

# Report: Lab1-Dectectron2

Lloyd A. Thompson  
Computer Vision-2 (28-April-2020)  
African Masters for Machine Intelligence  
African Institute for Mathematical Sciences, Rwanda

## A. Introduction:

This report covers my exploration of Detectron2, which is an object recognition library written in PyTorch. Detectron2 contains state-of-the-art pre-trained models for implementing Machine Learning/Computer Vision projects and concepts.

I used the official colab notebook provided for this exercise, colab provides GPU capabilities for computation. Installation is pretty simple, it took a fraction of a second to install Detectron2 and all other dependencies.

## B. Pretrained/Instance Segmentation

Initially, I loaded a pretrained model and a sample image from the COCO dataset to see how it works. After visualization, the following objects were detected with their predictions and bounding boxes respectively. 9people with 100, 99, 98, 97 and 67percent estimations. A horse was also segmented with 100percent, then 3-umbrellas out of 5 were detected with 80, 89 and 91percent predictions. It is impressive how the model was able to estimate 99percent for a person with only half hair showing in the image.

## C. Pretrained Pose Estimations

I repeated "part B" again, but this time around I tried a set of images from my gallery to segment the objects. During visualization, I got 96 and 98percent for persons, 99percent for book, 99percent for chair/couch. The model seems to have high predictions for objects found in the dataset irrespective of their orientation or location. The model is still able to predict high estimations for objects with only small parts visible. However, it struggles with crowded objects. It did miss a few

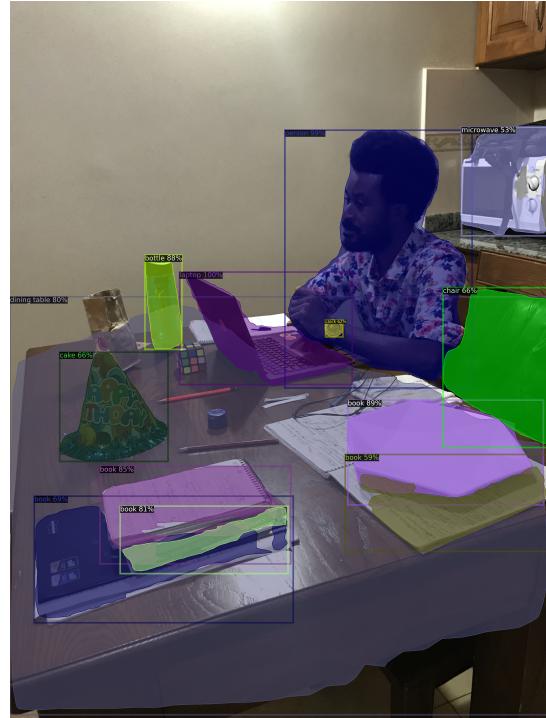


Fig. 1. Instance Segmentation

heads or objects, and also some human pose positions were not accurate. The model architecture used is Mask-RCNN, with Resnet-50-FPN as the backbone

## D. Error Modes

The model misclassified a birthday hat as cake. It also failed to detect a pencil and a lip-balm on the same table, obviously, those classes are not in the dataset. With grouped images, the model consistently missed a lot of objects especially those in tight positions. I had high estimations (over 90percent) for objects that have related classes in the dataset. If the labels are not found in the



Fig. 2. Key Points

dataset, the model either fails to segment or gives a low value estimations.

#### E. Conclusion

In conclusion, this has been a useful lab exercise for understanding the capabilities of Detectron2. I learnt how to install Detectron2 and its dependencies, how to detect objects and their instance segmentation. I was able to see how bounding box works by running pretrained models. Finally, I look forward to exploring custom dataset, especially for balloon segmentation, which does not have a label in the COCO dataset then subsequently run panoptic segmentation on video.

#### REFERENCES

- [1] <http://cocodataset.org/home>
- [2] <https://colab.research.google.com/drive/16jcaJoc6bCFAQ96jD>
- [3] <https://github.com/facebookresearch/detectron2/blob/master/>