CS 483 HW7

November 30, 2021

[1]:
```
#install opencv library
!pip install opencv-python
import cv2
```

```
Requirement already satisfied: opencv-python in
c:\users\huyvu\anaconda3\lib\site-packages (4.5.4.60)
Requirement already satisfied: numpy>=1.17.3 in
c:\users\huyvu\anaconda3\lib\site-packages (from opencv-python) (1.19.5)
```

[2]:
```
import matplotlib.pyplot as plt
```

[3]:
```
#config files downloaded from github.opencv wiki
config_file = 'ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt'
frozen_model = 'frozen_inference_graph.pb'
```

[4]:
```
model = cv2.dnn_DetectionModel(frozen_model,config_file)
```

[5]:
```
#get all label that library can detect
classLabels = []
file_name = 'Labels.txt'
with open(file_name, 'rt') as fpt:
    classLabels = fpt.read().rstrip('\n').split('\n')
```

[6]:
```
print(classLabels)
```

```
['person', 'bicycle', 'car', 'motorbike', 'aeroplane', 'bus', 'train', 'truck',
'boat', 'traffic light', 'fire hydrant', 'stop sign', 'parking meter', 'bench',
'bird', 'cat', 'horse', 'dog', 'sheep', 'cow', 'elephant', 'bear', 'zebra',
'giraffe', 'backpack', 'umbrella', 'handbag', 'tie', 'suitcase', 'frisbee',
'skis', 'snowboard', 'sports ball', 'kite', 'baseball bat', 'baseball glove',
'skateboard', 'surfboard', 'tennis racket', 'bottle', 'wine glass', 'cup',
'fork', 'knife', 'spoon', 'bowl', 'banana', 'apple', 'sandwich', 'orange',
'broccoli', 'carrot', 'hot dog', 'pizza', 'donut', 'cake', 'chair', 'sofa',
'pottedplant', 'bed', 'diningtable', 'toilet', 'tvmonitor', 'laptop', 'mouse',
'remote', 'keyboard', 'cell phone', 'microwave', 'oven', 'toaster', 'sink',
'refrigerator', 'book', 'clock', 'vase', 'scissors', 'teddy bear', 'hair drier',
'toothbrush']
```

[7]:
```
print(len(classLabels))
```

80

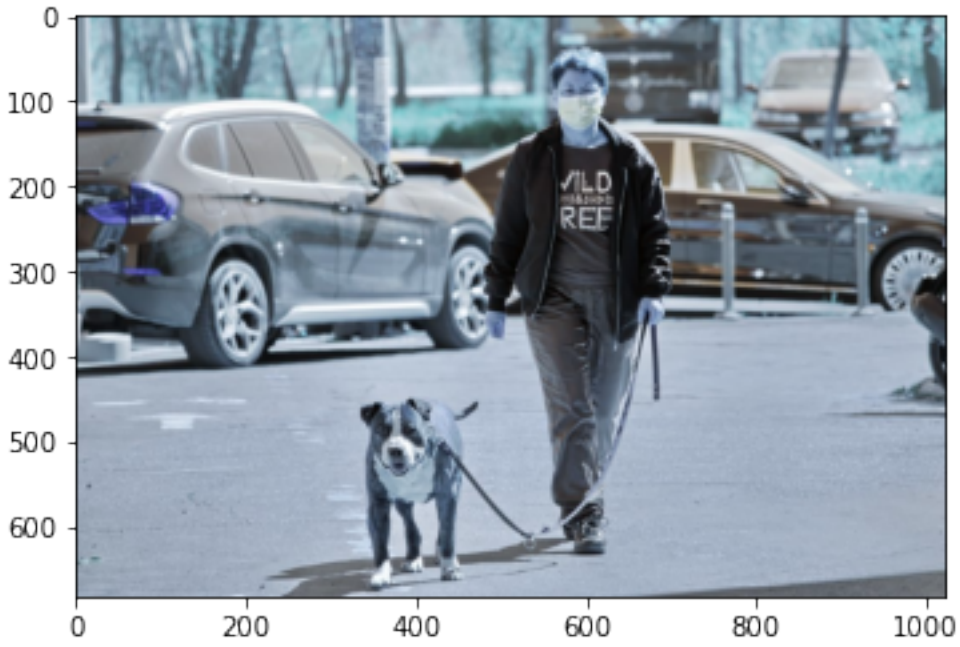
[8]:
```
#in the config file, it is specify that the image should be 320x320 so we have␣
 ↪to resize the image
model.setInputSize(320,320)
model.setInputScale(1.0/127.5)
model.setInputMean((127.5,127.5,127.5))
model.setInputSwapRB(True)
```

[8]: <dnn_Model 000001DF885B2290>

[9]:
```
#read image
img = cv2.imread('dog_ppl.jpg')
```

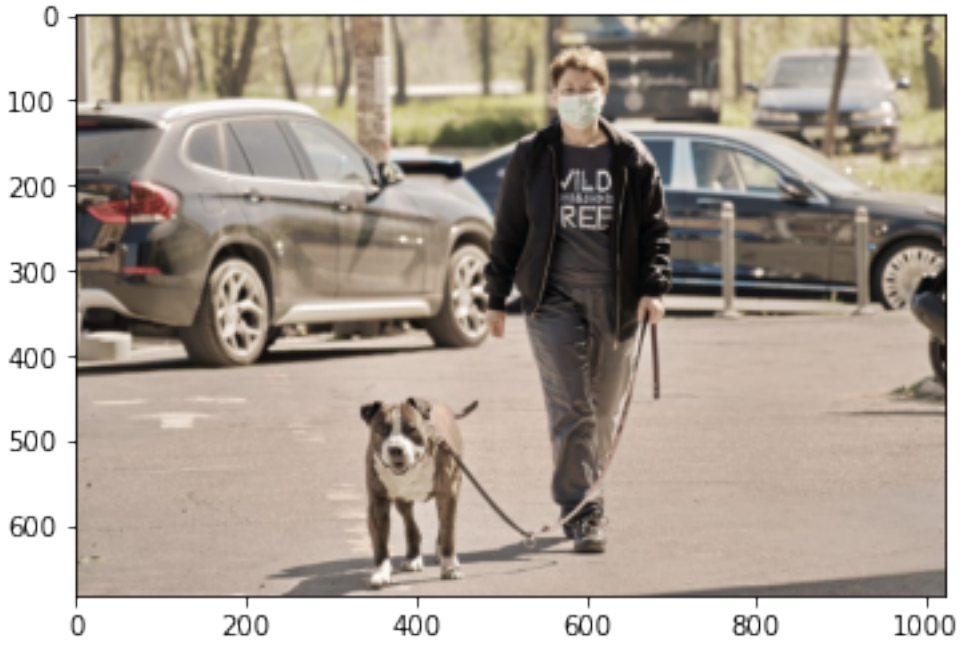
[10]:
```
#this is bgr form
plt.imshow(img)
```

[10]: <matplotlib.image.AxesImage at 0x1df88609370>



[11]:
```
#convert it to real form
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
```

[11]: <matplotlib.image.AxesImage at 0x1df89f4ac40>



[12]:
```
#train the model with 60% accuracy
ClassIndex, confidence, bbox = model.detect(img, confThreshold = 0.6)
```

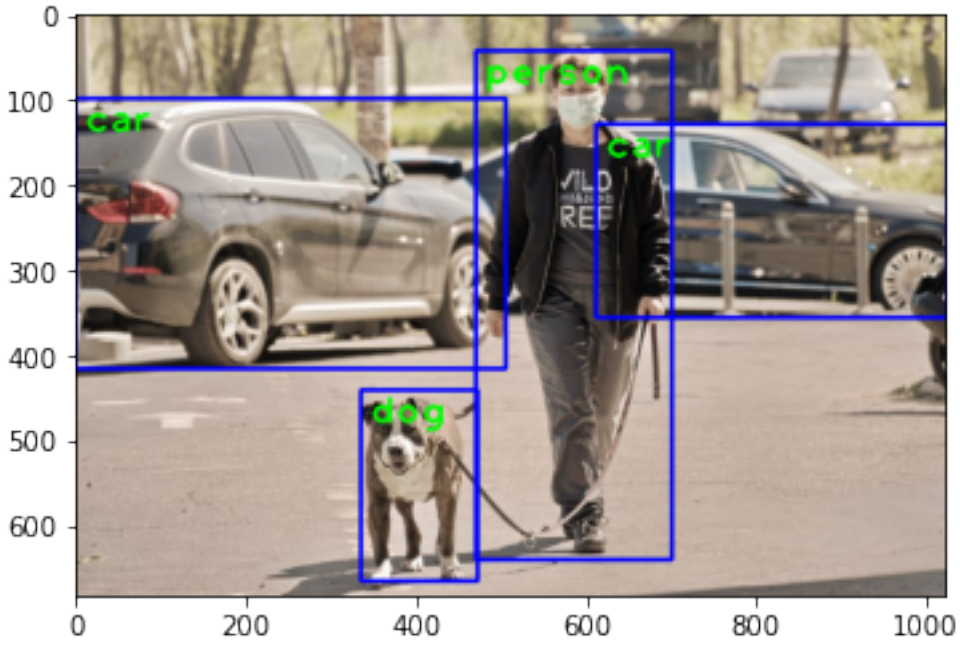
[13]:
```
#we can see that the model detected 4 objects that corresponding to index in␣
 ↪the labels text file
#for example: 1 is person
print(ClassIndex)
```

```
[18  1  3  3]
```

[14]:
```
#draw out boxes and label them according to labels index
font_scale = 3
font = cv2. FONT_HERSHEY_PLAIN
for ClassInd, conf, boxes in zip(ClassIndex.flatten(), confidence.flatten(),␣
 ↪bbox):
    cv2.rectangle(img, boxes, (255,0,0), 3)
    cv2.putText(img, classLabels[ClassInd-1],(boxes[0]+10,boxes[1]+40),font,␣
 ↪fontScale=font_scale,color=(0,255,0), thickness=3)
```

[15]:
```
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
```

[15]:
```
<matplotlib.image.AxesImage at 0x1df89d69730>
```


person
car
car
dog
0
100
200
300
400
500
600
0
200
400
600
800
1000

[ ]: