



## Title: README Tiny Yolo v4 GPU

Document Number: 105-1b

CTI One Corporation

**Table 1a. Document History**

2022-09-28	Establish this document, document archive: harry@workstation:/media/harry/easystore1/backup- 2020-2-15/CTI/3proejcts/3-8-smart-tech/3-8-4-CTI/3-8-4- 6-products/Sabs100/105-yolov4-nano-ebike/105-3-readme- yolo4\$	YY
2022-09-30	Approved for release for SJSU education use.	HL

**Table 1b. Testing and Release Approval Form**

2022-09-28	Tested by YY and approved for release by HL	HL

**Table 2. References**

Number	Name and URL	Note
1.	YOLOv4  <a href="https://github.com/theAIGuysCode/yolov4-custom-functions">https://github.com/theAIGuysCode/yolov4-custom-functions</a>	
2.		

**Table 3. Prerequisite**



Software Prerequisite No.	Description and Version	Note
1.	Ubuntu 18.04	
2.	Anaconda version 4.7.12 or later	
3.	git version 2.17.1 or later	
Hardware Prerequisite No.	Description and Version	
1.	Intel CPU	Tested with Core i7
2.	NVIDIA GPU	Tested with RTX 2070



## 1. Setup Yolo v4 environment

1.1. Clone the GitHub folder;

```
$ git clone https://github.com/theAIGuysCode/yolov4-custom-functions.git
```

1.2. Create YAML file for building Anaconda environment;

Modify yolov4-custom-functions/conda-gpu.yml;

```
=====
```

```
name: yolov4-gpu
```

```
dependencies:
```

- python==3.7.13
- pip
- matplotlib
- opencv
- cudnn
- cudatoolkit==10.1.243
- pip:
  - tensorflow-gpu==2.3.1
  - opencv-python==4.1.1.26
  - lxml
  - tqdm
  - absl-py
  - easydict
  - pillow
  - pytesseract

```
=====
```

NOTE: Using TensorFlow 2.4.0 or later for this YOLO v4 implementation performs 4 times slower than TensorFlow 2.3.1 with Nvidia RTX 2070.

1.3. Create a Anaconda environment;



Open a terminal in yolov4-custom-functions folder and perform the below command;

```
conda env create -f conda-gpu.yml
```

1.4. Activate the Anaconda environment;

```
conda activate yolo4-gpu
```

1.5. Download Tiny YOLO v4 model files;

```
wget -P data https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v4_pre/yolov4-tiny.weights
```

1.6. Convert the Darknet model file to TensorFlow saved model format;

```
$ python save_model.py --weights ./data/yolov4-tiny.weights --output ./checkpoints/yolov4-tiny-416 --input_size 416 --model yolov4 --tiny
```

The TensorFlow saved model format files will be saved in  
yolov4-custom-functions/checkpoints/yolov4-tiny-416 folder.

NOTE: If you have the error, "ModuleNotFoundError: No module named 'pytesseract'", close the terminal and open it again, and execute save\_model.py.



## 2. Execute YOLO v4 demo program

### 2.1. Activate the Anaconda environment

conda activate yolo4-gpu

### 2.2. Execute the demo program;

```
python detect_video.py --weights ./checkpoints/yolov4-tiny-416 --size 416 --model yolov4 --
```

```
video ./
```

```
data/vi
```

```
--tiny
```

deo/video.mp4

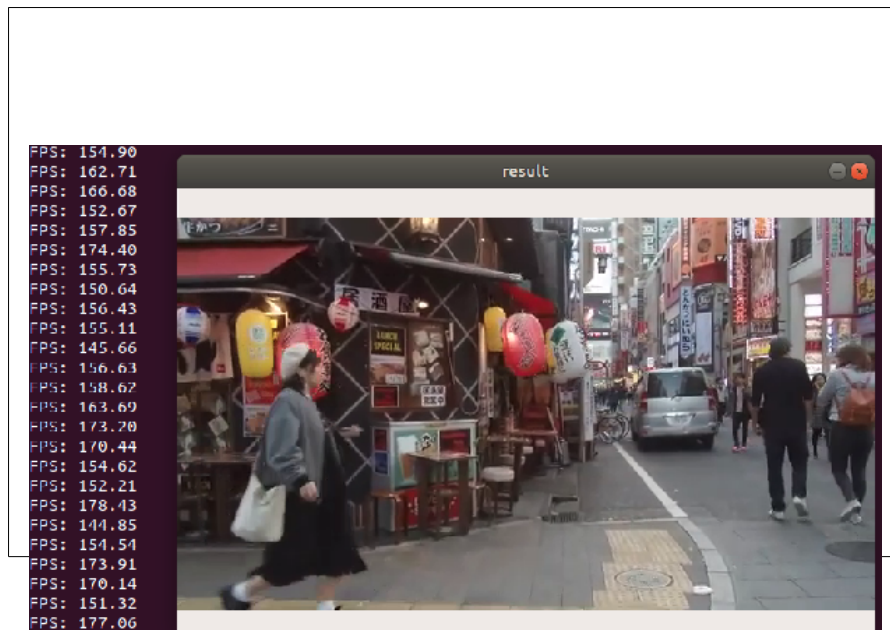


Figure 1. Original Image

## 3. Modify the demo program

### 3.1. Restrict classes to detect

Open detect\_video.py and change the following lines;

```
128     # by default allow all classes in .names file
129     # allowed_classes = list(class_names.values())
130
```



```
131     # custom allowed classes (uncomment line below to allow detections for only people)
132     allowed_classes = ['person', 'bicycle', 'car', 'motorbike', 'bus', 'truck']
```

NOTE: The class names are in yolov4-custom-functions/data/classes/coco.names

Approved for release for SJSU education use.

(END)