Reflection and Analysis Questions + Journal

1. Conceptual Understanding:

Difference Between Image Classification and Object Detection

- Image Classification: This involves assigning a label to the whole image. For example, if you have a picture of a moose, then the output from the image classification model will be "moose" as the label. It does not tell you where the object is with respect to the location in the image.

Object Detection: It is a task involving image classification of an object and its location through drawing a bounding box around it. Suppose there is an image containing a cat and a dog, then this technique of object detection will label those objects and provide their locations with bounding boxes.

- Distinctive Difference: The output of object detection will include bounding boxes around the detected objects with their class labels, along with the confidence scores, which is not the case for image classification that simply provides a single label for the whole image.

- Why SSD MobileNet V2:

- Advantages:

• Speed: SSD (Single Shot MultiBox Detector) MobileNet V2 is optimized for fast object detection at low computational cost and is hence very suitable for applications in real-time.

• Efficiency: MobileNet V2 is lightweight and efficient, helping to run the model on devices with limited computational resources.

- Versatility: It can detect a wide range of objects and work well under various conditions.

- Limitations

- Accuracy: Effective though it is, it will not be as accurate as larger models that have more important computational powers, especially when placed on small or faraway objects.

- Complexity: For instance, architecture has design trade-offs for efficiency. It might miss highly small or overlapping objects.

2. Code Interpretation:

- Function <find\_images\_with\_classes>:

- Purpose: This function should filter a dataset in search of images containing certain classes of interest. In large datasets like COCO, you might want to process only images that have specific classes in them, which saves processing time and computational resources.

- Usefulness: It narrows down the dataset to relevant pictures and hence makes it more operational and relevant for some tasks or analyses.

- Impact of the Threshold Value (0.5) in <plot\_detections>:

• Threshold Effect: The threshold value sets the minimum required confidence score a detection must return to be regarded as a valid detection. This means that with a threshold value of 0.5, only detections having confidence scores above 50 percent will be shown.

- Number of Objects Displayed: Decreasing the threshold will show more objects, while increasing it will show fewer objects with higher confidence. A balance is needed to avoid false positives and missed detections.

- Heatmap Visualization:

- Visualization Purpose: Plots a heatmap of the confidence of the model respecting its detections. This is a visible cue about which areas are detected by the model and at what level of confidence such detection occurs.

- Understanding the Confidence Score: A high confidence score would tend to bring more prominence and accuracy in the visualization, while a low score might hint toward a less reliable detection.

3. Observing Results and Limitations:

- Model Accuracy Observations:

• Accurate Objects: These models work well on common objects, characterized by distinct features and adequate training data.

• Challenging Objects: The model is less likely to correctly classify small, overlapping, or less distinctive objects.

• Reasons: Often, the object size, occlusion, and image quality become major influencing factors in detection performance.

- Bounding Box Accuracy:

- Inaccuracies: Missed or inaccurate bounding boxes could occur in such scenarios as:

· Object Overlap: There can be confusion among the detectors due to overlapping objects.

· Scale Variability: Confusion around bounding box accuracy may arise when target objects exist in multiple scales.

· Background Clutter: Complex backgrounds result in false positives or missing detections.

- Dataset Size Impact on Accuracy:

-More Data: The model could also be trained on a larger dataset, such as the full Pascal VOC 2007, since it would provide more examples and more data to train from.

- Why: Large datasets will make the model learn more comprehensive features and patterns; thus, there will be better generalization.

4. Critical Thinking:

- Modifying Code for Specific Objects:

Targeting specific objects: modify the pipeline for detection to filter or train the model only for certain object classes, such as animals or vehicles.

Implementation You would adjust the list of class names and set filters in the detection pipeline to only process and visualize detections for the classes you are interested in. Train Your Own Object Detection Model:

Steps:

1. Data Collection: Create a dataset with the objects to be detected, then annotate it.

2. Model Selection: Choose a model architecture that could be such as SSD and Faster R-CNN depending on one's needs.

3. Training: Train the model in your annotated dataset, tuning the hyperparameters and settings of your model where necessary.

4. Evaluation: The model is validated on another independent dataset for the evaluation of performance and making any requisite changes.

• Challenges

Data annotation: Generation of a large data set and its annotation is time-consuming.

Computational Resources: Training in complex models requires huge computation power.

Overfitting: To balance the model in generalization without getting into the overfitting of training data.

Real-World Scenarios of the Model:

Usefulness: While models like SSD MobileNet V2 have some shortcomings, they can still be immensely helpful in scenarios requiring real-time detection with lower computational cost—mobile applications or embedded systems, for example—or where speed is more critical than absolute accuracy. This could be applied in real-time surveillance systems, autonomous vehicles, and mobile apps used to recognize different objects.

