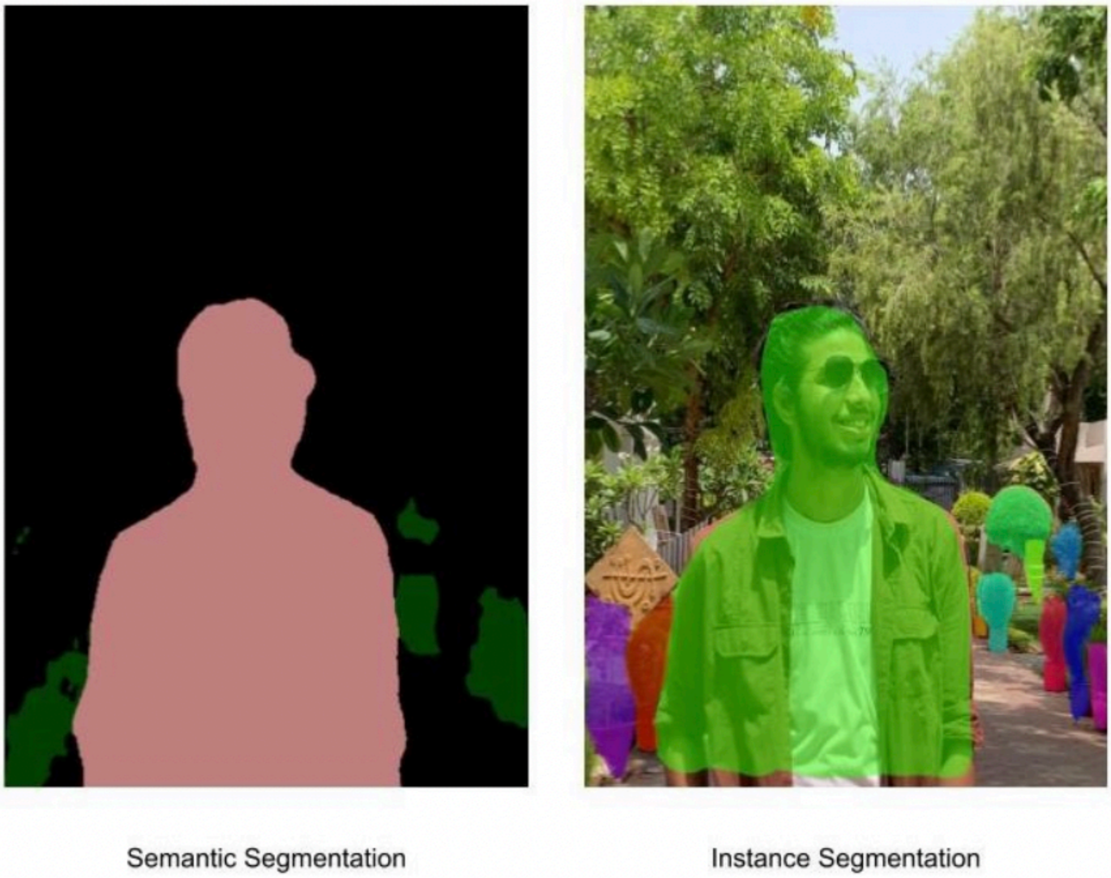
NOTES: DATATHON COMPETITION

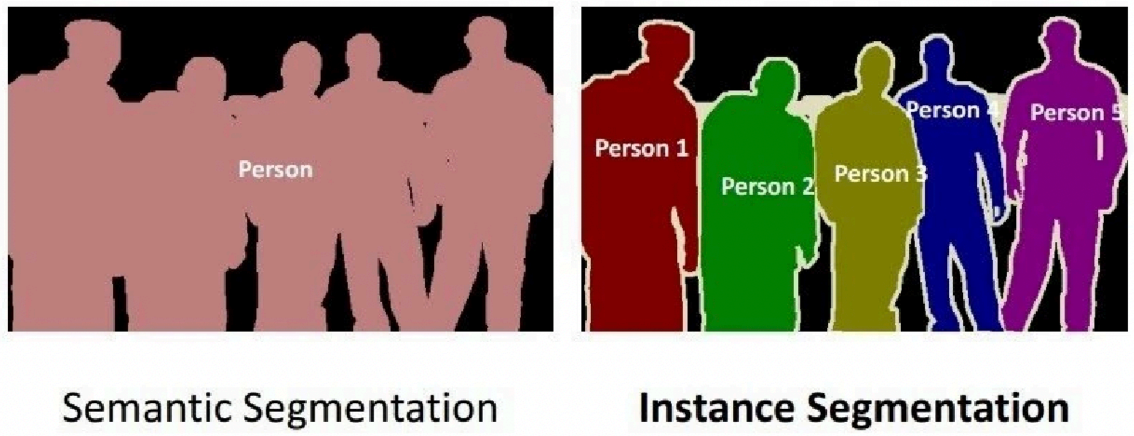
From Instruction Sheet

- 5 Labels: (1)Broken bricks (2)cementitious debris (3)plastic wires (4)PVC pipes (5)Rebars
- TEMAS HAVE TO LABEL THE INITIAL DATASET
- TEAMS ARE EXPECTED TO DETECT THE SILHOUETTES OF OBJECTS IN IMAGES (Semantic Segmentation?)

From Self

- Semantic segmentation: just to differentiate between the different types of objects present in a particular image by giving different colours to different objects
- Instance Segmentation: Is to not just give different colours to different objects but to also have the exact background mentioned in the same image
- Difference between Semantic and Instance segmentation: Semantic Segmentation is more about just differentiating the object from the background, while Instance Segmentation includes separately identifying each time an object makes an appearance



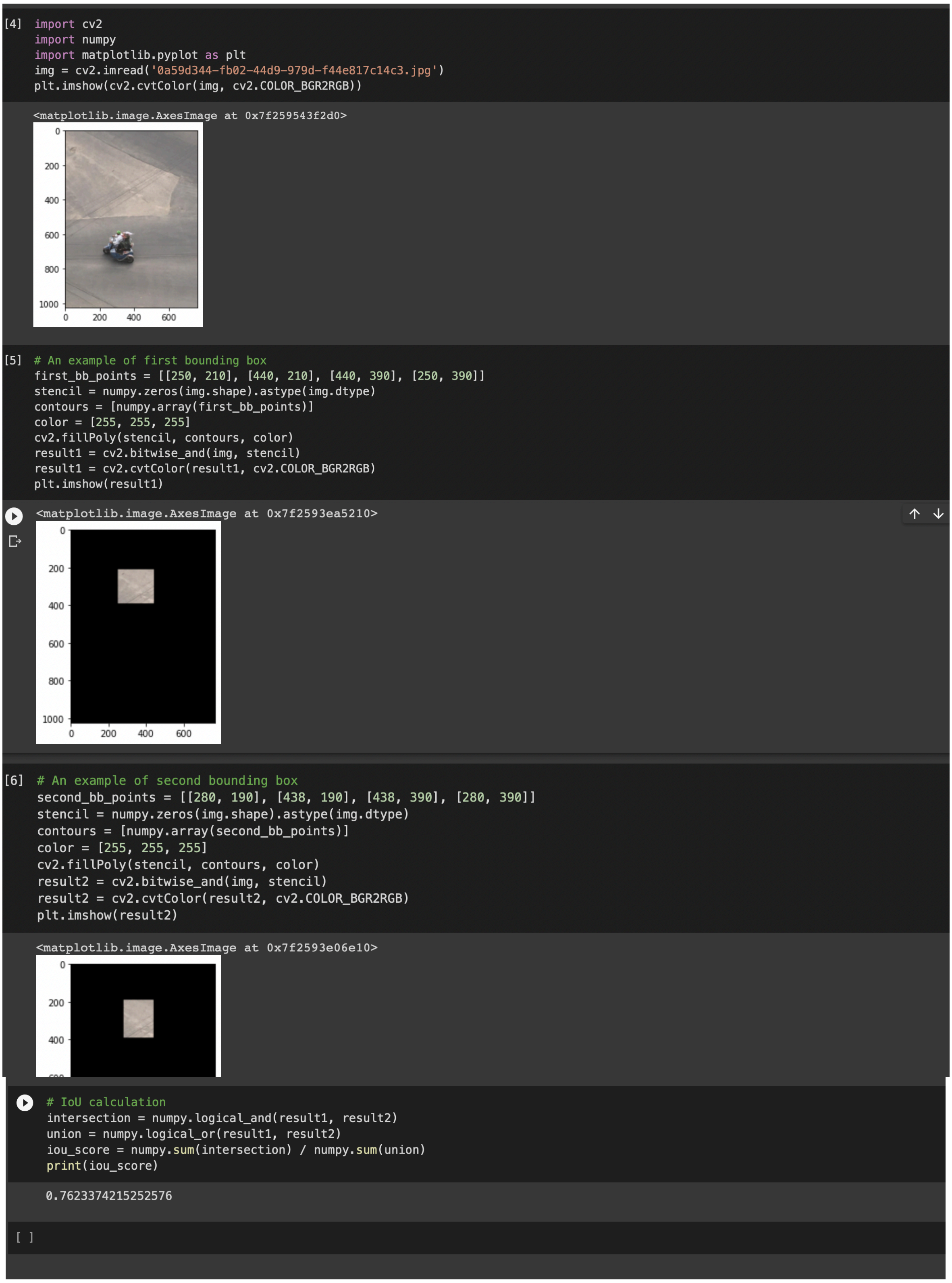


Differences of Semantic Segmentation versus Instance Segmentation

- So basically, Image Segmentation entails (Semantic Segmentation) and (Instance Segmentation)

- CNN can be made to be region based i.e. there is no emergent need to run the whole convolutional operation on the whole image which is why we can first identify a region and then run the more “computationally expensive” operations like Convolution: this is the idea behind R-CNN’s.
- Mask R-CNN (Masked Region Based ConvolutionL Neural Networks):
- For Object Detection (mainly putting up boxes around the objects); use Faster-RCNN’s. For Semantic Segmentation: Fully Convolutional Network (FCN)
- Mask R-CNN uses “ROI Align” which kinda leads to almost no loss of data

- IoU (Intersection over Union) is a metric to evaluate the model in object detection
- Another way to represent IoU: (true_positives)/[(true_positives) + (false_positives) + (false_negatives)]
- Code for IoU as calculated for different boxes in a particular image



```
[4]  import cv2
     import numpy
     import matplotlib.pyplot as plt
     img = cv2.imread('0a59d344-fb02-44d9-979d-f44e817c14c3.jpg')
     plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
```

```
<matplotlib.image.AxesImage at 0x7f259543f2d0>
```

```
[5]  # An example of first bounding box
     first_bb_points = [[250, 210], [440, 210], [440, 390], [250, 390]]
     stencil = numpy.zeros(img.shape).astype(img.dtype)
     contours = [numpy.array(first_bb_points)]
     color = [255, 255, 255]
     cv2.fillPoly(stencil, contours, color)
     result1 = cv2.bitwise_and(img, stencil)
     result1 = cv2.cvtColor(result1, cv2.COLOR_BGR2RGB)
     plt.imshow(result1)
```

```
<matplotlib.image.AxesImage at 0x7f2593ea5210>
```

```
[6]  # An example of second bounding box
     second_bb_points = [[280, 190], [438, 190], [438, 390], [280, 390]]
     stencil = numpy.zeros(img.shape).astype(img.dtype)
     contours = [numpy.array(second_bb_points)]
     color = [255, 255, 255]
     cv2.fillPoly(stencil, contours, color)
     result2 = cv2.bitwise_and(img, stencil)
     result2 = cv2.cvtColor(result2, cv2.COLOR_BGR2RGB)
     plt.imshow(result2)
```

```
<matplotlib.image.AxesImage at 0x7f2593e06e10>
```

```
# IoU calculation
intersection = numpy.logical_and(result1, result2)
union = numpy.logical_or(result1, result2)
iou_score = numpy.sum(intersection) / numpy.sum(union)
print(iou_score)
```

```
0.7623374215252576
```

```
[ ]
```

- Approaches in Image Segmentation: (1) Similarity Approach (based on detecting similarity between adjacent pixels and segregating into different classes if the similarity coefficient exceeds a particular threshold) (2) Discontinuity Approach (works on the sudden change in pixel values as observed in detection of lines, points, edges)

- Image Segmentation Techniques: (1) Threshold Based Segmentation (2) Edge Based Segmentation (3) Region Based Segmentation (4) Clustering Based Segmentation (5) ANN Based Segmentation

- Threshold segmentation basically is a bi-modal setup which is preferred to be applied only in cases wherein we are able to observe two peaks in the pixel intensity histogram of that particular image. The threshold is chosen between the peaks so that the values less than the threshold can be given a value of 0 and the others a value of 255. This way, we can ideally separate an object from the background. In cases wherein bi-modal division is not possible, a straightforward thresholding process is pretty much useless which is why there are various mathematics based methods to figure out the threshold by using tolerances as desired by the user.

- Edge Based Segmentation basically uses kernels already discovered by people to figure out the magnitudes and directions of the edges and then use them to differentiate between the two 'distinct' sides of the edges.

- Region Based Segmentation is pretty much a sophistication of the edge based segmentation since it is based on finding similarity within nearby pixels. As a point to be noted, the similarity parameter here need not be dependent on just the edges but is mainly based on other parameters like intensity, colour, etc. There are further two types of Region Based Segmentations (1. Region Growing Technique: bottom-up 2. Region Splitting and Merging Method: kinda top-down)

- Clustering Based Segmentation works basically on the K-Means Clustering Algorithm which is an unsupervised algorithm. Basically, centroids of clusters are initially randomised and then all points are allotted to a particular cluster whose centroid is closest to that particular point. With the clusters now newly formed, the centroids are recalculated and the algorithm saturates fast enough and does not usually bounce.

From Lu et.al. 2022

- Aim: using computer vision and semantic segmentation technique to learn more about the composition of the construction waste
- Two parameters to quantify model performance: mean IoU over all (9) classes AND time of performance per image (formally A.K.A. Segmentation Accuracy & Time Consumption)
- Data collection, variation in data (especially CW related), parameters like: hue, saturation, brightness (HSB)
- They've recognised 5 main hyper-parameters: effects of (backbone, pre-trained datasets, input pre-processing at inference stage, output stride, image resolution)
- Confusion matrix to represent and quantify model performance

From Na et.al. 2022

- Frechet Inception Distance (FID): used to increase the number of images through augmentation. It is a generative model, like a GANS, to result in more number of training points and hence possibly more robust outcomes.
- Application of YOLACT and YOLO algorithms
- YOLACT Demonstration: https://www.youtube.com/watch?v=P2-C8ya3bdw
- YOLACT Google Colab Comprehensive Tutorial: https://www.youtube.com/watch?v=x9PZ1xjIkng (Demonstration of a YOLACT working real-time)

From Dong et.al. 2022

- Attention, Transformer

CONCLUSIONS:

1. Got a quite comprehensive and much better understanding of the problem at hand
2. Image Processing; entails a lot of other things than just CNN; will have to take a look at it in greater detail
3. Multiple methodologies have been exemplified already for categorising CW; need to analyse them and see what fits our needs the best