NYU TANDON SCHOOL OF ENGINEERING

iSight

Reach Out and Grab It!

Seo Ho Kang
Nicholas Milton
Fabio Vulpi

Problem

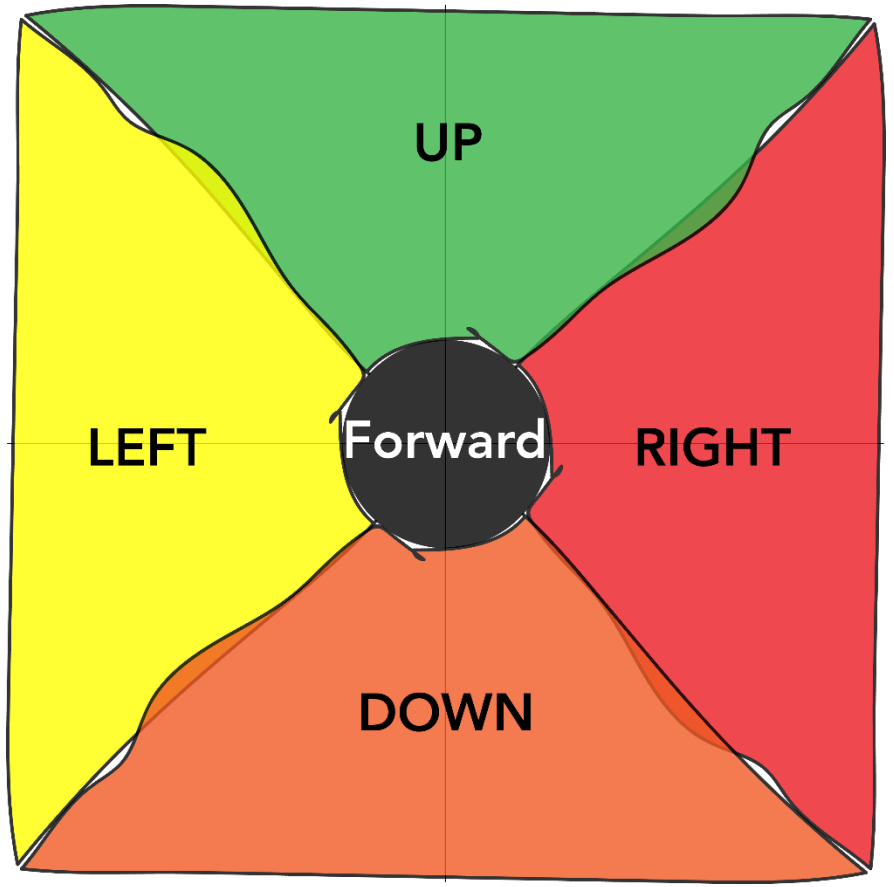
- Visually impaired people have difficulties to detect and retrieve objects in unknown environment.

Status Quo

- Orcam
- Aira
- Personal Assistance

User Requirements

- Fashionable
- Can pull it out only when you need it
- Lightweight
- Voice control like siri
- distinguish between coke and pepsi



Semester Goals

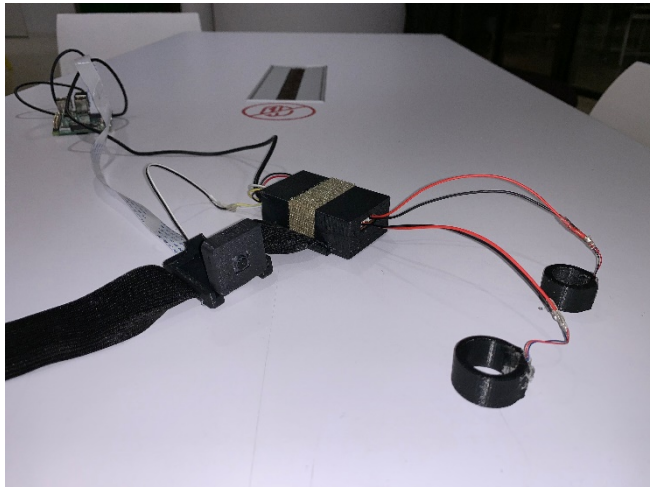
- Computer vision navigation
- Haptic feedback glove to guide hand
- Audio interface for object request

Actuation Hardware

rFduino on custom PCB on Glove

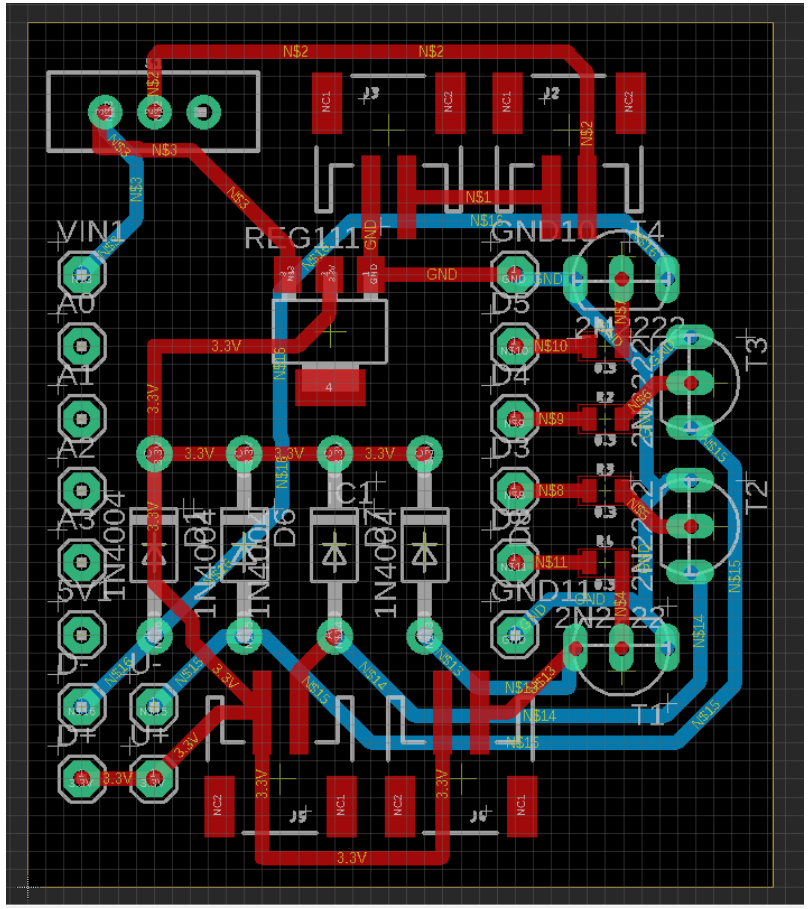
- Compact and lightweight
- Vibrating Motors protected
- Single wireless article
- Gazelle proprietary communication Protocolc

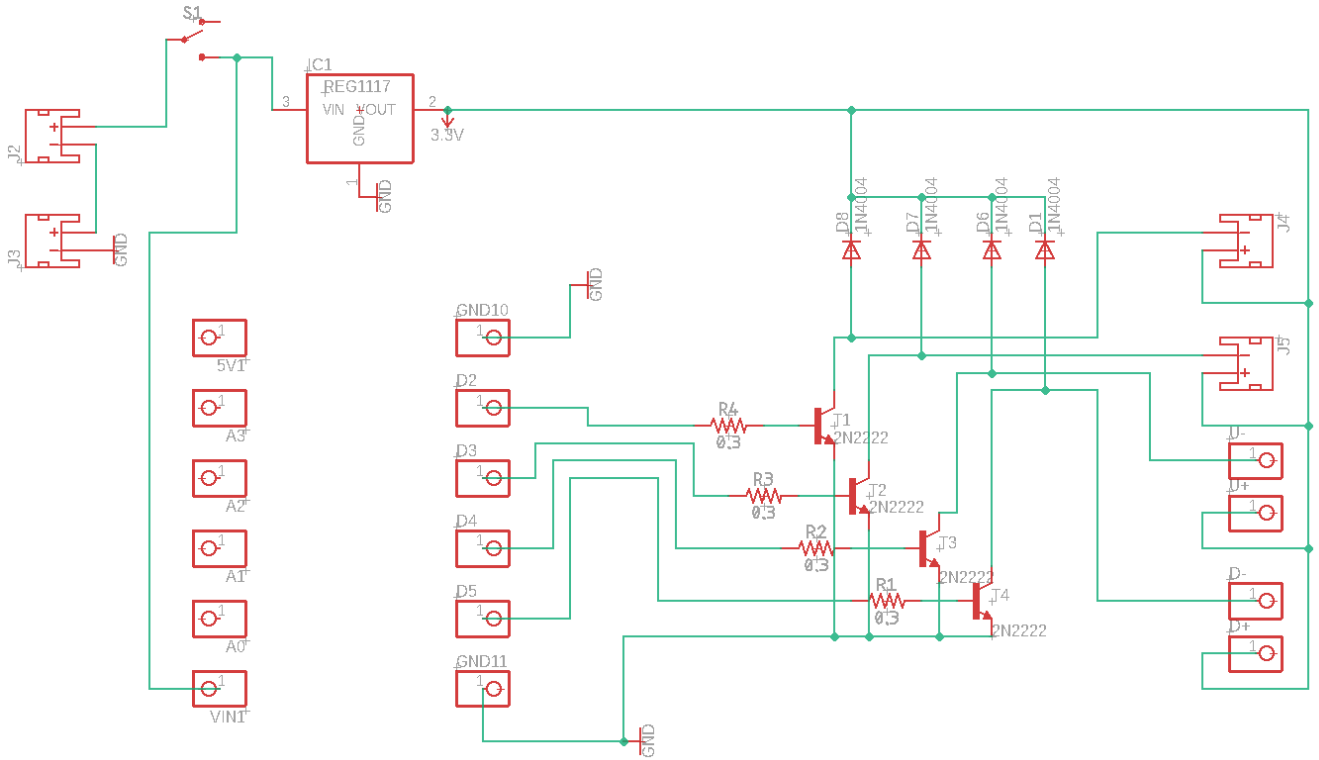
Bluno Beetle on watch



- Compact and lightweight
- Vibrating Motors in rings
- More Stable camera mount
- Arduino ide support

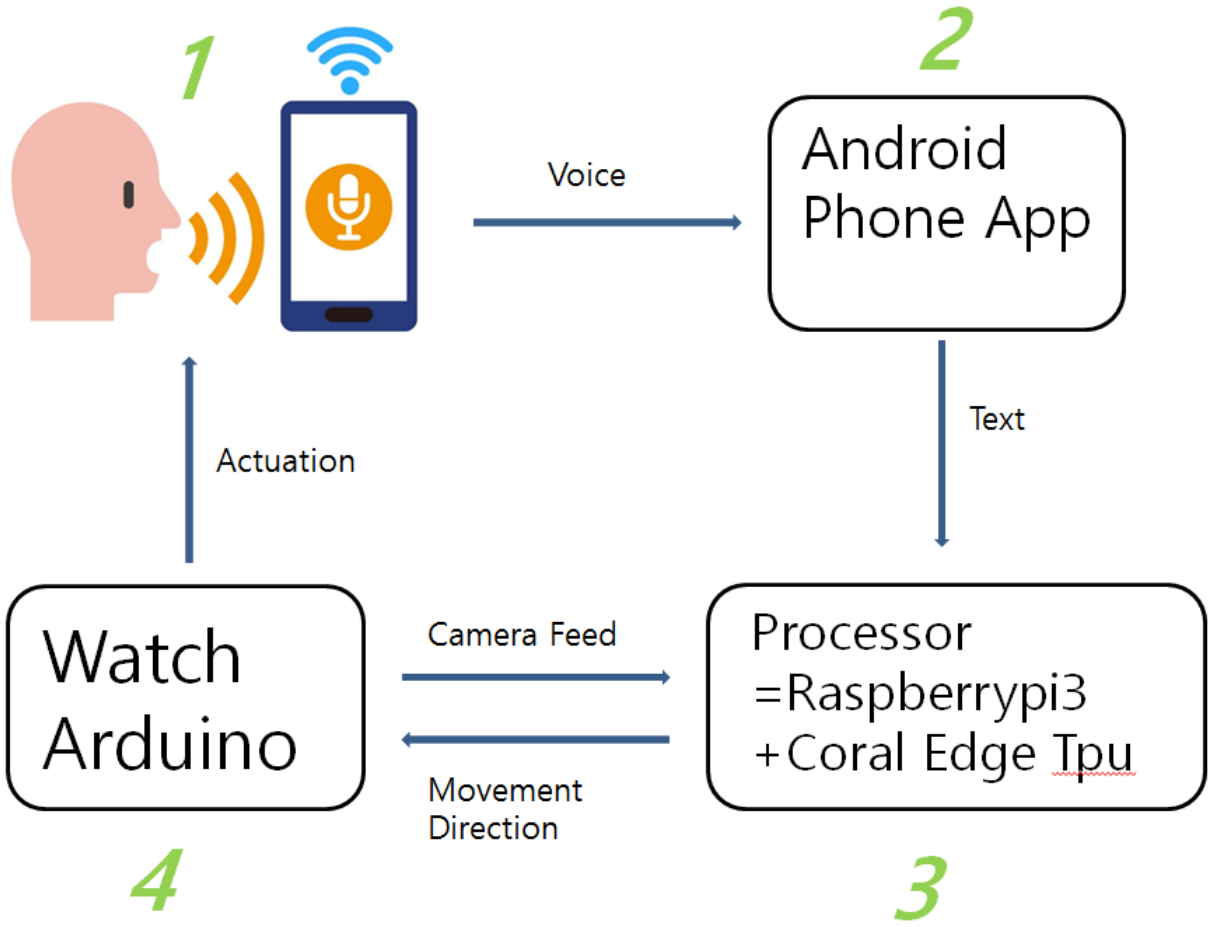
Design





PCB schematic and board made with Eagle software, showing Bluno Beetle, four identical motor control circuits (transistor, resistor, diode for back EMF) voltage regulator, switch and battery support.

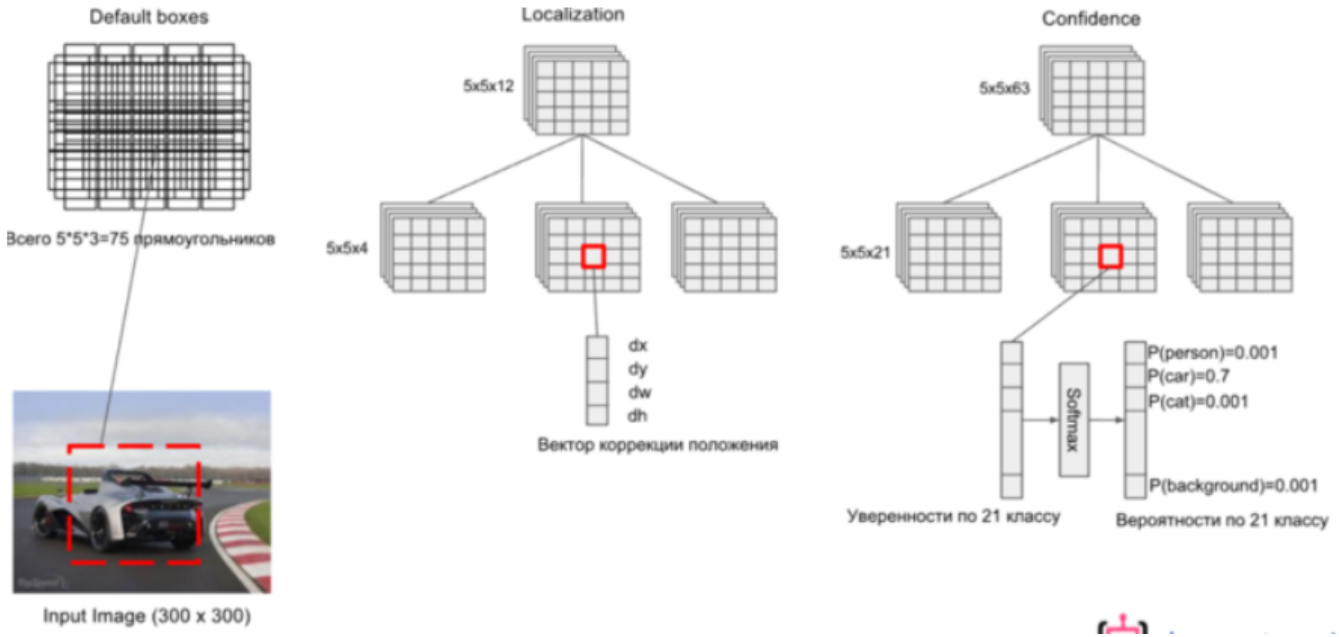
Software


1
2
Voice
Android
Phone App
Text
Actuation
Watch
Arduino
Camera Feed
Movement
Direction
Processor
=Raspberrypi3
+Coral Edge Tpu
4
3

Pronunciation Dictionary

```
GARAGE  G ER AA ZH
IN  IH N
IS  IH Z
OKAY    OW K EY
OPEN    OW P AH N
PI  P AY
RAMONA  R AH M OW N AH
SHUTDOWN    SH AH T D AW N
START   S T AA R T
THE DH AH
THE(2)  DH IY
TIME    T AY M
TRANSLATOR  T R AE N S L EY T ER
TRANSLATOR(2)   T R AE N Z L EY T ER
WEATHER W EH DH ER
WHAT    W AH T
WHAT(2) HH W AH T
```

Design


Default boxes
Всего 5*5*3=75 прямоугольников
Input Image (300 x 300)
Localization
5x5x12
5x5x4
dx
dy
dw
dh
Вектор коррекции положения
Confidence
5x5x63
5x5x21
Softmax
P(person)=0.001
P(car)=0.7
P(cat)=0.001
P(background)=0.001
Уверенности по 21 классу
Вероятности по 21 классу


75 детектируемых
Фильтрация по confidence
P(car)=0.6
P(car)=0.7

1-1) Run script with command below.

[Command]
python3 demo/object_detection_ver5.py \
 --model test_data/mobilenet_ssd_v2_coco_quant_postprocess_edgetpu.tflite \
 --label test_data/coco_labels.txt

2-1) FPS optimization-used parser to organize script
[Snap of object_detection_ver5.py]

```
 parser = argparse.ArgumentParser()
 parser.add_argument(
     '--model', help='Path of the detection model.', required=True)
 parser.add_argument(
     '--label', help='Path of the labels file.')
 parser.add_argument(
     '--input', help='File path of the input image.')
 parser.add_argument(
     '--output', help='File path of the output image.')
 args = parser.parse_args()
```

2-2) FPS optimization-use array input

We chose to input image array to “mobilenet_ssd_v2” directly after flatten the image. We previously input image but after changing it to input array, FPS increased by 5~10. Flatten step was done because Google TF lite api asked us to put flatten image.

Plus, in order to get fast FPS you should follow the way we implemented with pi-camera.

```
camera.framerate = 60
raw_capture = PiRGBArray(camera, size=(IM_WIDTH, IM_HEIGHT))
raw_capture.truncate(0)
start = time.time()
counter = 0
for frame1 in camera.capture_continuous(raw_capture, format='bgr', use_video_$
  frame = frame1.array.flatten()

  ans = engine.DetectWithInputTensor(frame)
  raw_capture.truncate(0)
```

3-1) Communication (Android app <-> Raspberry Pi3)
We use socket serial communication between app and rpi3. App converts voice to text and send it to raspberry pi3.

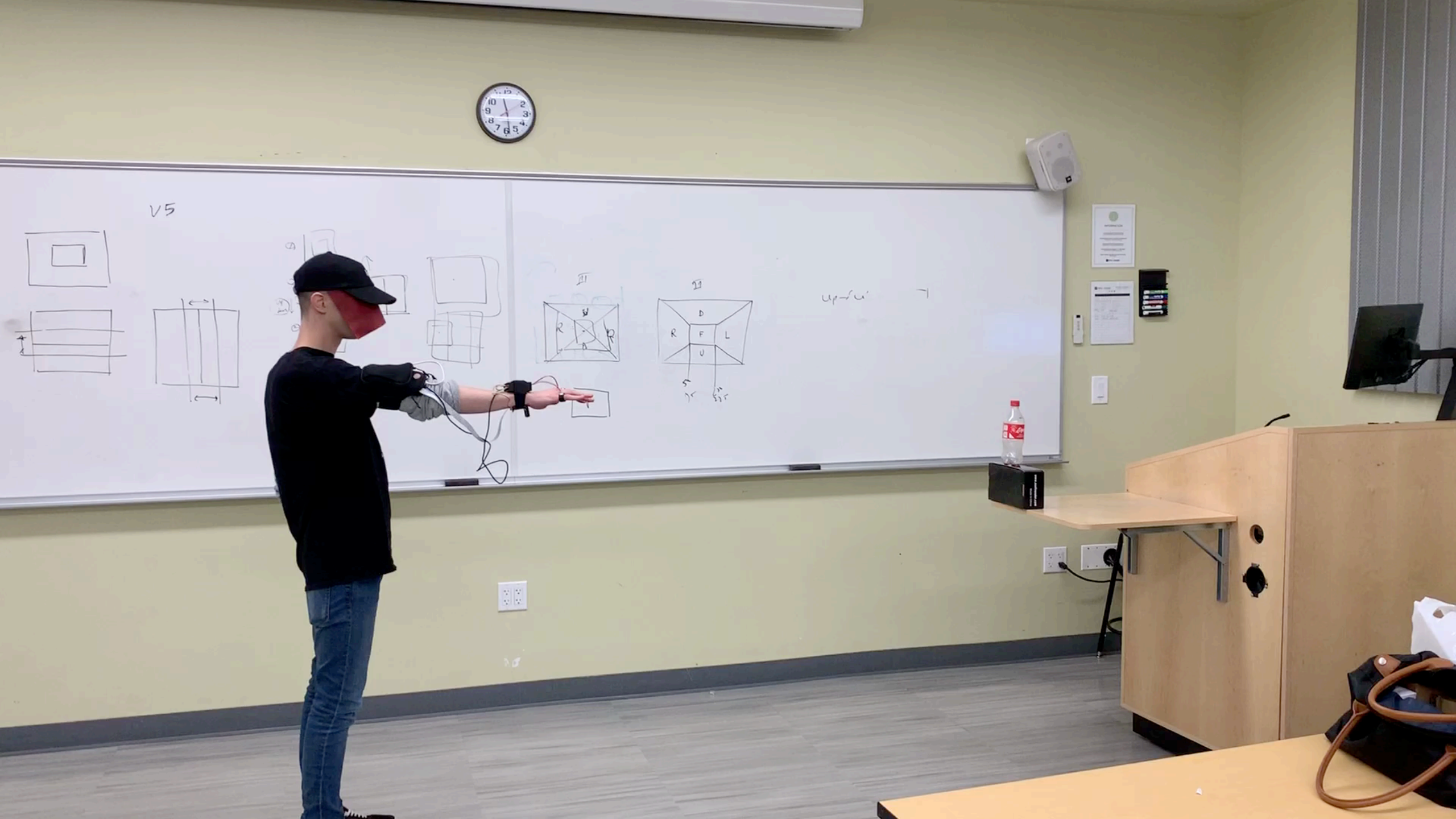
```
from socket import *
from time import ctime
import time
import RPi.GPIO as GPIO
import os

def setup():

    with open("coco_labels.txt") as mf:
        ta = mf.readlines()

    labels = [l[:-1] for l in ta]

    HOST = ''
    PORT = 21567
    BUFSIZE = 1024
    ADDR = (HOST,PORT)
    tcpSerSock = socket(AF_INET, SOCK_STREAM)
    tcpSerSock.bind(ADDR)
    tcpSerSock.listen(5)

#    return tcpSerSock

#def get_label(tcpSerSock):
    while True:
        print('Receiving data')
        tcpCliSock,addr = tcpSerSock.accept()
        print('Received data : \n', addr)
        try:
            while True:
                print("Now start receiving data")
                data = ''
                data = tcpCliSock.recv(BUFSIZE)
                print(type(data))
                data = str(data)
                print(type(data))
                if not data:
                    break
                for label in labels:
                    if label in data:
                        return label
                    elif "describe" in data:
                        os.system("python3 demo/object_detection2.py --model test_data/mobilenet_ssd_v2_coco_quant_postprocess_edgetpu.tflite --label test_data/coco_labels.txt")
                time.sleep(0.1)
        except:
            print("error")
```

3-2) Communication (Arduino <-> Raspberry pi3)
We use ttyACM0 serial communication.

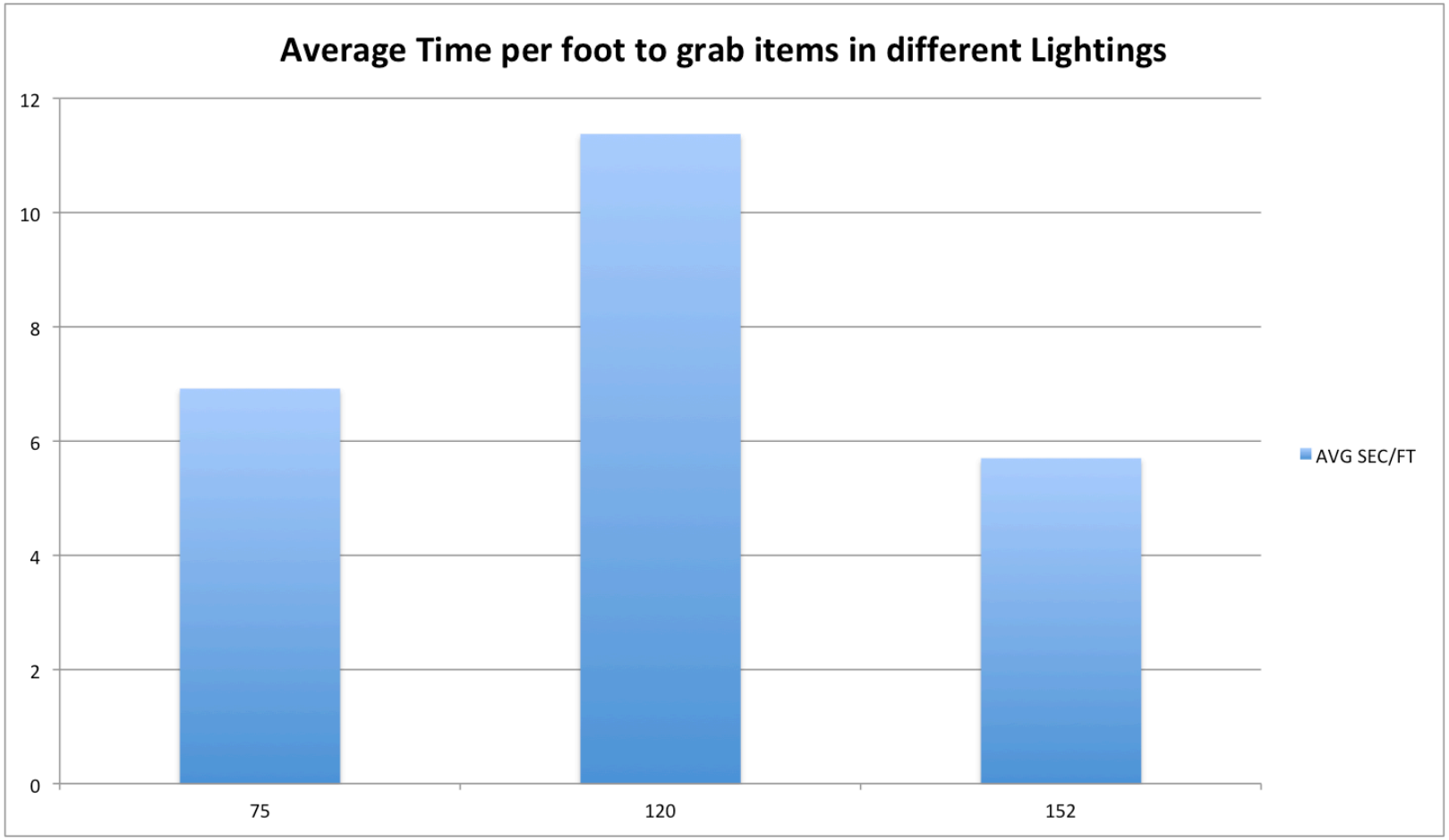
```
ser = serial.Serial("/dev/ttyACM0", 115200)
ser.baudrate = 115200
```

4) SSD Object detection model
Input is [300, 300] image array and we use label, and box (x,y) coordinate output. We use pretrained model.

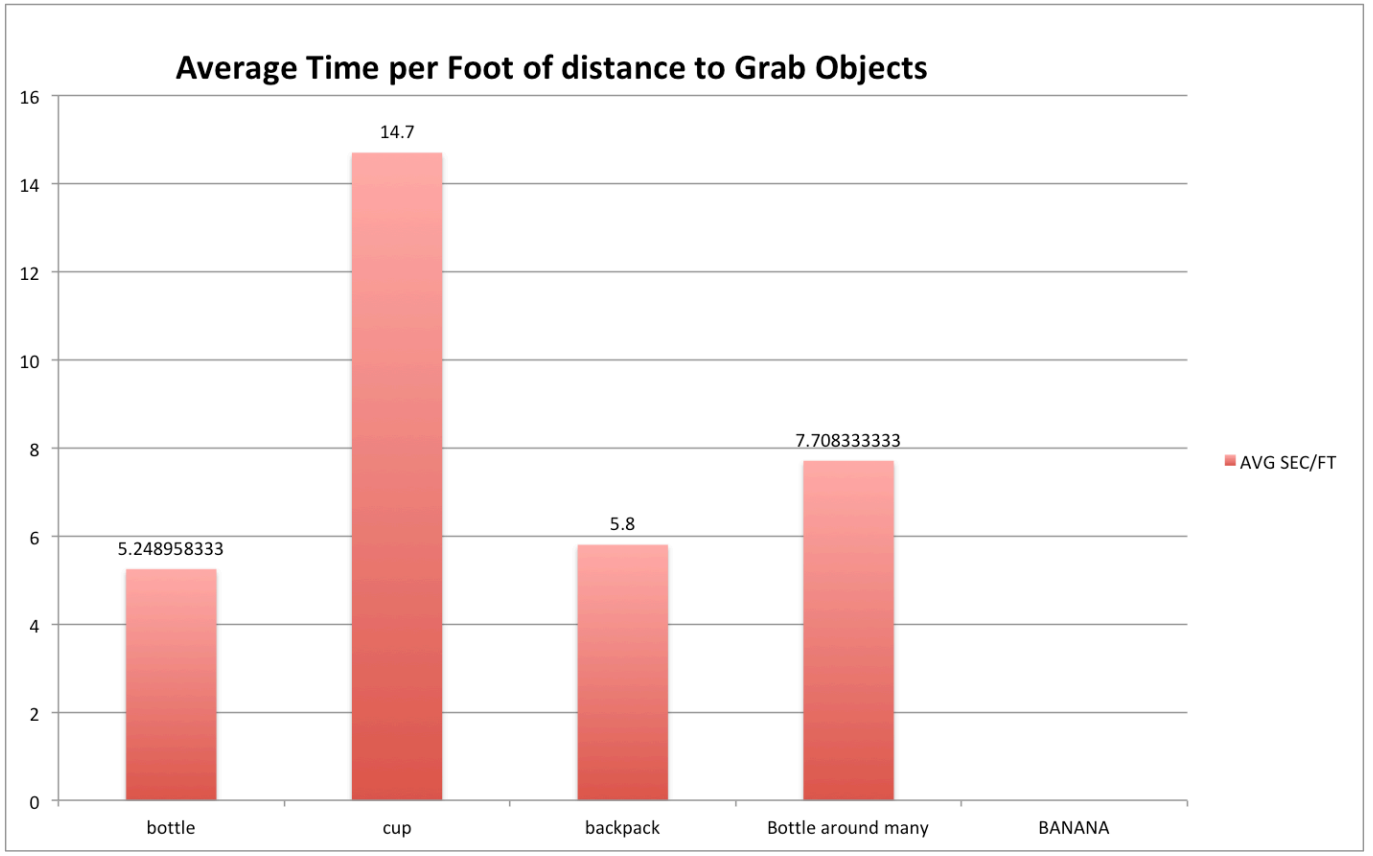
```
ans = engine.DetectWithInputTensor(frame)
raw_capture.truncate(0)
#print('captured %s' % filename)
frame1.truncate(0)
#rawCapture.seek(0)
counter += 1

ser = serial.Serial("/dev/ttyACM0", 115200)
ser.baudrate = 115200

if counter == 10000000:
  break
if ans:
  for obj in ans:
    if labels[obj.label_id] == lab:
      box = obj.bounding_box.flatten()
      box *= 300
      center_x = ((box[0] + box[2]) / 2)
      center_y = ((box[1] + box[3]) / 2)
```


V5
D
R
F
L
U

Object Detection Different Light

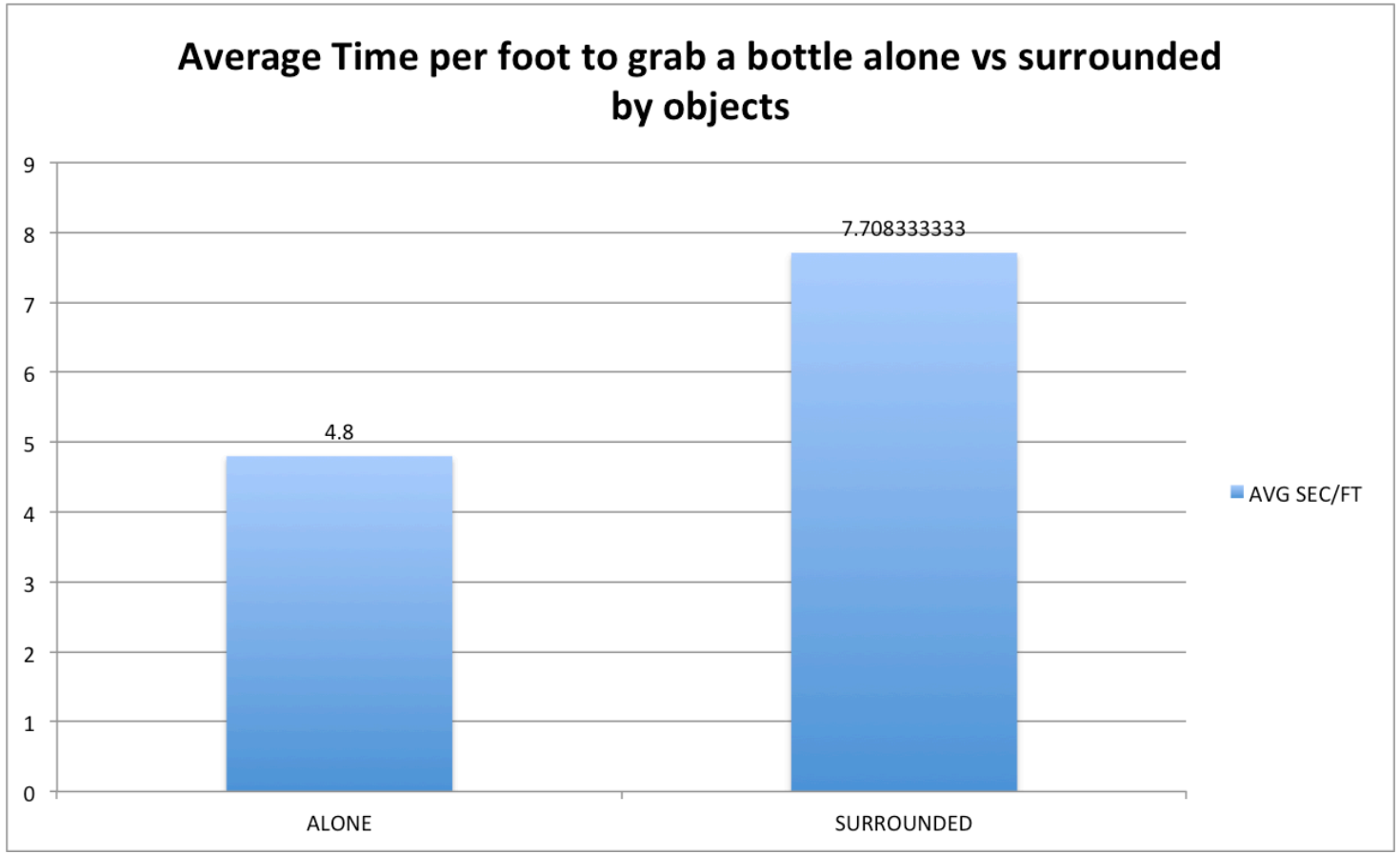


Object Detection Different Objects



lux

Object Detection Different Senario



lux