MaSSP AI PROJECT DAILY REPORT NO. 1

Team 3

Monday, July 8th, 2019

Project theme: OBJECT DETECTION – Finding certain objects from input images or videos

1 General Progress

- Pinpointed a reliable way to develop this project
- Collected a usable training dataset of images online, 12GB of images and annotations from *http://cocodataset.org/*, specfically the 2014 image dataset.
- Produced a working code for the AI core of the program (thanks to the help of mentor Giang), using the pre-trained model YOLOv3, specifically the YOLOv3-tiny weights set. (details in section 4.1)
- Tweaked the code in order to specifically find one desirable category of item in the image, instead of everything findable. (details in section 4.1)

2 Future Plan

- Deciding on what front-end development path will be used as the wrapper for the AI Core (the votes are leaned to creating an Android application)
  - Tools considering to use for Android Application: Kivi (*https://kivy.org*)
  - Will make use of the pre-made code from the Facebook group if choose to create a desktop application instead
- Probably re-training the model with a specific category in mind
- Thinking of analyzing camera in realtime, but the analyzing speed is not quite guaranteed to make an application out of it.

3 Obstacles

- Problem with the possibility of continuing to train the pre-trained model: no idea where to start
    - Already inquired mentors about it, the problem with the solution is that we have to ourseves cherry-picked the layers from the YOLO model to retrain it

4 Demo

The Colab link containing the progress so far: *https://drive.google.com/open?id=186OWo8ZU0dRYfMjsqumBpMVNRWlR2kwh*

4.1 Code Analysis

Summary: The output images contain every "Pizza" item that is found by the model from the input image, each bounded by a green bounding box. The images are produced chronologically as the bounding boxes are plotted.

- Input image:

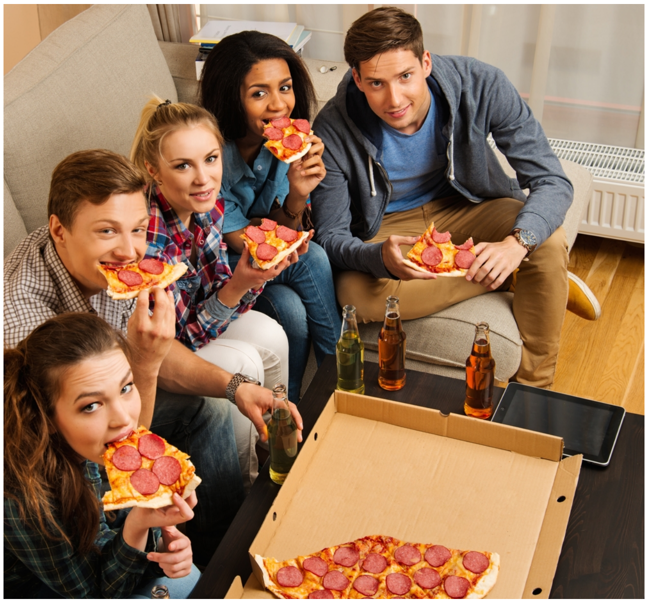


Image 1: Input image. Resolution 1000 x 929. Source: Internet

- Output picture:

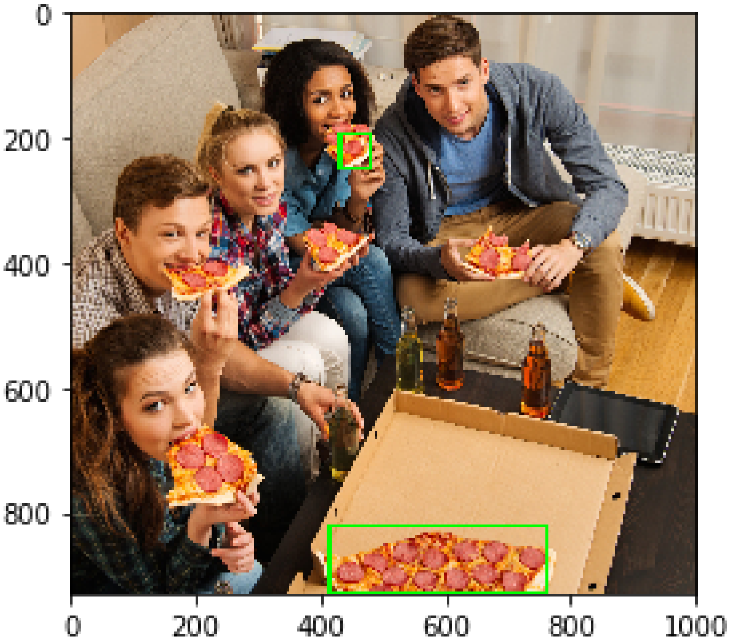
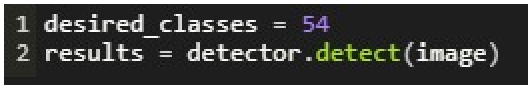


Image 2: Output image. The ”Pizza” items are identified and pinpointed in the bounding boxes.

- Due to the tininess of the yolo-tiny weight, although the analyzing speed is fast, the YOLO-tiny model did not capture every ”Pizza” items in the image.

- Can set what item category to find in the image by changing the `desired_classes` variable.



```
desired_classes = 54
results = detector.detect(image)
```

Image 3: `desired_classes` responsible for finding specific item

- Should the user want to find, say, ”Person” category inside the image, he/she can do it by changing `desired_classes` to `1`. The output image afte the change:

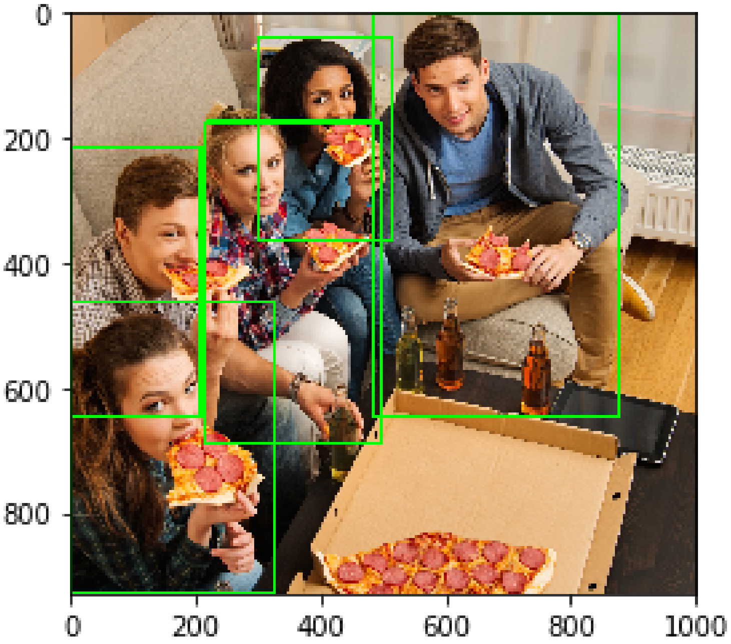


Image 4: Output image of ”Person” items founded in the image

-