Analysis of Instance Segmentation using MASK RCNN

Team Members:

Likhita Chandana Adabala

Naveen Kumar Reddy Desireddygari

CONTENTS:

Data Collection:

My teammate and I have Collected images of Cars and Sign Boards in the university across one-stop parking and around Bergami Hall and in the Maxcy hall Parking area. Our main aim is to detect these 2 classes using Instance Segmentation. Instance segmentation is a computer vision technique that identifies individual objects within an image and assigns each object a unique label and pixel-level mask or polygon that precisely delineates its boundaries. However, I have encountered a challenge in using this technique for the cars in your dataset, as their shape is better suited to be represented by a bounding box or rectangle rather than a polygon. Consequently, I have decided to use bounding boxes to annotate the car images instead.

To annotate your custom dataset, I have opted to use the MakeSense.ai application, which enables you to generate bounding box annotations efficiently. However, you are considering switching to the LabelMe library in Python, which offers instance segmentation capabilities, to annotate the images more accurately. I have collected all the images in a single folder and in the process of annotating them for use in your instance segmentation model.

Dataset Link:

https://drive.google.com/drive/folders/1dHrAyntXZg8TqdsV6AIWdl0uByS9c9pr?usp=share_link

Data Processing:

In the Processing step, I have used Pillow Library to read images and pandas for reading csv file and while loading images and annotations simultaneously and used same pillow library to draw bounding boxes on the images. We are creating a drawing object to draw the bounding boxes on the image. Now, extracting the coordinates of the bounding box in YOLO format. Converting the YOLO Coordinates to pixel Coordinates and then drawing the bounding box on the image.

Data Loading:

- The code first sets up the data transformations that will be used on the dataset's photos. It develops transformation specifically. Create an object to transform each image into a PyTorch tensor.
- It then produces a CSV file with the picture metadata and a Dataset object that corresponds to the dataset's root directory. Additionally, it specifies the already defined transformations. Create an object that represents the changes that will be made to each image.
- The program generates a list of indices for every image in the dataset.
- Then, depending on the ratios indicated by TRAIN_SPLIT and VAL_SPLIT and the dataset's length, it determines the index ranges for the training, validation, and testing splits.
- The method generates SubsetRandomSampler objects for each split of the dataset using these index ranges. Batches of photos are produced using these samplers during training, validation, and testing.
- In order to load and preprocess photos in batches during training, validation, and testing, the code constructs three DataLoader objects, one for each division of the dataset. It creates batches of photos using the SubsetRandomSampler objects and sets the batch size to BATCH_SIZE.

Conclusion:

For each example picture, we have produced instance segmentation results using Mask R-

CNN. The drawback of our approach is that it creates several masks from a single image, indicating the existence of two or more classes in the same image. In the upcoming release, we intend to strengthen our model by including a more capable instance segmentation method that can handle many masks per picture. We intend to optimize the model to increase its effectiveness because the high picture size of 2500 pixels causes our existing model's processing time to be slow.

References:

Datasets & Data Loaders — PyTorch Tutorials 2.0.0+cu117 documentation

GitHub Link : arp95/mask_rcnn_instance_segmentation: Instance Segmentation using Mask R-CNN on a Custom Dataset (github.com)

Mask RCNN - https://bjornkhansen95.medium.com/mask-r-cnn-for-segmentation-using-pytorch-8bbfa8511883

NumPy Documentation - https://numpy.org/doc/stable/reference/generated/numpy.floor.html

Pytorch Discussion: https://discuss.pytorch.org/t/exact-coordinates-of-bounding-box-for-mask-r-cnn/89729