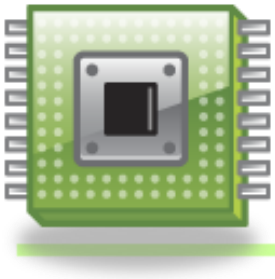
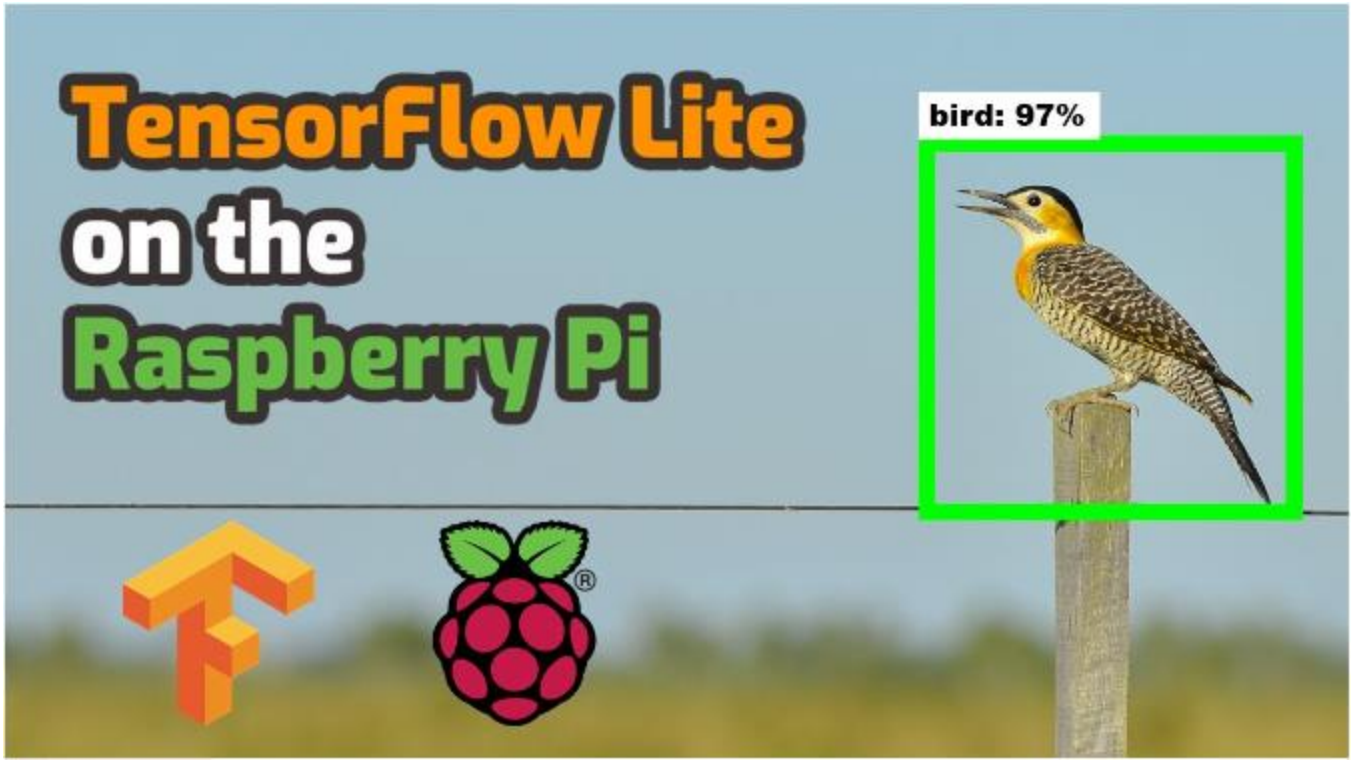


TFLite on Raspberrypi

Speaker : Tse-Yu Chen

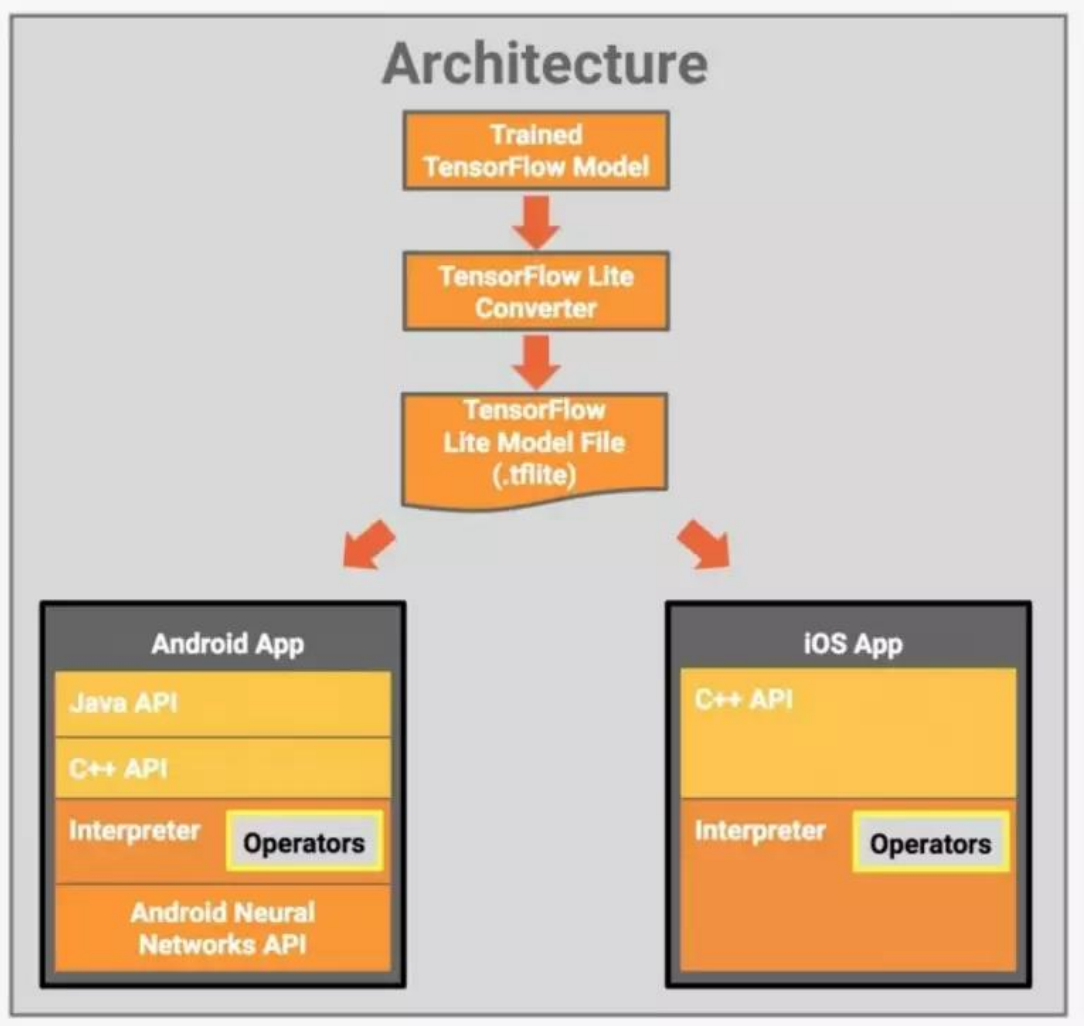
Advisor : Prof. Tong-Yu Hsieh

Install TFLite on Pi



Why TFLite (1/2)

- Lighter
- Faster



Why TFLite (2/2)

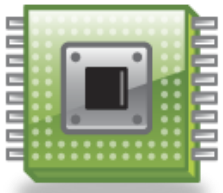
[1] “How To Run TensorFlow Lite on Raspberry Pi for Object Detection”, https://www.youtube.com/watch?v=aimSGOAUI8Y&t=12s

Install TFLite on Pi (1/5)

1. sudo apt-get update
2. sudo apt-get dist-upgrade
3. git clone https://github.com/EdjeElectronics/TensorFlow-Lite-Object-Detection-on-Android-and-Raspberry-Pi.git
4. mv TensorFlow-Lite-Object-Detection-on-Android-and-Raspberry-Pi tflite1
5. cd tflite1
6. bash get_pi_requirements.sh
7. wget https://storage.googleapis.com/download.tensorflow.org/models/tflite/coco_ssd_mobilenet_v1_1.0_quant_2018_06_29.zip

Install TFLite on Pi (2/5)

1. unzip coco_ssd_mobilenet_v1_1.0_quant_2018_06_29.zip -d Sample_TFLite_model
2. sudo apt-get install -y libhdf5-dev libc-ares-devlibeigen3-dev
3. python3 -m pip install keras_applications==1.0.8 --no-deps
4. python3 -m pip install keras_preprocessing==1.1.0 --no-deps
5. python3 -m pip install h5py==2.9.0
6. sudo apt-get install -y openmpi-bin libopenmpi-dev
7. sudo apt-get install -y libatlas-base-dev



Install TFLite on Pi (3/5)

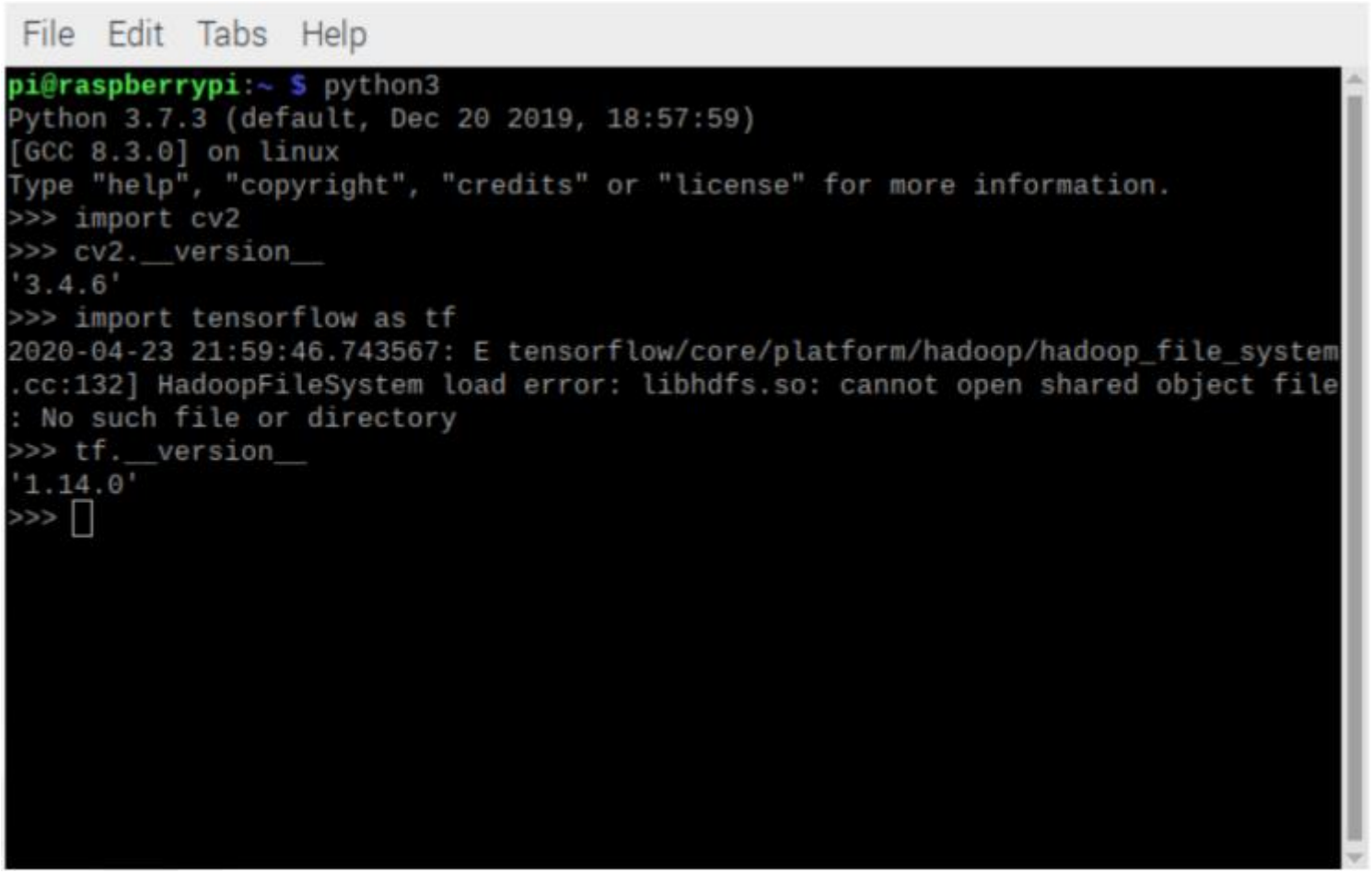
1. python3 -m pip install -U six wheel mock
2. wget https://github.com/lhelontra/tensorflow-on-arm/releases/download/v2.0.0/tensorflow-2.0.0-cp37-none-linux_armv7l.whl
3. python3 -m pip install tensorflow-2.0.0-cp37-none-linux_armv7l.whl

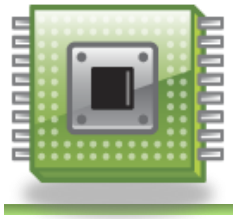
Install TFLite on Pi (4/5)

- **“can’t connect to archieve.raspberripi.org” problem happened in your updating**
  - sudo nano /etc/resolv.conf
  - Add “nameserver 8.8.8.8” in your document
  - Run the command “sudo apt-get update” again

Install TFLite on Pi (5/5)

- **Test your tflite and opencv module**



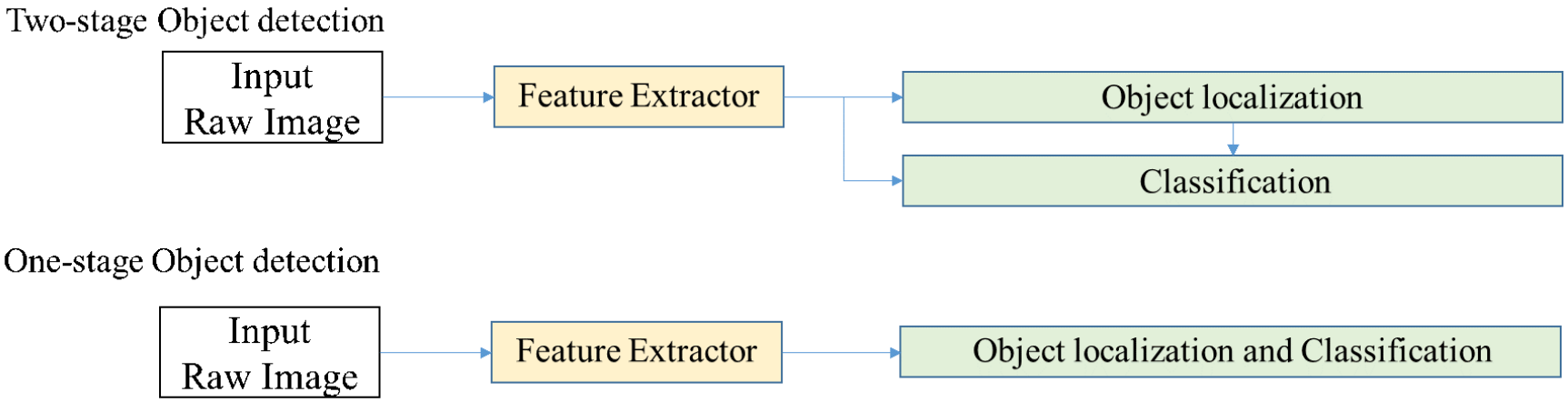


Single Shot MultiBox Detector (1/3)

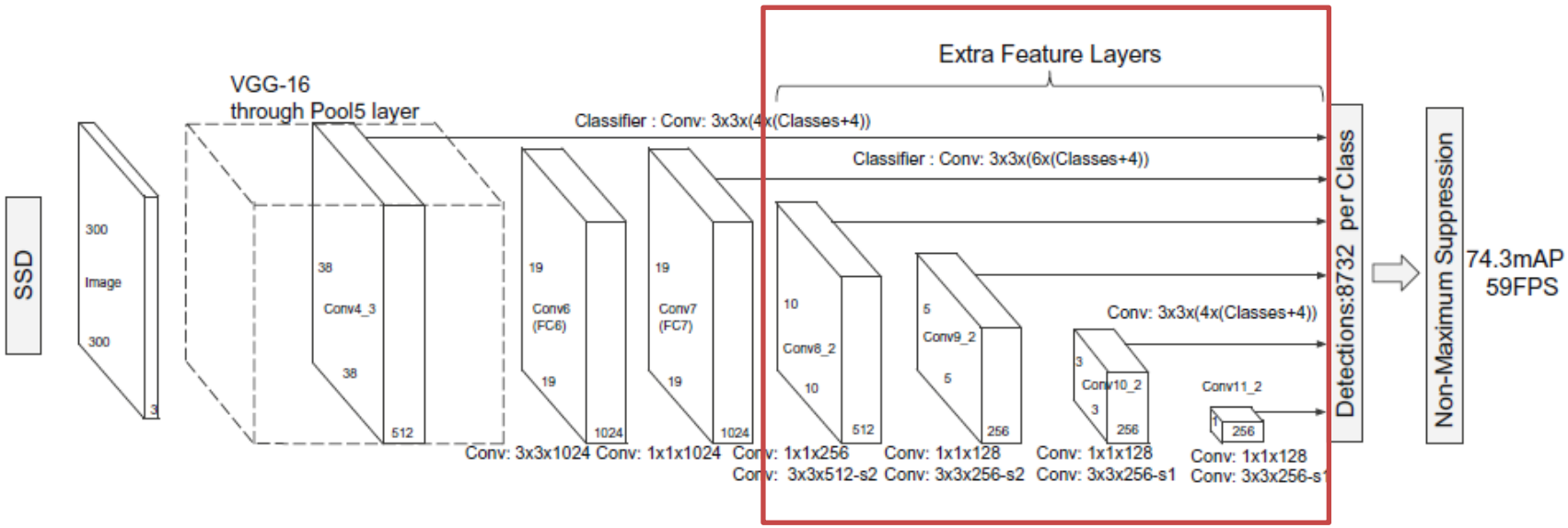
- **Faster than YOLO(v1) and has comparable accuracy of Faster R-CNN**
- **Much better accuracy with small input image size compare to other single stage methods**
- **Features**
  - **Multi-scale feature maps for detection**
  - **Convolutional predictors for detection**
  - **Boxes with different aspect ratios**

[2] W. Liu et al. , “SSD: Single Shot MultiBox Detector”, *2016 European Conference on Computer Vision(ECCV)*, pp. 21-37, 2016.

Single Shot MultiBox Detector (2/2)



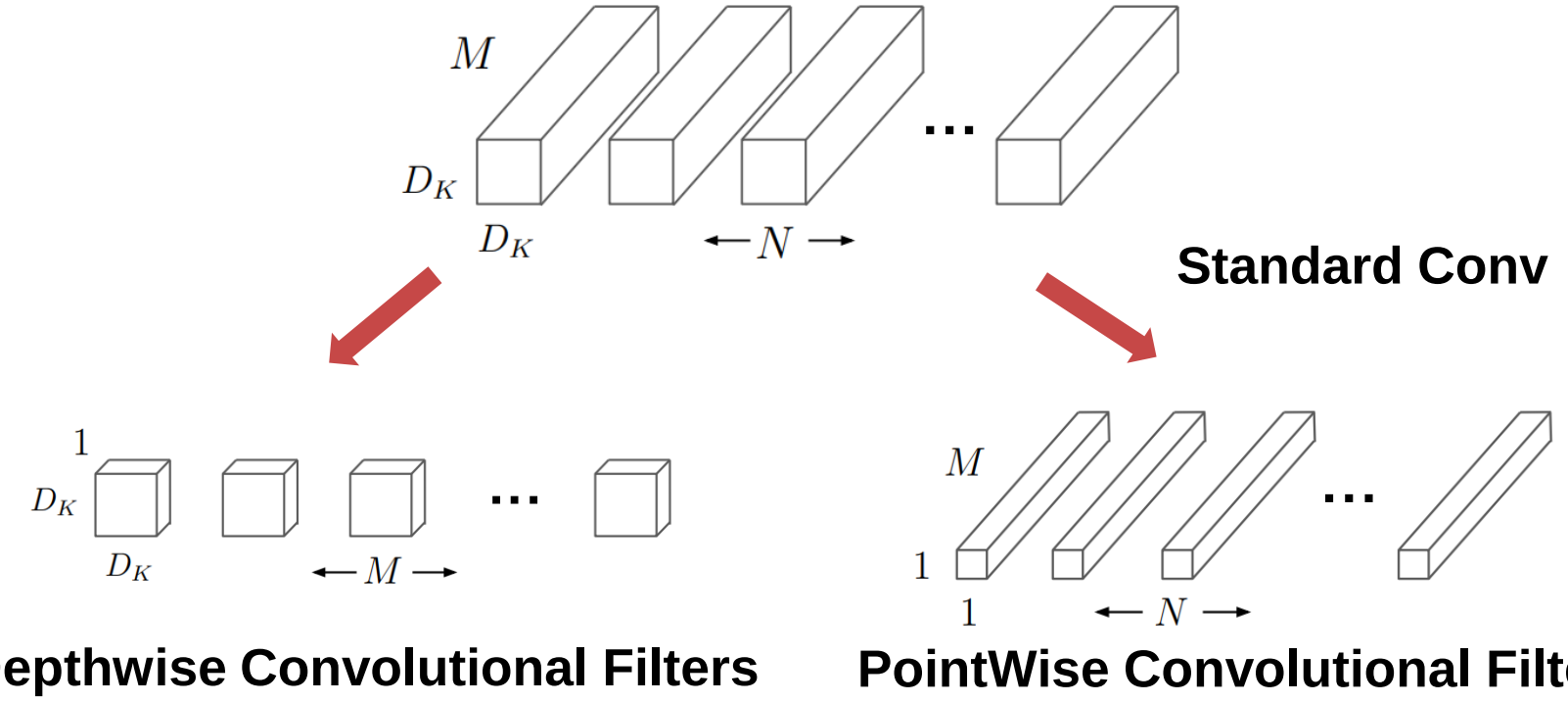
Single Shot MultiBox Detector (3/3)



Backbone (feature extractor): VGG16

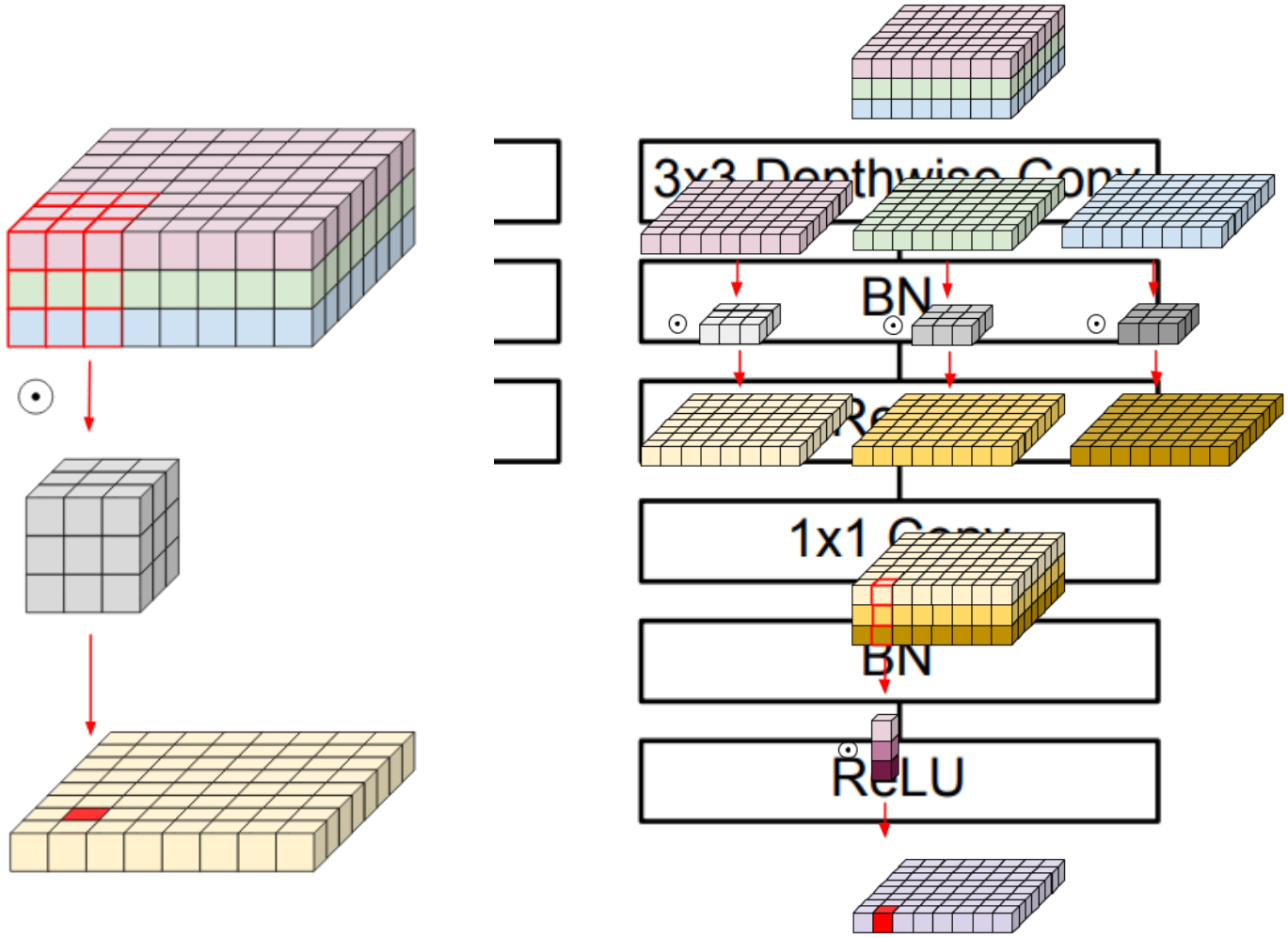
Mobilenet (1/2)

- **Less Calculation**
- **Similar performance**



[3] A. G. Howard et al. , “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”, *Google Inc. 2017*

Mobilenet (2/2)



[4] https://medium.com/@zurister/depth-wise-convolution-and-depth-wise-separable-convolution-37346565d4ec

Demo (1/4)

- **Real-time Detection**
  - python3 TFLite_detection_webcam.py --modeldir=Sample_TFLite_model
- **Image Detection**
  - python3 TFLite_detection_image.py --modeldir=Sample_TFLite_model
- **Video Detection**
  - python3 TFLite_detection_video.py --modeldir=Sample_TFLite_model

- **Press ‘q’ to exit**

Demo (2/4)

- Tflite models provided by tensorflow

Name	Latency(ms)	COCO mAP
ssd_mobiledet_cpu_coco	113 (on Pixel1)	24.0
ssd_mobilenet_v2_mnasfpn_coco	183 (on Pixel1)	26.6
ssd_mobilenet_v3_large_coco	119 (on Pixel1)	22.6
ssd_mobilenet_v3_small_coco	43 (on Pixel1)	15.4
ssd_mobiledet_dsp_coco	12.3 (on Pixel4 DSP)	28.9(fp32) / 28.8(uint8)

https://drive.google.com/drive/folders/1iWeGJc1a-lJKW7ibEbpS99eaSRi_EHiT?usp=sharing

Demo (3/4)

- Change File directory by adding “--modeldir=XX” to command line,

  e.g. --modeldir=Sample_TFLite_model

- Change tflite model by adding “--graph=XX” to command line,

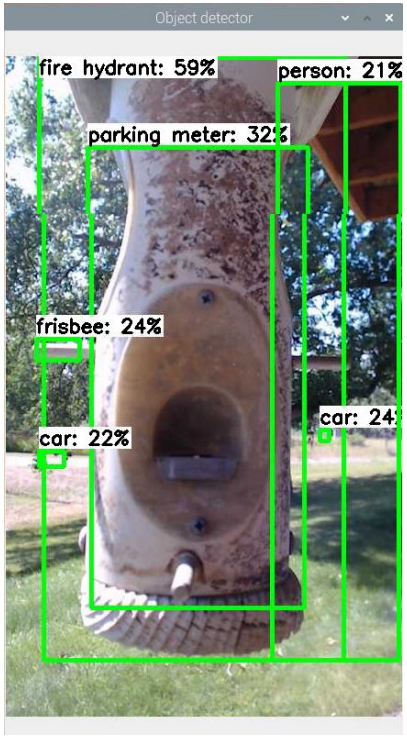
  e.g. --graph=ssd_mobilenet_v2_cpu.tflite

- Change video by adding “--video=XX” to command line,

  e.g. --video=sample.mp4

Demo (4/4)

- Change threshold by adding “--threshold XX” to command line, e.g. --threshold 0.5 (default)



Fps of Detection (1/2)

- Modify the python code
  - [line116]

    Add “frame_rate_calc = 1”

    Add “freq = cv2.getTickFrequency()”
  - [line127]

    Add “t1 = cv2.getTickCount”
  - [line162]

    Add “cv2.putText (frame,’FPS: {0: .2f}’.format (frame_rate_calc , (30,50), cv2.FONT_HERSHEY_SIMPLEX,1,(255,255,0),2 ,cv2.LINE_AA)”

    Need to TAB before adding the line.

Fps of Detection (2/2)

- **[line 171]**

  Add “t2 =cv2.getTickCount()”

  Add “time = (t2-t1)/freq”

  Add “fram_rate_calc = 1/time”

cv2.putText
(image, words, coordinate, font, color, line width, type of line)

Appendix

- 【How to setup your tensorflow on your PC】
  - https://github.com/EdjeElectronics/TensorFlow-Object-Detection-API-Tutorial-Train-Multiple-Objects-Windows-10
- 【Tensorflow Lite on PI】
  - https://github.com/EdjeElectronics/TensorFlow-Lite-Object-Detection-on-Android-and-Raspberry-Pi#step-1-train-quantized-ssd-mobilenet-model-and-export-frozen-tensorflow-lite-graph
- 【Tensorflow Model Zoo】
  - https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md