ITAI 1378 Computer Vision

Spring 2024

Professor: Patricia McManus

Mayela Miguel

## Questions for Reflection and Analysis:

1. \*\*Conceptual Understanding:\*\*

- What is the main difference between image classification and object detection? How is this difference evident in the output of this exercise?

Image classification focuses only on preparing the data into classes; for example, if a picture of a chihuahua is given, then it should go into the category of dog. On the other hand, object detection is about using bounding boxes to detect the location of an object and place it into a predefined category. The differences between this exercise and the CNN exercise are drastically different. In our convolutional neural network assignment, the final output gave us the data (the chihuahua/muffin pictures) attached with the predicted class and that the model made along with the true class. On this assignment, this model tries to predict the location of the object with bounding boxes and, at the same time, place the objects into a category.

- Explain why we chose the SSD MobileNet V2 model for this task. What are its advantages and limitations, especially in the context of limited computational resources?

For the detector of the images, we chose the Single Shot Detector (SSD) MobileNet V2 model because it is efficient for object detection tasks. Because we had limited computational resources, we used this light weight detection model, which lowers the accuracy of the model. We would need to use more diverse and large samples.

2. \*\*Code Interpretation:\*\*

- Describe the role of the find\_images\_with\_classes function. Why is it useful when working with a large dataset like COCO?

find\_images\_with\_classes is a function to find images containing our target classes. By choosing only the photos that have particular target classes, it makes filtering more effective.

- In the plot\_detections function, how does the threshold value (threshold=0.5) impact the number of objects displayed?

By adding labels and bounding boxes to the image, this method makes the discovered items visible. Only items that have a confidence score higher than or equal to this cutoff are displayed.

- Explain how the heatmap visualization helps you understand the model's confidence in its detections.

The model's predicted bounding boxes and the actual object positions can be directly compared thanks to the grid with ground truth boxes presentation.

3. \*\*Observing Results and Limitations:\*\*

- Run the exercise multiple times. Which types of objects does the model tend to detect more accurately? Which ones are more challenging? Can you explain why?

This model tends to detect people more accurately; in the six pictures that I prompt it to process, it at least labeled the humans with, of course, something unrelated, but it never gave the ‘unkown’ label.

- Observe the bounding boxes. Are there any instances where the boxes are inaccurate or miss the object entirely? What factors in the images might be contributing to these errors?

The bounding boxes are very close to the ground truth in most of the 6 pictures I looked at; where it did have some trouble was when the picture was low quality; it seems that the model had trouble identifying the object.

- How would you expect the accuracy of the model to change if we had used the entire Pascal VOC 2007 dataset instead of a small subset? Why?

The model learns more robust features when the photos are more diversified. Additionally, it lessens overfitting. The chance of memorization of certain images is reduced with a larger dataset. With a less visible class, we could make even more progress.

4. \*\*Critical Thinking:\*\*

- How could you modify the code to detect a specific set of objects, like only animals or only vehicles?

First, you would need to identify the class IDs associated with the objects you want to detect. Then update the detection and visualization code to check if the predicted class IDs are in the defined target list. so that you can detect the target ID only to animals or anything else.

- If you wanted to train your own object detection model, what steps would you need to take? What are some challenges you might encounter?

1. First, I would need to know what it is I want my model to detect.

2. Get a large data set, preferably labeled, that is aligned with my interest, then label the images with bounding boxes around the object, then convert the annotations into what I need to start training the framework, such as the Pascal VOC.

3. Choose what model architecture best suits my needs; preferably, I would choose CNN because I’m already familiar with it.

4. Install the libraries and get enough hardware for the project.

5. augment the dataset to improve model robustness./normalize

6. Start training the model, then validate the model and improve on it.

Some of the challenges I might encounter are overfitting, the quality of my data being poor, multiple annotation errors, or even running out of computational resources.

- Given the limitations of this model, in what real-world scenarios might it still be useful for object detection?

Although our model's accuracy in correctly categorizing the images is lacking, the bouding boxes did not deviate significantly from the ground truth boxes, which may be helpful for identifying things in crowded images, like a puzzle solver.

Works cited

Germanov, Andrey. “How to Detect Objects in Images Using the YOLOv8 Neural Network.”

*FreeCodeCamp.org*, 4 May 2023, www.freecodecamp.org/news/how-to-detect-objects-in-

images-using-yolov8/.