Object Detection with Caffe and Movidious Stick on Raspberry pi

Submitted by
Abhinav Kumar Challa- 31593
Venkat Musunuru- 31660
Vinod Alase- 31634

June 29, 2019

Contents

Chapter 1

Object Detection

1.1 Introduction

In recent years, Many advancements in the fields of Image Processing, Artificial Intelligence have made significant changes in our lifestyle and these advancements have increased our reliance on Computers. Either it is Object detection in the field of Autonomous driving or Working of a service Robot installed in our home this area has made our life more comfortable however Computer Vision is having huge significance as it extends its contribution in every possible application we can imagine in the area of Artificial Intelligence.

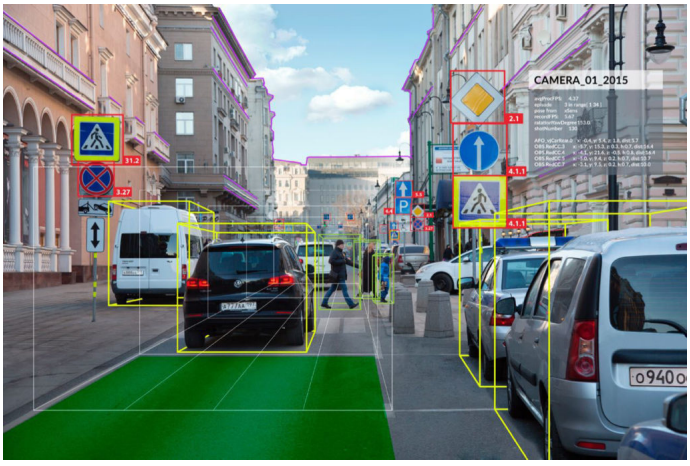

Figure 1.1: Object and Lane Detection on a busy street

Object Detection as the name indicates, is the art of detecting instances of Objects like Animals, Pedestrians, Automobiles in an image or video. It is one of the major application in the field of Computer Vision. As the applications increase the complexity also increases for example, Object classification is straight forward which is acheived by assigning class labels to an image where as Object detection draws a bounding

box around each object of interest and assigns them a class label.

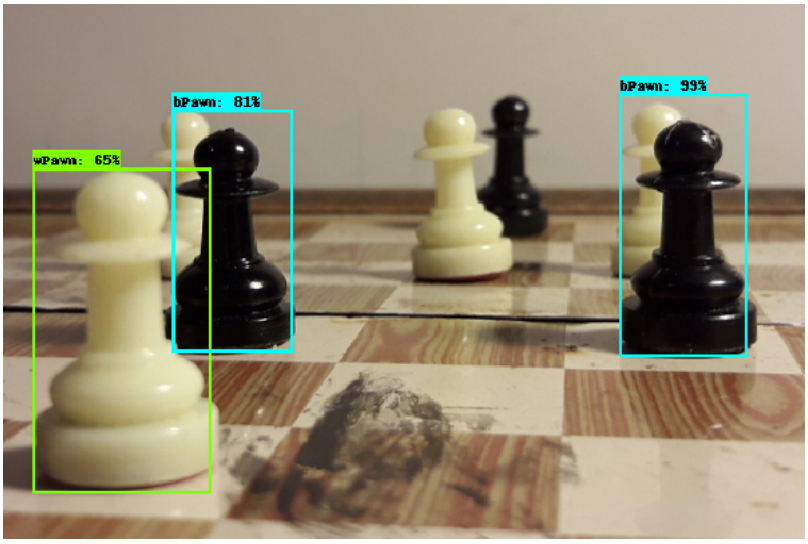

Figure 1.2: Object Detection on Chess board with Labels

1.1.1 Purpose

Object detection (OD) is an important cue in developing products for many domains: Digital Security and Surveil-lance, Autonomous vehicles, etc. Usually, complex solutions solve multiple tasks in parallel, so it is essential to have a fast and accurate algorithm.The main purpose of object detection in ADAS applicaton is to detect all instances of objects from a known class, such as people,cars or trucks in an image.Basically to get know about the surroundings.

1.1.2 Goal

Object Detection with Caffe and Movidius stick on Raspberry pi
The main task of project is Object detection with caffe using intel-movidius stick on rasberry pi processor.It also involves to increase accuracy of predection with higher speed.The other peculiar task was to maximizing the number of frames per second can be achieved using different models.

1.2 Hardware

1.2.1 Raspberry-Pi

A Raspberry-Pi is a single board computer used for scientific applications. The Raspberry Pi 3+ uses a Broadcom BCM2837B0 SoC with a 1.4 GHz 64-bit quad-core ARM Cortex-A53 processor, with 512 KB

shared L2 cache. It also supports Wireless connectivity. It has about 40 GPIO pins.

1.2.2 Camera Module

We are currently using a Raspberry Pi Camera Module version 1, It has a Still resolution of about 5 MegaPixels with a Video Modes of 1080p30, 720p60 and 640 ÃŮ 480p60/90. It accepts JPEG (accelerated), JPEG + RAW, GIF, BMP, PNG, YUV420, RGB888 formats.

1.2.3 Intel Movidius Stick

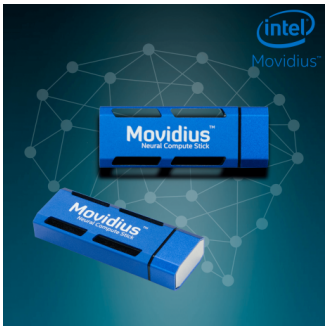

Figure 1.3: intel-movidius neural compute stick

The Intel Movidius Neural Compute Stick (NCS) is a tiny fan less deep learning device that can be used to learn AI programming at the edge. NCS is powered by the same low power high performance Intel Movidius Vision Processing Unit (VPU). which has the convolution neural network of frameworks which is utilized for object detection which requires setting up of software tools to compile profile and validate Deep Neural Networks (DNNs) as well as Neural Compute API on the NCS. We use the NCSDK.

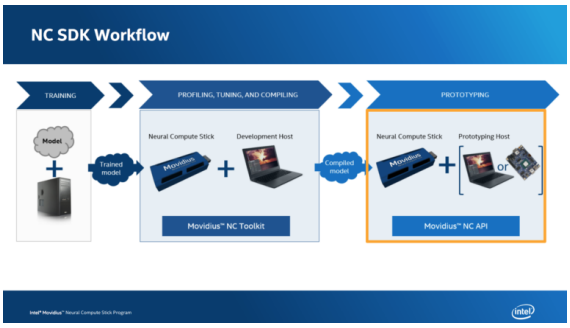

Figure 1.4: Work-Flow of NCS

The NCSDK includes a set of software tools to compile, profile, and validate DNNs as well as the Intel Movidius Neural Compute API known as NCAPI for application development in C/C++ or Python.

Chapter 2

Hardware Setup

2.1 Setup

The object detection using rasberry pi with intel NCS is as follows.

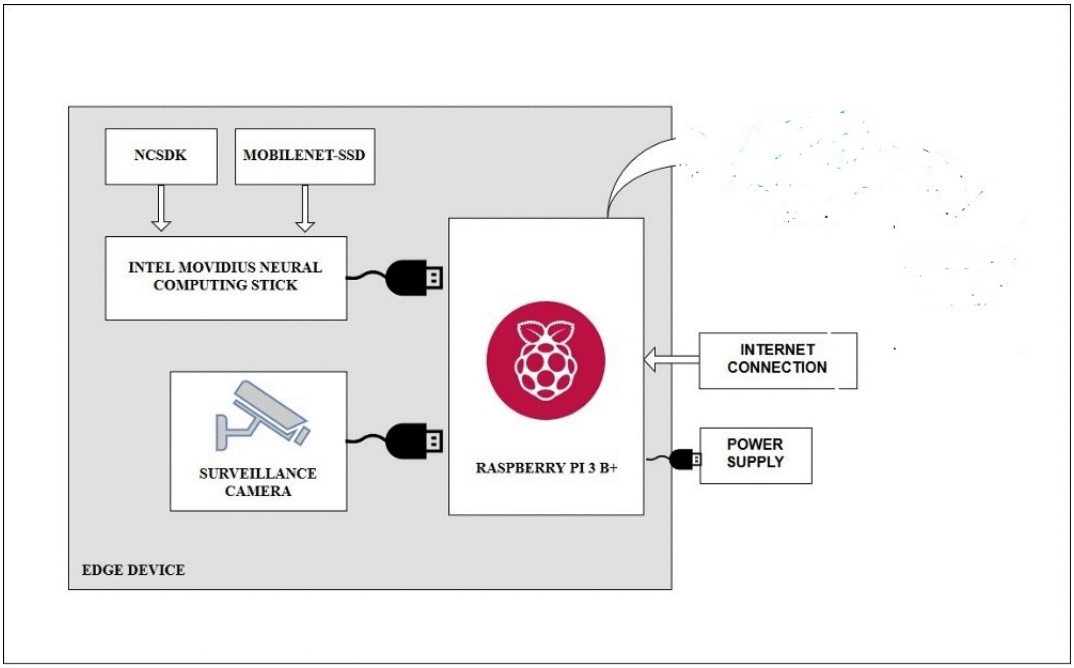

Figure 2.1: Block Diagram

Figure 2.2: Hardware setup

The pi camera module has to be attached rasberry pi module section.Neural computing stick has to be inserted in the one of the USB ports.Finaly whole circuit need to be supplied with power supply.

Chapter 3

Software Design

A object Detection involves finding whether there is object present or not secondly which type of object is present and how many objects are present. Basically, object presence needs to be detected after detecting a object it has to be classified. Classification is the main part which means what type of object it is (car, bus, bike,bottle,TV-monitor etc.).Implementation of object detection by camera, it finds primary application in ADAS applications.

The Edge device (Raspberry Pi, Intel Movidius and a camera) requires no internet connection for object detection utilizes internet only for connection to the cloud for real time data transfer.The detection happens on the rasberry pi device itself.

3.1 Use of Mobile Net-SSD Model Framework

Mobile Nets SSD (Single Shot Multibox Detection) is an Efficient convolution Neural Network architecture for mobile and embedded vision applications. MobileNets are based on a streamlined architecture that uses depth wise separable convolutions to build light weight deep neural networks. Mobile Net is an architecture which is more suitable for mobile and embedded based vision applications where there is lack of compute power.

This architecture uses depth wise separable convolutions which significantly reduces the number of parameters when compared to the network with normal convolutions with the same depth in the networks. This results in light weight deep neural networks. The normal convolution is replaced by depth wise convolution followed by point wise convolution which is called as depth wise separable convolution. By using depth wise separable convolutions, there is some sacrifice of accuracy for low complexity deep neural network. Employing Single Shot Multi-Box Detection compensate that and improves accuracy as well.

- **Single Shot:** this means that the tasks of object localization and classification are done in a single forward pass of the network
- **Multi-box:** this is the name of a technique for bounding box regression developed by Szegedy et al.

- **Detector:** The network is an object detector that also classifies those detected objects.

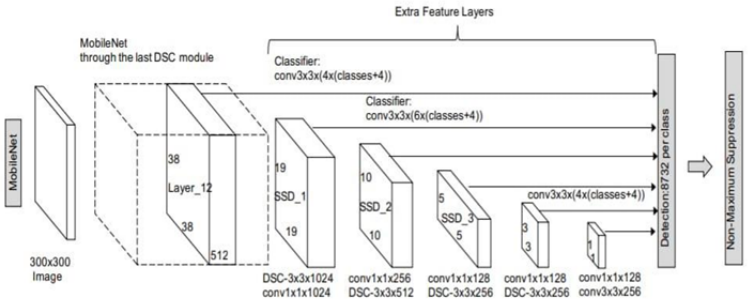

Figure 3.1: Block Diagram Of MobileNet-SSD Architecture

3.2 Algorithm Implemetation

- **Step 1:** Get the objects from the camera's field of view.Simentenoulsy instantiate to FPS counter.
- **Step 2:** Pre-process the image and feed it to the Network.It will detect the object with pre-trained models.
- **Step 3:** The network will predict the object and also classifies those detected objects only if the predection rate is greater than threshold value.
- **Step 4:** Once the network classfies the corresponding objects then the bounding box will be drawn around the objects.
- **Step 5:** FPS will be updating continuosly and displayed.The bounding box value will be displayed and published on ROS topic.

Chapter 4

Results

Using Mobile Net-SSD model which trained on caffe model we were able to detect the objects with higher efficiency.With available resources and pre-trained models we were able to detect objects of **20 type** i,e 20 object classes.Efficiency of the object detection was quiet outstanding and achieved around **7** FPS was remarkable.

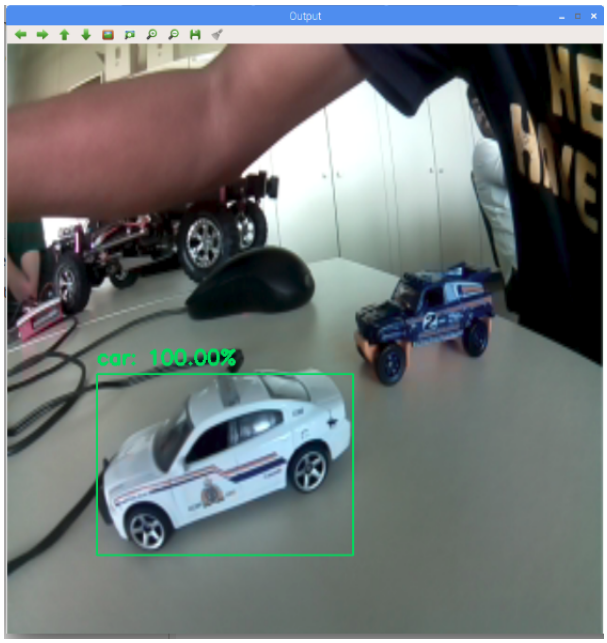

Figure 4.1: object detection

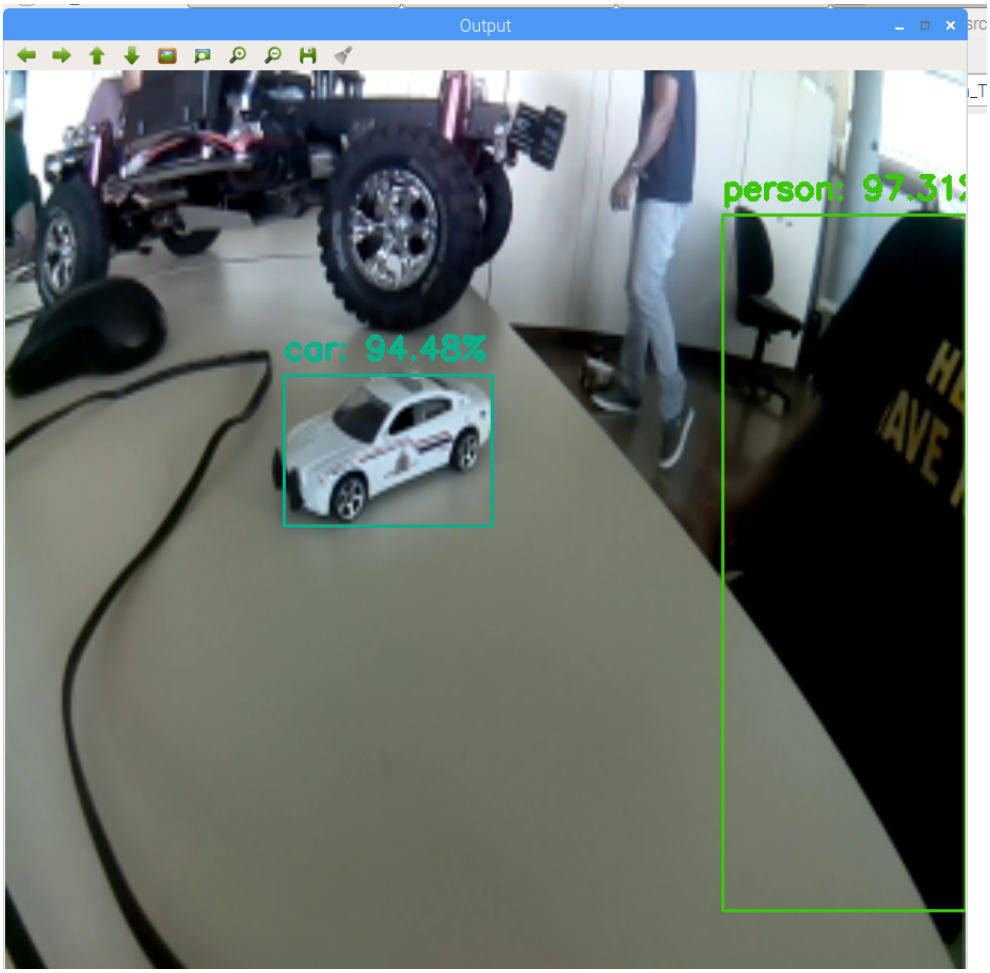

Figure 4.2: object detection with more objects

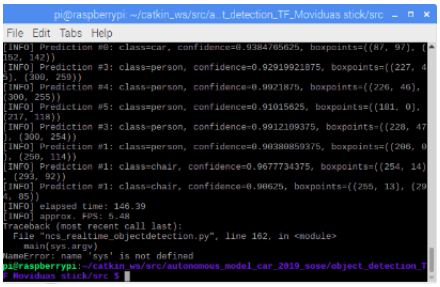

Figure 4.3: obtaining of FPS

The NCS can propel the Raspberry Pi to a 6.88x speedup over the standard CPU object detection!

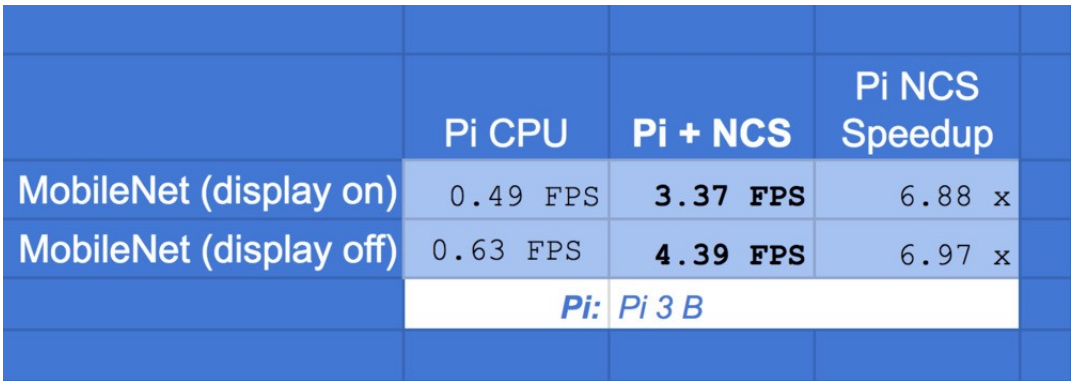

Figure 4.4: Object detection results on the Intel Movidius Neural Compute Stick (NCS) when compared to just the Raspberry Pi (CPU). The NCS helps the Raspberry Pi to achieve a 6.88x speedup.

Chapter 5

Conclusion and further Improvements

MobileNet-SSD vs YOLO model Comparision: When both the models are trained on a COCO dataset (330K Images, 80+ object) following results are obtained.

Type Of Model	Size On Disk	Detection Speed
YOLO_V2	269.9MB	2-3 fps
MobileNet	27.5MB	6-8 fps

Figure 5.1: comparisions of YOLO and Mobile-SSD

Thus for implementing on a low power device MobileNet-SSD is more suitable therefore, MobileNet-SSD framework has been installed on the Movidius Neural Computing Stick. Using pre-trained models it is able to achieve higher efficiency and FPS.As part of enhancements class names and bounding box value can be used in sensor fusion.This will increase the real time object detection with more precision.

Chapter 6

Reference

- P. F. Felzenszwalb, R. B. Girshick, D. Mcallester, and D. Ramanan, âĂIJObject detection with discriminatively trained part-based models,âĂİ
- C. Chen, A. Seff, A. L. Kornhauser, and J. Xiao, âĂIJDeepdriving: Learning affordance for direct perception in autonomous driving,âĂİ in ICCV, 2015
- X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, âĂIJMulti-view 3d object detection network for autonomous driving,âĂİ in CVPR, 2017.
- A. Dundar, J. Jin, B. Martini, and E. Culurciello, âĂIJEmbedded streaming deep neural networks accelerator with applications,âĂİ IEEE Trans. Neural Netw. Learning Syst., vol. 28, no. 7, pp. 1572âĂŞ1583, 2017.
- https://medium.com/intel-student-ambassadors/object-detection-a-comparison-of-performance-of-deep-learning-models-on-edge-using-intel-f66eb7f45b17.
- https://mc.ai/object-detection-a-comparison-of-performance-of-deep-learning-models-on-edge-using-intel-2/
- https://www.pyimagesearch.com/2018/02/19/real-time-object-detection-on-the-raspberry-pi-with-the-movidius-ncs/
- https://software.intel.com/en-us/neural-compute-stick/documentation/view-all?