1. **Catastrophic forgetting of the backbone:** When we tried to train Masked RCNN by unfreezing all of the layers in the network, we observed that the train and validation loss started to decrease very fast in the first few epochs. Validass loss plateaued, but the training loss kept decreasing signaling that the network overfitted on the training data. Experiments with different learning rates and weight decays did not resolve this issue. Our hypothesis is that this is caused because backbone that is responsible for extracting rich features from the image overfits quickly on the training data and ‘forgets’ the general feature representation capabilities like detecting edges and corners that comes from pretraining on the huge ImageNet dataset. We resolved this issue by performing the training stage by stage. The details of the training scheme can be found in the word document general\_training\_and\_val\_methodology’.
2. **Huge size of data:** The cityscapes dataset is huge, each RGB image is 2mb on average (with annotation data for each image also taking around 200kb). This means more than 10GB of data must be loaded and manipulated for full scale training of MaskedRCNN. This introduced 3 main challenges: long loading times, fitting the data to GPU and CPU RAM, and increased time and compute power required for the training procedure.

To counter this challenges the following measures were taken:

* Using mixed precision training, mixed precision training can greatly reduce training time and GPU memory usage allowing larger batch sizes by performing inference operations in lower FP.
* Partitioning the training and validation data to 8 and 4 partitions respectively and dynamically loading and deleting them from GPU and CPU each epoch to free memory.
* Renting Google Colab L4 as our computer’s GPU was not enough to handle training and inference.