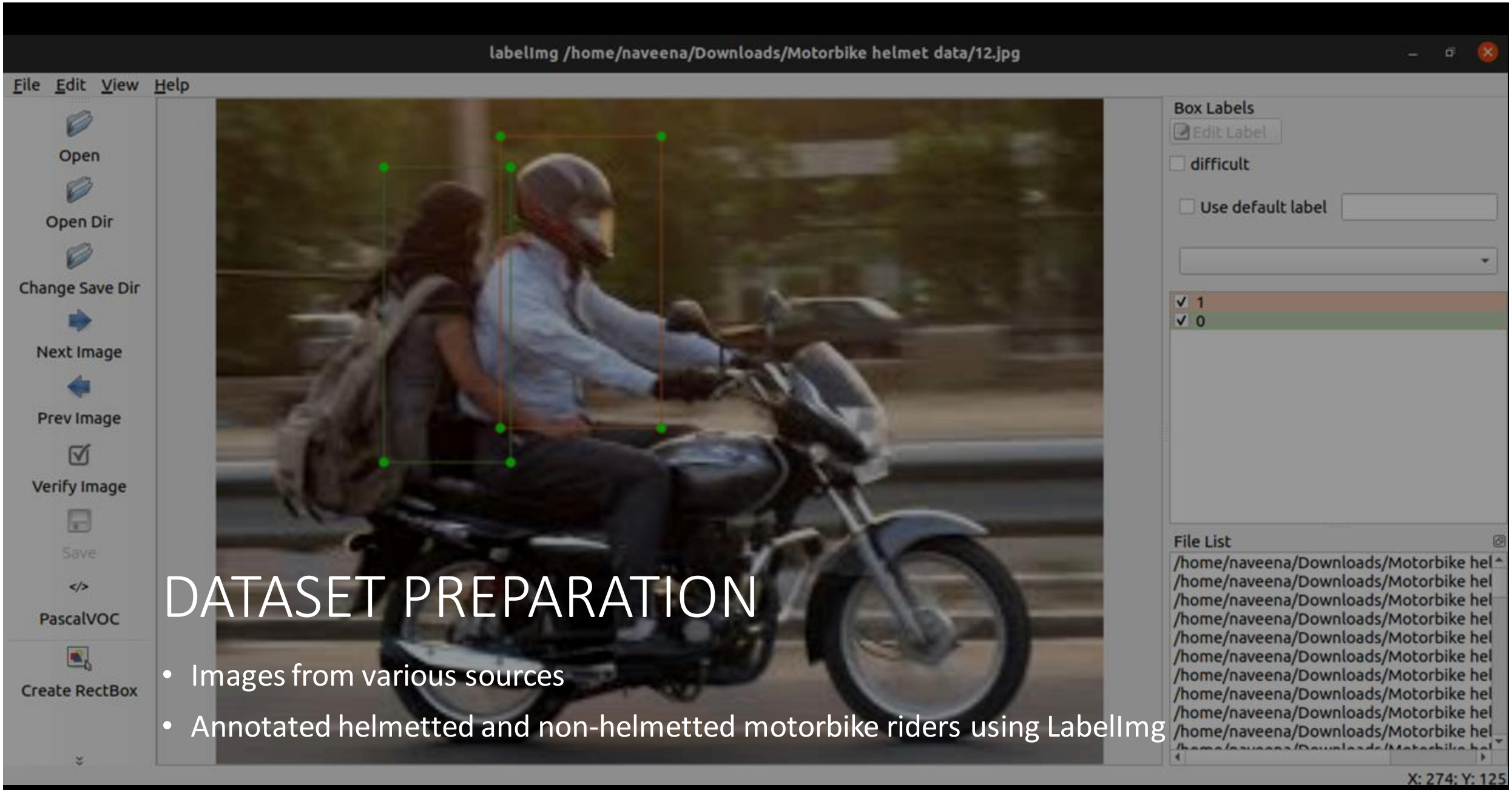
	With helmet	Without helmet
With helmet	78	7
Without helmet	2	**135**

Table 3
Precision, recall and F1-score generated from the confusion matrix.

	Precision (%)	Recall (%)	F1-Score (%)
With helmet	97.5	91.76	94.54
Without helmet	95.07	98.54	96.77
Weighted avg	96	95.94	95.92



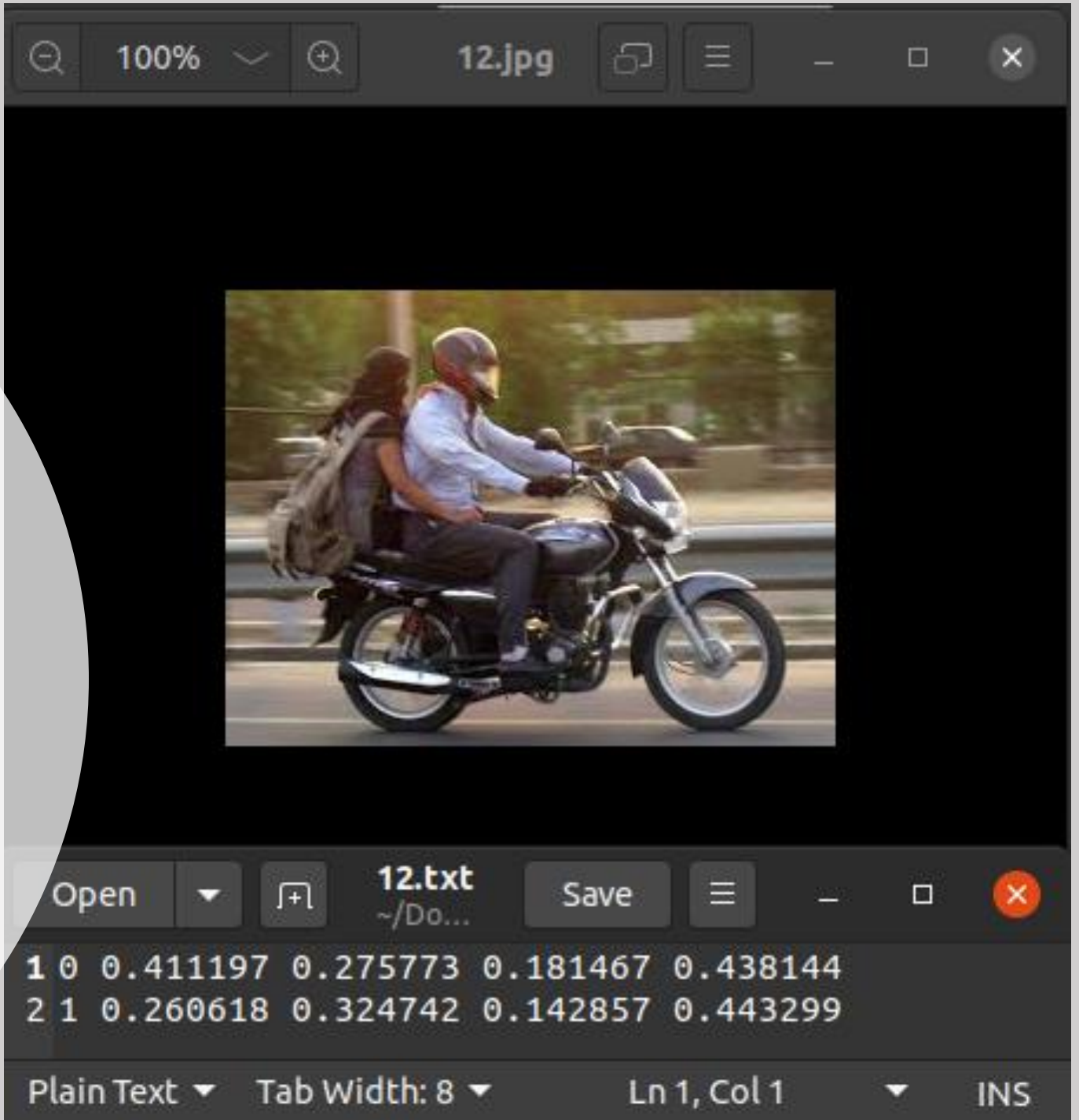
DETECTING MOTORBIKE RIDERS WITH AND WITHOUT HELMET USING YOLOV2 ARCHITECTURE



DATASET PREPARATION

- Images from various sources
- Annotated helmetted and non-helmetted motorbike riders using LabelImg

INPUT IMAGE AND ANNOTATED FILE

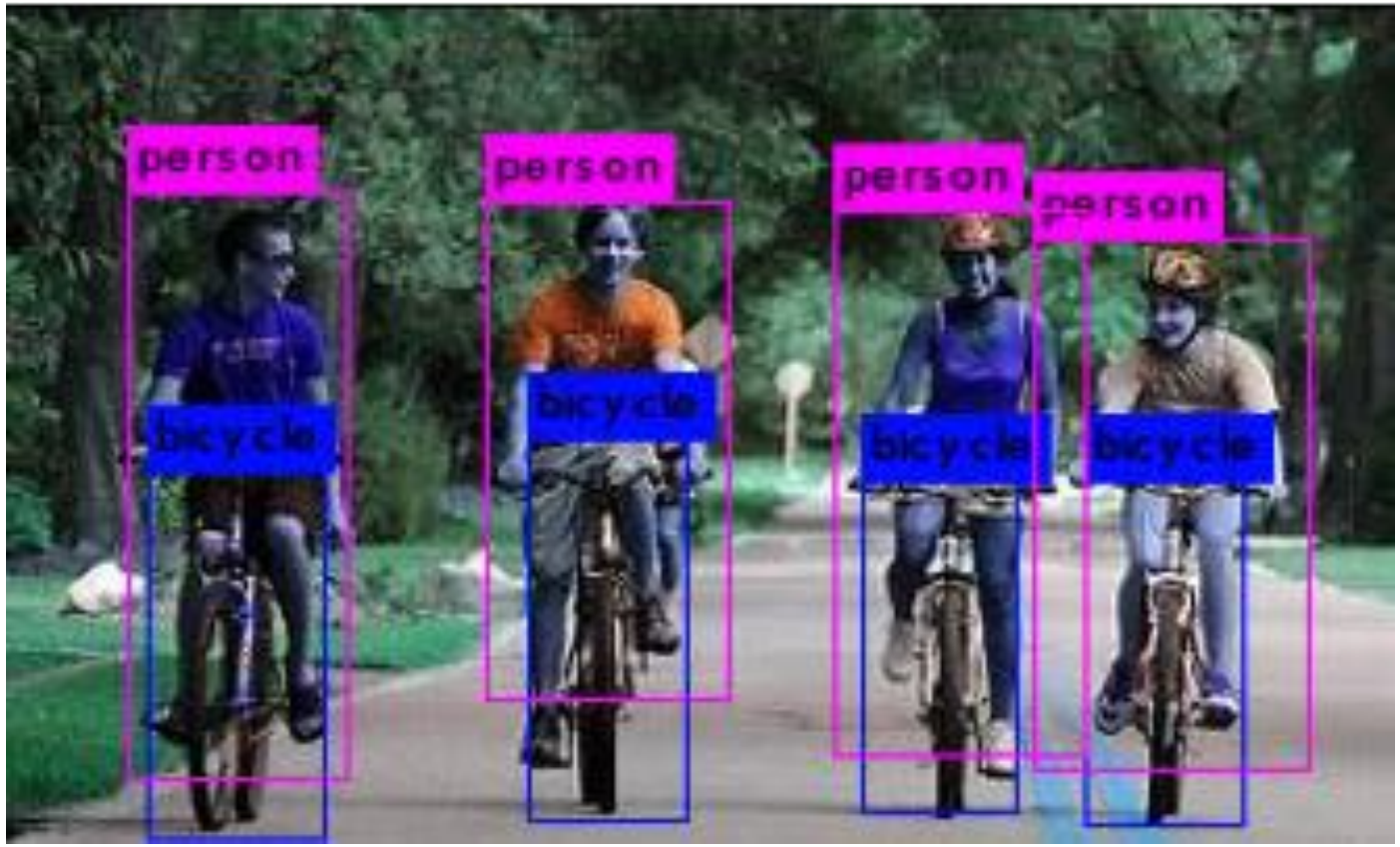


GETTING STARTED

- YOLO built on top of darknet
- Official website : https://pjreddie.com/darknet/yolov2/
- Experimented with:
  - YOLOv2 544x544
  - Tiny YOLO

  (See slides 18 and 19)

PREDICTION PRE-TRAINING



```
<annotation>
  <folder>Motorbike helmet data</folder>
  <filename>12.jpg</filename>
  <path>/home/naveena/Downloads/Motorbike helmet data/12.jpg</path>
  <source>
    <database>Unknown</database>
  </source>
  <size>
    <width>259</width>
    <height>194</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>1</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>83</xmin>
      <ymin>11</ymin>
      <xmax>130</xmax>
      <ymax>96</ymax>
    </bndbox>
  </object>
  <object>
    <name>0</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>49</xmin>
      <ymin>20</ymin>
      <xmax>86</xmax>
      <ymax>106</ymax>
    </bndbox>
  </object>
</annotation>
```

TRAINING DARKFLOW ON CUSTOM DATA

- Darkflow repo : https://github.com/thtrieu/darkflow
- Requirements : Python 3.7 or lesser, Tensorflow 1.15
- Weight file : yolo.weights
- Input : 50 images (.jpg) and their annotations (.xml) stored in separate directories
- Class labels : stored in labels.txt
  - 1 => rider wearing helmet
  - 0 => rider not wearing helmet
- After 100 epochs : loss 4.507138252258301, moving avg loss 4.397005312339386

CUSTOM CONFIGURATION OF CFG FILES

- yolo.cfg

```
1 [net]
2 # Testing
3 batch=1
4 subdivisions=1
5 # Training
6 # batch=64
7 # subdivisions=8
```

```
233 [convolutional]
234 size=1
235 stride=1
236 pad=1
237 filters=425
238 activation=linear
239
240
241 [region]
242 anchors =  0.57273, 0.677385, 1.87446,
    2.06253, 3.33843, 5.47434, 7.88282, 3.52778,
    9.77052, 9.16828
243 bias_match=1
244 classes=80
245 coords=4
246 num=5
```

- yolo_custom.cfg

```
1 [net]
2 # Testing
3 #batch=1
4 #subdivisions=1
5 # Training
6 batch=12
7 subdivisions=4
```

```
233 [convolutional]
234 size=1
235 stride=1
236 pad=1
237 filters=35
238 activation=linear
239
240
241 [region]
242 anchors =  0.57273, 0.677385, 1.87446,
    2.06253, 3.33843, 5.47434, 7.88282, 3.52778,
    9.77052, 9.16828
243 bias_match=1
244 classes=2
245 coords=4
246 num=5
```

TRAINING

Statistics:

1: 102

0: 53

Dataset size: 50

Dataset of 50 instance(s)

Training statistics:

Learning rate : 1e-05

Batch size : 8

Epoch number : 100

Backup every : 2000

step 1 -

loss 55.92295455932617 -moving ave loss 55.92295455932617.

.

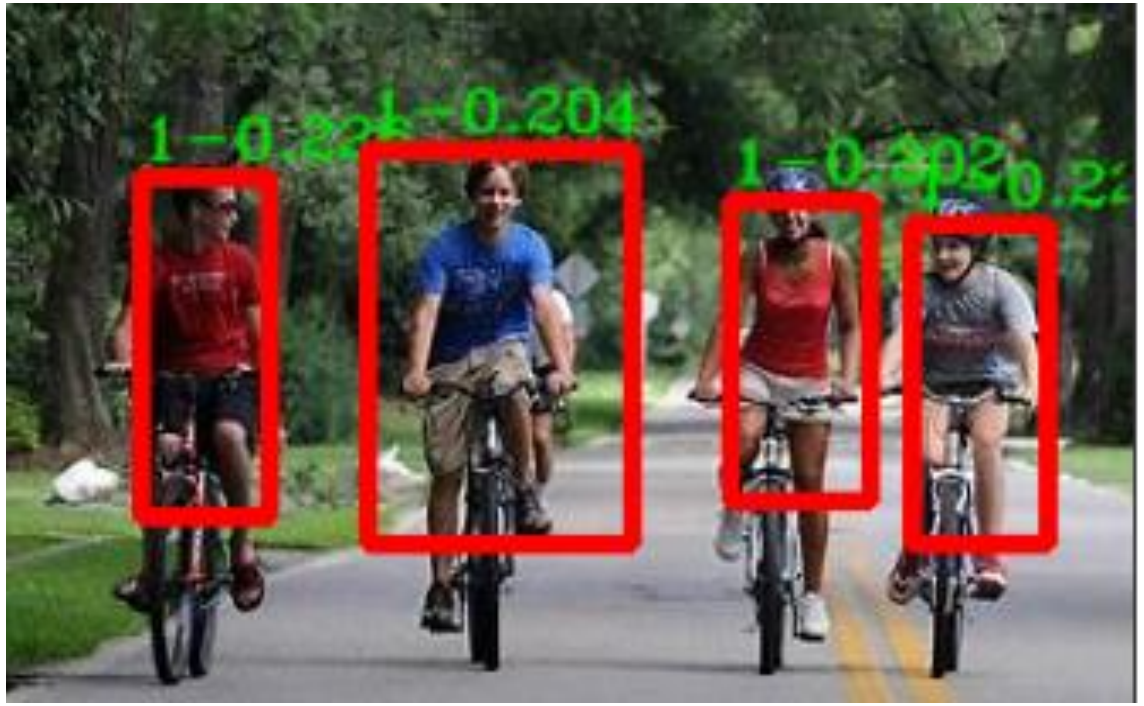
.

step 600 -

loss 4.507138252258301 - moving ave loss 4.397005312339386

Finish 100 epoch(es)

TESTING WITH UNSEEN IMAGES



TESTING WITH VIDEOS

The model was tested on two video files. See here!

WHAT COULD BE WRONG?

- Small dataset
- **Imbalanced** count of images in each label
- Differences in annotation
- Training for too many epochs

WHAT CAN BE DONE?

- Increase dataset size
- Balance number of images in each label
- Implement centroid tracking method
- Extend to license plate detection

PREVIOUS ATTEMPTS

ATTEMPT 1:

training YOLOv2 with custom data

- Split images and annotation files into training and testing sets.
- Configured cfg file to match requirements.
- Weight file : darknet19_448.conv.23
- After 6 hours of training, iteration 1194: total loss 376.851868, avg loss 327.294830
- Tested with most recent weights (after 900 iterations)

… FAILED!

ATTEMPT 2 :

training tiny-YOLO with custom data

- Split images and annotation files into training and testing sets.
- Configured cfg file to match requirements.
- Weight file : darknet19_448.conv.23
- After 4 hours oftraining, iteration 824: total loss 235.150879, avg loss 280.017517
- Tested with most recent weights (after 800 iterations)

… FAILED!

THANK YOU!