ASSIGNMENT-6

K.Sai Ganaamruth

19BCG10083

I used my laptop camera as an object detecting gadget.

```
import time
import numpy as np
import cv2
 car_classifier = cv2.CascadeClassifier('\haarcascade_car.xml')
cap = cv2.VideoCapture('/vehicle.mp4')
while cap.isOpened():
    time.sleep(.05)
   ret, frame = cap.read()
   gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
   cars = car_classifier.detectMultiScale(gray, 1.1, 2)
   for (x,y,w,h) in cars:
      cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 255), 2)
      cv2.imshow('Cars', frame)
if cv2.waitKey(1) == 13: #enter key
      break
cap.release()
cv2.destroyAllWindows()
```

<?xml version="1.0"?>
<!-- Add more negative to training. -->
<opencv_storage>
<cars3 type_id="opencv-haar-classifier">
  <size>
    20 20</size>
  <stages>
    <_>
      <!-- stage 0 -->
      <trees>
        <_>
          <!-- tree 0 -->
          <_>
            <!-- root node -->
            <feature>
              <rects>
                <_>
                  6 12 8 8 -1.</_>
                <_>
                  6 16 8 4 2.</_></rects>
              <tilted>0</tilted></feature>
            <threshold>0.0452074706554413</threshold>

 <left_val>-0.7191650867462158</left_val>

```
    <right_val>0.7359663248062134</right_val></_></_>
  <_>
   <!-- tree 1 -->
   <_>
    <!-- root node -->
    <feature>
     <rects>
      <_>
       1 12 18 1 -1.</_>
      <_>
       7 12 6 1 3.</_></rects>
     <tilted>0</tilted></feature>
    <threshold>-0.0161712504923344</threshold>
    <left_val>0.5866637229919434</left_val>
    <right_val>-0.5909150242805481</right_val></_></_>
  <_>
   <!-- tree 2 -->
   <_>
    <!-- root node -->
    <feature>
     <rects>
      <_>
       7 18 5 2 -1.</_>
      <_>
```

```
          7 19 5 1 2.</_></rects>


<tilted>0</tilted></feature>
      <threshold>0.0119725503027439</threshold>
      <left_val>-0.3645753860473633</left_val>
      <right_val>0.8175076246261597</right_val></_></_>
    <_>
     <!-- tree 3 -->
     <_>
      <!-- root node -->
      <feature>
       <rects>
        <_>
         5 12 11 4 -1.</_>
        <_>
         5 14 11 2 2.</_></rects>
       <tilted>0</tilted></feature>
      <threshold>0.0554178208112717</threshold>
      <left_val>-0.5766019225120544</left_val>
      <right_val>0.8059020042419434</right_val></_></_></trees>
  <stage_threshold>-1.0691740512847900</stage_threshold>
  <parent>-1</parent>
  <next>-1</next></_>
```

```
<_>
    <!-- stage 1 -->
    <trees>
     <_>
      <!-- tree 0 -->
      <_>
       <!-- root node -->
       <feature>
        <rects>
         <_>
          1 12 18 2 -1.</_>
         <_>
          7 12 6 2 3.</_></rects>
        <tilted>0</tilted></feature>
       <threshold>-0.0243058893829584</threshold>
       <left_val>0.5642552971839905</left_val>
       <right_val>-0.7375097870826721</right_val></_></_>
     <_>
      <!-- tree 1 -->
      <_>
       <!-- root node -->
       <feature>
        <rects>
```

<_>

3 1 14 6 -1.</_>

<_>

3 3 14 2 3.</_></rects>

<tilted>0</tilted></feature>

<threshold>-0.0302439108490944</threshold>

<left_val>0.5537161827087402</left_val>

<right_val>-0.5089462995529175</right_val></_></_>