Jacob Ehling

Prof McManus

ITAI 1378

Lab 9 Object Detection

**Introduction**

Today, we went from image classification to object detection, completely changing our type of classification. We had to use an entire swathe of tools, using the SSD MobleNet V2 model, and the Pascal VOC 2007 dataset this time. Also, I was able to explore other pre-trained models as well, furthering my understanding of detection models and workflows.

**Objectives**

Our lab objectives today comprised of

* Setting up our environment with the correct hardware
* Loading and processing some of the Pascal VOC dataset
* Using SSD MobileNetV2 for object detection, and experiment with others

These objectives allow us to successfully build and run our object detection model efficiently, ensuring all of the correct tools are imported and ready to work with, allowing for us to learn and build our experiment.

**Techniques**

For object detection, our approach was different from previous labs. Using visualization tools, and other methods, we were able to not only complete the lab, but understand what the AI “sees” as well. To do this in our lab, we

* Used transfer learning with our pre-trained models
* Used visualized bordering boxes and detection confidence
* Saw new techniques in action such as Intersection over Union, precision, and recall

**Key Concepts**

Object Detection

Object detection is different from image classification since it not only identifies objects, but where they are in the image using bounding boxes. This adds another layer of complexity, which is why we have specific models for this use case.

Transfer Learning

Since we used transfer learning with SSD MobileNetV2, we were able to not only reduce training time, but reduce our computational requirements as well. This means that by also refining the learning method, we can further refine our models and their efficiency overall.

Evaluation Metrics

The IoU measurement helped us see how well the model did at predicting bounding boxes, vs their actual bounding boxes. Meanwhile, our precision and recall metrics showed the importance of reducing false positives and missed objects

**Challenges and Solutions**

Initially, our GPU was having issues when attempting to run our code. To solve this, I attempted restarting our notebook to no avail. Eventually, I switched from GPU to TPU, and our issue was solved. This was my main issue, as it took quite awhile to remedy.

**Observations and Limitations**

Our limitations are easily observable, even in the instructions for the lab. For example, we are using a piece of a whole dataset, as it is too large. This means we are limited computationally, hence why we borrow Google’s, and then must worry about storage or time. While Google does a great job of issue mitigation, limitations will stifle AI due to how much data and power AI needs.

**Ethical Considerations**

The largest ethical consideration that came to mind in this lab was biases in the dataset. This would cause issues with model fairness, and mean that some people or objects would be singled out unfairly, simply due to computational error. While this alone is unethical, so is object detection 24/7/365. These systems must be regulated in terms of installation, or we will have AI consuming data on our entire lives, something that does not sound particularly ethical.

**Real-World Applications**

Object detection is absolutely vital for tasks such as pedestrian detection, suspicious activity prevention, and at-a-glance inventory management. These are all tasks object detection was built for, and thus must perform well at. We can see these day-to-day at the airport, in Tesla’s, or even when asking ChatGPT about a photo.

**Reflection**

1. **Conceptual Understanding:**

* What is the main difference between image classification and object detection?
  + Image classification assigns one label per image, where Object detection identifies multiple objects in one image with the help of bounding boxes.
* How is this difference evident in the output of this exercise?
  + Not only can we see multiple objects identified in one image, but we see the bounding boxes the model uses as well.
* 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?
  + The SSD MobileNetV2 model is extremely efficient, making it ideal for our lab such as this. The benefit is we have good speed and accuracy, as well as the ability to run the model on small devices. The model is also pre-trained on large datasets, improving its results.
  + The downside to this model is the sacrifice of quality. Some larger models, while less efficient, would yield more accurate results.

1. **Code Interpretation:**

* Describe the role of the find\_images\_with\_classes function. Why is it useful when working with a large dataset like COCO?
  + The find\_images\_with\_classes function retrieves images that have specific image classes to them.
  + This is good with large datasets like COCO because it can isolate relevant examples, while filtering out others
* In the plot\_detections function, how does the threshold value (threshold=0.5) impact the number of objects displayed?
  + The threshold value is what determines the confidence level required before an object is displayed.
  + This means by lowering this threshold, we may receive more false positives, whereas if we increase it, precision goes up as well
* Explain how the heatmap visualization helps you understand the model's confidence in its detections.
  + The heatmap shows sweeps of confidence scores across multiple objects in the image, focused on highlighting areas.
  + This means we can see exactly where our model needs improvement, and on what type of detection

1. **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?
  + Smaller objects, or objects that overlapped, are notable harder to detect than their counterparts. This was evident in our trials, as small animals or objects were harder to detect.
* 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 main factors of incorrect bounding boxes appears to be lighting, low contrast between object and background, or similarly textured objects nearby.
* 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?
  + With more data, comes more accurate models. This principle works due to showing the AI more correct info to work with, and watching as it succeeds with this extra data. The more data AI has, the better, as a general rule.

1. **Critical Thinking:**

* How could you modify the code to detect a specific set of objects, like only animals or only vehicles?
  + To do this, you would modify the find\_images\_with\_classes function to find the specific classes you want
* If you wanted to train your own object detection model, what steps would you need to take? What are some challenges you might encounter?
  + Data Collection (with bounding boxes)
  + Normalize data
  + Choose best model for my use case
  + Train model using transfer learning
  + Evaluate model, fine tune where necessary
* Given the limitations of this model, in what real-world scenarios might it still be useful for object detection?
  + The model would still be suitable for many use cases, for example simple camera security, pedestrian detection, road sign detection, wildlife monitoring, or inventory management. These are all relatively simple tasks that object detection models would excel at.