

I am using 1 late day for this assignment.

301439536

Part 1

List of the configs and modifications that you used.

Original: MAX_ITER = 500, BATCH_SIZE_PER_IMAGE = 512, IMS_PER_BATCH = 2, BASE_LR = 0.00025

My altered version: MAX_ITER = 500, BATCH_SIZE_PER_IMAGE = 512, IMS_PER_BATCH = 2, BASE_LR = 0.001

Performance enhancing factors

I used "COCO-Detection/faster_rcnn_X_101_32x8d_FPN_3x.yaml" instead of "COCO-Detection/faster_rcnn_R_50_FPN_3x.yaml" because it converged better.

Ablation Study and accuracy plot

BASE_LR: When I increase its size it would have less stable improvement over time; likely due to skipping over the local minimum. As you can see in the BASE_LR=0.001 example below, the total loss is less predictable. In contrast, BASE_LR=0.00025 was small enough to continually improve without improving too slowly, nor settling at a relatively high local minimum. Despite this, BASE_LR=0.001 had slightly better results, so I used that in my project.

Unfortunately, I could not get 'tensorboard' to function properly, so here is the output of my training for trainer.train() in which BASE_LR is equal to 0.001 and 0.00025 respectively.

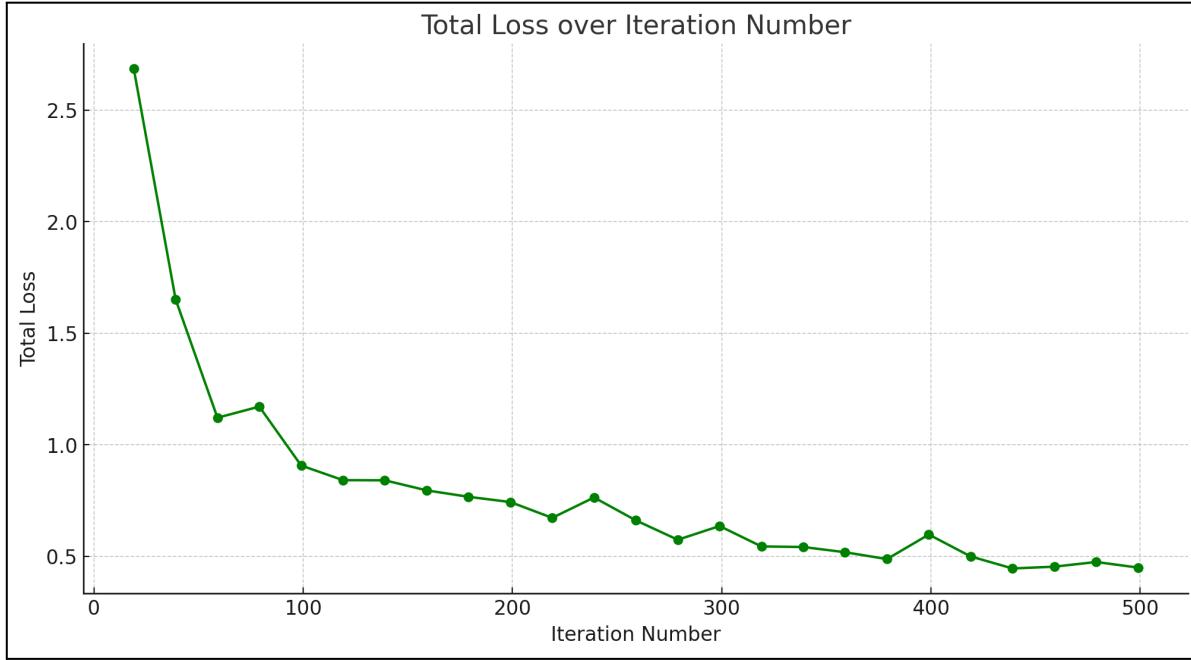
```
[11/18 02:01:32 d2.data.build]: Removed 0 images with no usable annotations. 100 images left.
[11/18 02:01:32 d2.data.dataset._Mapper]: Augmentations used in Training: [ResizeShortEdge(short_edge_length=(640, 672, 704, 736, 768, 800), max_size=1333, sample_style='choice'), RandomFlip()]
[11/18 02:01:32 d2.data.dataset._Mapper]: Using training sampler TrainingSampler
[11/18 02:01:32 d2.data.dataset._Mapper]: Serializing 100 elements to byte tensors and concatenating them all ...
[11/18 02:01:32 d2.data.dataset._Mapper]: Serializing dataset takes 9.02 MB
[11/18 02:01:32 d2.data.dataset._Mapper]: Total training data size: 100.00000000000001 MB
[11/18 02:01:32 d2.data.dataset._Mapper]: Total training data size: 100.00000000000001 MB
[11/18 02:01:32 d2.checkpoint._checkpoint]: (DetectionCheckpointer) Loading from https://dl.fbaipublicfiles.com/detectron2/COCO-Detection/faster_rcnn_R_50_FPN_3x/c137845459/model_final_280758.pkl ...
WARNING:rcvcore.common.checkpoint:Skip loading parameter 'roi_heads.box_predictor.cls_score.weight' to the model due to incompatible shapes: (81, 1024) in the checkpoint but (2, 1024) in the model! You might want to double check if this is expected.
WARNING:rcvcore.common.checkpoint:Skip loading parameter 'roi_heads.box_predictor.bbox_pred.weight' to the model due to incompatible shapes: (320, 1024) in the checkpoint but (4, 1024) in the model! You might want to double check if this is expected.
WARNING:rcvcore.common.checkpoint:Skip loading parameter 'roi_heads.box_predictor.bbox_pred.bias' to the model due to incompatible shapes: (320,) in the checkpoint but (4,) in the model! You might want to double check if this is expected.
WARNING:rcvcore.common.checkpoint:None of the parameters or buffers were found in the checkpoint.
ext heads.box_predictor.bbox_pred.(bias, weight)
roi_heads.box_predictor.cls_score.(bias, weight)
[11/18 02:01:48 d2.engine.events]: Iter: 114.48   total_loss: 1.998   loss_cls: 0.6896   loss_box_reg: 0.2753   loss_rpn_cls: 0.5784   loss_rpn_loc: 0.2762   time: 1.8979   last_time: 0.5838   data_time: 1.3931   last_data_time: 0.0816   lr: 3.8942e-05   max_norm: 4961M
[11/18 02:02:37 d2.engine.events]: Iter: 39   total_loss: 1.187   loss_cls: 0.4135   loss_box_reg: 0.258   loss_rpn_cls: 0.2634   loss_rpn_loc: 0.1657   time: 1.5388   last_time: 2.1844   data_time: 0.7038   last_data_time: 1.6245   lr: 7.8922e-05   max_norm: 4961M
[11/18 02:03:05 d2.engine.events]: Iter: 59   total_loss: 0.9895   loss_cls: 0.2081   loss_box_reg: 0.1348   loss_rpn_cls: 0.2394   loss_rpn_loc: 0.1632   time: 1.4697   last_time: 1.8045   data_time: 0.9037   last_data_time: 0.4098B   lr: 8.0001888   max_norm: 4961M
[11/18 02:03:33 d2.engine.events]: Iter: 79   total_loss: 0.8955   loss_cls: 0.1767   loss_box_reg: 0.1252   loss_rpn_cls: 0.2252   loss_rpn_loc: 0.1532   time: 1.4238   last_time: 1.7482   data_time: 0.9037   last_data_time: 0.4098B   lr: 8.0001888   max_norm: 4961M
[11/18 02:04:03 d2.engine.events]: Iter: 99   total_loss: 0.985   loss_cls: 0.1511   loss_box_reg: 0.2373   loss_rpn_cls: 0.2054   loss_rpn_loc: 0.1481   time: 1.4737   last_time: 0.5067   data_time: 0.7372   last_data_time: 0.0837   lr: 8.0001888   max_norm: 4961M
[11/18 02:04:39 d2.engine.events]: Iter: 119   total_loss: 0.8228   loss_cls: 0.1383   loss_box_reg: 0.1993   loss_rpn_cls: 0.1859   loss_rpn_loc: 0.1837   time: 1.5282   last_time: 2.3338   data_time: 1.2733   last_data_time: 1.7343   lr: 8.00023876   max_norm: 4961M
[11/18 02:04:57 d2.engine.events]: Iter: 139   total_loss: 0.8122   loss_cls: 0.1324   loss_box_reg: 0.1961   loss_rpn_cls: 0.1834   loss_rpn_loc: 0.1824   time: 1.5452   last_time: 1.8843   data_time: 1.2733   last_data_time: 1.7343   lr: 8.00023876   max_norm: 4961M
[11/18 02:05:17 d2.engine.events]: Iter: 159   total_loss: 0.8721   loss_cls: 0.1509   loss_box_reg: 0.2383   loss_rpn_cls: 0.1192   loss_rpn_loc: 0.2233   time: 1.5123   last_time: 1.2332   data_time: 0.9888   last_data_time: 0.7764   lr: 8.00031868   max_norm: 4961M
[11/18 02:06:03 d2.engine.events]: Iter: 179   total_loss: 0.8756   loss_cls: 0.1914   loss_box_reg: 0.2498   loss_rpn_cls: 0.09487   loss_rpn_loc: 0.1272   time: 1.4871   last_time: 0.5421   data_time: 0.7931   last_data_time: 0.6114   lr: 8.00035864   max_norm: 4961M
[11/18 02:06:31 d2.engine.events]: Iter: 199   total_loss: 0.8108   loss_cls: 0.1391   loss_box_reg: 0.1985   loss_rpn_cls: 0.1866   loss_rpn_loc: 0.1851   time: 1.5088   last_time: 0.5562   data_time: 1.1863   last_data_time: 0.6189   lr: 8.00035864   max_norm: 4961M
[11/18 02:07:07 d2.engine.events]: Iter: 219   total_loss: 0.6899   loss_cls: 0.1601   loss_box_reg: 0.2144   loss_rpn_cls: 0.09398   loss_rpn_loc: 0.1313   time: 1.5688   last_time: 0.5662   data_time: 1.4863   last_data_time: 0.6189   lr: 8.00043856   max_norm: 4961M
[11/18 02:07:37 d2.engine.events]: Iter: 239   total_loss: 0.58   loss_cls: 0.1331   loss_box_reg: 0.2656   loss_rpn_cls: 0.05977   loss_rpn_loc: 0.07455   time: 1.5081   last_time: 2.1732   data_time: 0.9831   last_data_time: 1.7154   lr: 8.00047852   max_norm: 4961M
[11/18 02:08:05 d2.engine.events]: Iter: 259   total_loss: 0.5104   loss_cls: 0.1273   loss_box_reg: 0.2489   loss_rpn_cls: 0.06066   loss_rpn_loc: 0.07466   time: 1.5081   last_time: 0.5116   data_time: 0.9831   last_data_time: 1.7154   lr: 8.00047852   max_norm: 4961M
[11/18 02:08:33 d2.engine.events]: Iter: 279   total_loss: 0.4109   loss_cls: 0.0644   loss_box_reg: 0.2398   loss_rpn_cls: 0.07399   loss_rpn_loc: 0.1211   time: 1.5038   last_time: 0.5116   data_time: 0.9831   last_data_time: 1.7154   lr: 8.00047852   max_norm: 4961M
[11/18 02:09:08 d2.engine.events]: Iter: 299   total_loss: 0.3228   loss_cls: 0.0826   loss_box_reg: 0.2277   loss_rpn_cls: 0.0655   loss_rpn_loc: 0.137   time: 1.5072   last_time: 0.5662   data_time: 1.0844   last_data_time: 0.0844   lr: 8.00055984   max_norm: 4961M
[11/18 02:09:35 d2.engine.events]: Iter: 319   total_loss: 0.624   loss_cls: 0.1156   loss_box_reg: 0.2097   loss_rpn_cls: 0.1018   loss_rpn_loc: 0.1347   time: 1.4974   last_time: 0.8237   data_time: 0.8221   last_data_time: 0.1792   lr: 8.00063836   max_norm: 4961M
[11/18 02:10:12 d2.engine.events]: Iter: 339   total_loss: 0.4245   loss_cls: 0.0815   loss_box_reg: 0.2202   loss_rpn_cls: 0.0755   loss_rpn_loc: 0.0958   time: 1.4955   last_time: 1.2111   data_time: 0.6586   last_data_time: 0.4098B   lr: 8.00071828   max_norm: 4961M
[11/18 02:10:34 d2.engine.events]: Iter: 359   total_loss: 0.4095   loss_cls: 0.1864   loss_box_reg: 0.1735   loss_rpn_cls: 0.08117   loss_rpn_loc: 0.0954   time: 1.4956   last_time: 1.3131   data_time: 0.6586   last_data_time: 0.4098B   lr: 8.00075224   max_norm: 4961M
[11/18 02:11:15 d2.engine.events]: Iter: 379   total_loss: 0.4681   loss_cls: 0.09398   loss_box_reg: 0.1862   loss_rpn_cls: 0.07595   loss_rpn_loc: 0.1276   time: 1.4974   last_time: 0.5887   data_time: 1.0496   last_data_time: 0.0812   lr: 8.00075224   max_norm: 4961M
[11/18 02:11:43 d2.engine.events]: Iter: 399   total_loss: 0.3712   loss_cls: 0.0715   loss_box_reg: 0.1839   loss_rpn_cls: 0.07595   loss_rpn_loc: 0.1276   time: 1.4955   last_time: 0.5887   data_time: 1.0496   last_data_time: 0.0812   lr: 8.00075224   max_norm: 4961M
[11/18 02:12:12 d2.engine.events]: Iter: 419   total_loss: 0.4392   loss_cls: 0.1103   loss_box_reg: 0.1996   loss_rpn_cls: 0.08447   loss_rpn_loc: 0.1326   time: 1.5156   last_time: 0.5176   data_time: 1.1721   last_data_time: 0.0673   lr: 8.00083816   max_norm: 4961M
[11/18 02:12:37 d2.engine.events]: Iter: 439   total_loss: 0.399   loss_cls: 0.07945   loss_box_reg: 0.1559   loss_rpn_cls: 0.0341   loss_rpn_loc: 0.06212   time: 1.5029   last_time: 1.4491   data_time: 0.7253   last_data_time: 0.5565   lr: 8.0008782   max_norm: 4961M
[11/18 02:13:05 d2.engine.events]: Iter: 459   total_loss: 0.3411   loss_cls: 0.08916   loss_box_reg: 0.1898   loss_rpn_cls: 0.05851   loss_rpn_loc: 0.1289   time: 1.5045   last_time: 0.5165   data_time: 1.0497   last_data_time: 0.08709   lr: 8.00091008   max_norm: 4961M
[11/18 02:13:42 d2.engine.events]: Iter: 479   total_loss: 0.2666   loss_cls: 0.1141   loss_box_reg: 0.1808   loss_rpn_cls: 0.08146   loss_rpn_loc: 0.1348   time: 1.5148   last_time: 0.5157   data_time: 1.2438   last_data_time: 0.08109   lr: 8.00095844   max_norm: 4961M
[11/18 02:14:12 d2.engine.events]: Iter: 499   total_loss: 0.5779   loss_cls: 0.1874   loss_box_reg: 0.1845   loss_rpn_cls: 0.0329   loss_rpn_loc: 0.1269   time: 1.5092   last_time: 1.3085   data_time: 0.8962   last_data_time: 0.8109   lr: 8.000998   max_norm: 4961M
[11/18 02:14:42 d2.engine.events]: Overall training speed: 499 iterations in 0:12:31 (1.5093 s / it)
```

```

}
[11/19 01:36:56 d2.data.build]: Removed 8 images with no usable annotations. 100 images left.
[11/19 01:36:56 d2.data.dataset_mapper]: [DatasetMapper] Augmentations used in training: [ResizeShortestEdge(short_edge_length=(640, 672, 784, 736, 768, 800), max_size=1333, sample_style='choice'), RandomFlip()]
[11/19 01:36:56 d2.data.dataset_mapper]: [DatasetMapper] Sampler used in training: [detection_rpn.data.common.TorchSerializedList]
[11/19 01:36:56 d2.data.common]: Serializing 100 elements to byte buffers and concatenating them all ...
[11/19 01:36:56 d2.data.common]: Serialized dataset takes 9.92 MB
[11/19 01:36:56 d2.data.common]: Writing batched data loader with batch_size=2
WARNING:fcvcore.common.checkpoint: Writing batched data loader with batch_size=2
WARNING:fcvcore.common.checkpoint: Using solver: 'SGD' due to incompatible shapes: (81, 1) in the model but (2, 81) in the checkpoint but (2, 1) in the model! You might want to double check if this is expected.
WARNING:fcvcore.common.checkpoint: Using solver: 'SGD' due to incompatible shapes: (162, 1) in the model but (2, 162) in the checkpoint but (2, 1) in the model! You might want to double check if this is expected.
WARNING:fcvcore.common.checkpoint: Using solver: 'SGD' due to incompatible shapes: (324, 1) in the checkpoint but (4, 162) in the model! You might want to double check if this is expected.
WARNING:fcvcore.common.checkpoint: Using solver: 'SGD' due to incompatible shapes: (326, 1) in the checkpoint but (4, 1) in the model! You might want to double check if this is expected.
WARNING:fcvcore.common.checkpoint: Using solver: 'SGD' due to incompatible shapes: (328, 1) in the checkpoint but (4, 1) in the model! You might want to double check if this is expected.
[11/19 01:36:56 d2_heads.box_predictor.bbox_pred.bias, weight]
[11/19 01:36:56 d2_heads.box_predictor.cls_score.bias, weight]
[11/19 01:36:56 d2.engine.train_loop]: Starting training from iteration 0
[11/19 01:36:56 d2.engine.train_loop]: iter: 15 total_loss: 1.591 loss_cls: 0.1518 loss_box_reg: 0.6899 time: 1.6411 last_time: 1.1811 data_time: 1.1611 last_data_time: 0.5547 lr: 9.7485e-06 max_mem: 4961M
[11/19 01:37:59 d2.util.events]: eta: 01:11:31 iter: 39 total_loss: 1.681 loss_cls: 0.2626 loss_box_reg: 0.2663 loss_rpn_cls: 0.4595 loss_rpn_loc: 0.2372 time: 1.5478 last_time: 1.2073 data_time: 0.9226 last_data_time: 0.7779 lr: 1.0731e-05 max_mem: 4961M
[11/19 01:38:28 d2.util.events]: eta: 00:08:03 iter: 59 total_loss: 1.403 loss_cls: 0.4962 loss_box_reg: 0.2881 loss_rpn_cls: 0.3239 loss_rpn_loc: 0.1483 time: 1.5786 last_time: 1.0451 data_time: 0.9532 last_data_time: 3.4813 lr: 2.0726e-05 max_mem: 4961M
[11/19 01:39:26 d2.util.events]: eta: 00:07:48 iter: 79 total_loss: 1.164 loss_cls: 0.2594 loss_box_reg: 0.1864 loss_rpn_cls: 0.2359 loss_rpn_loc: 0.1039 time: 1.4630 last_time: 1.0711 data_time: 0.9532 last_data_time: 0.4013 lr: 3.0625e-05 max_mem: 4961M
[11/19 01:39:26 d2.util.events]: eta: 00:07:48 iter: 99 total_loss: 0.9627 loss_cls: 0.2082 loss_box_reg: 0.1535 loss_rpn_cls: 0.259 loss_rpn_loc: 0.1043 time: 1.7980 last_time: 0.9889 last_data_time: 0.0363 lr: 4.0798e-05 max_mem: 4961M
[11/19 01:39:26 d2.util.events]: eta: 00:07:48 iter: 119 total_loss: 0.8666 loss_cls: 0.187 loss_box_reg: 0.1976 loss_rpn_cls: 0.2326 loss_rpn_loc: 0.1561 time: 1.9584 last_time: 0.4428 data_time: 1.1497 last_data_time: 0.0011 lr: 5.0965e-05 max_mem: 4961M
[11/19 01:40:15 d2.util.events]: eta: 00:08:24 iter: 139 total_loss: 0.7963 loss_cls: 0.1646 loss_box_reg: 0.1664 loss_rpn_cls: 0.2404 loss_rpn_loc: 0.1446 time: 1.8746 last_time: 0.4055 data_time: 1.1528 last_data_time: 0.0013 lr: 6.0844e-05 max_mem: 4961M
[11/19 01:40:15 d2.util.events]: eta: 00:08:24 iter: 259 total_loss: 0.7913 loss_cls: 0.1687 loss_box_reg: 0.1687 loss_rpn_cls: 0.2404 loss_rpn_loc: 0.1446 time: 1.8646 last_time: 0.4055 data_time: 1.1528 last_data_time: 0.0013 lr: 6.0844e-05 max_mem: 4961M
[11/19 01:40:15 d2.util.events]: eta: 00:08:48 iter: 159 total_loss: 0.8045 loss_cls: 0.1457 loss_box_reg: 0.2089 loss_rpn_cls: 0.1499 loss_rpn_loc: 0.2285 time: 1.4948 last_time: 1.8658 data_time: 1.0734 last_data_time: 1.3316 lr: 7.9671e-05 max_mem: 4961M
[11/19 01:40:15 d2.util.events]: eta: 00:08:48 iter: 179 total_loss: 0.8883 loss_cls: 0.1479 loss_box_reg: 0.2128 loss_rpn_cls: 0.1863 loss_rpn_loc: 0.2213 time: 1.4871 last_time: 0.5274 data_time: 1.0913 last_data_time: 0.0583 lr: 8.965e-05 max_mem: 4961M
[11/19 01:40:15 d2.util.events]: eta: 00:08:48 iter: 199 total_loss: 0.8864 loss_cls: 0.1762 loss_box_reg: 0.2111 loss_rpn_cls: 0.1598 loss_rpn_loc: 0.2391 time: 1.4871 last_time: 0.6692 data_time: 1.0724 last_data_time: 0.3115 lr: 8.00811964 max_mem: 4961M
[11/19 01:42:28 d2.util.events]: eta: 00:08:36 iter: 219 total_loss: 0.8864 loss_cls: 0.1762 loss_box_reg: 0.2111 loss_rpn_cls: 0.1598 loss_rpn_loc: 0.2391 time: 1.4871 last_time: 0.6692 data_time: 1.0724 last_data_time: 0.3115 lr: 8.00811964 max_mem: 4961M
[11/19 01:42:53 d2.util.events]: eta: 00:08:09 iter: 239 total_loss: 0.8284 loss_cls: 0.1584 loss_box_reg: 0.2042 loss_rpn_cls: 0.1689 loss_rpn_loc: 0.1155 time: 1.4787 last_time: 0.8217 data_time: 0.6912 last_data_time: 0.1476 lr: 8.00811964 max_mem: 4961M
[11/19 01:43:01 d2.util.events]: eta: 00:08:24 iter: 259 total_loss: 0.8146 loss_cls: 0.1488 loss_box_reg: 0.2042 loss_rpn_cls: 0.1689 loss_rpn_loc: 0.1155 time: 1.4971 last_time: 0.8185 data_time: 0.8323 last_data_time: 0.8323 lr: 8.00811964 max_mem: 4961M
[11/19 01:44:26 d2.util.events]: eta: 00:08:24 iter: 279 total_loss: 0.7913 loss_cls: 0.1687 loss_box_reg: 0.2042 loss_rpn_cls: 0.1689 loss_rpn_loc: 0.1155 time: 1.4946 last_time: 0.8146 data_time: 0.8323 last_data_time: 0.8323 lr: 8.00811964 max_mem: 4961M
[11/19 01:44:58 d2.util.events]: eta: 00:08:36 iter: 319 total_loss: 0.7664 loss_cls: 0.1433 loss_box_reg: 0.2132 loss_rpn_cls: 0.0935 loss_rpn_loc: 0.1846 time: 1.9598 last_time: 1.4022 data_time: 1.1304 last_data_time: 0.0166 lr: 8.00811964 max_mem: 4961M
[11/19 01:44:58 d2.util.events]: eta: 00:08:36 iter: 339 total_loss: 0.7664 loss_cls: 0.1433 loss_box_reg: 0.2132 loss_rpn_cls: 0.0935 loss_rpn_loc: 0.1846 time: 1.9598 last_time: 1.4022 data_time: 1.1304 last_data_time: 0.0166 lr: 8.00811964 max_mem: 4961M
[11/19 01:45:58 d2.util.events]: eta: 00:08:24 iter: 359 total_loss: 0.7376 loss_cls: 0.1373 loss_box_reg: 0.2107 loss_rpn_cls: 0.1717 loss_rpn_loc: 0.1717 time: 1.4986 last_time: 1.4364 data_time: 1.2577 last_data_time: 0.9737 lr: 8.00811964 max_mem: 4961M
[11/19 01:46:26 d2.util.events]: eta: 00:08:24 iter: 379 total_loss: 0.7653 loss_cls: 0.1323 loss_box_reg: 0.2142 loss_rpn_cls: 0.0980 loss_rpn_loc: 0.1835 time: 1.4947 last_time: 1.7863 data_time: 0.8985 last_data_time: 1.1993 lr: 8.00811964 max_mem: 4961M
[11/19 01:46:26 d2.util.events]: eta: 00:08:24 iter: 399 total_loss: 0.7653 loss_cls: 0.1323 loss_box_reg: 0.2142 loss_rpn_cls: 0.0980 loss_rpn_loc: 0.1835 time: 1.4947 last_time: 1.7863 data_time: 0.8985 last_data_time: 1.1993 lr: 8.00811964 max_mem: 4961M
[11/19 01:47:36 iter: 419 total_loss: 0.6661 loss_cls: 0.1445 loss_box_reg: 0.2047 loss_rpn_cls: 0.0654 loss_rpn_loc: 0.1011 time: 1.4949 last_time: 0.5977 data_time: 0.9247 last_data_time: 0.0020 lr: 8.00811964 max_mem: 4961M
[11/19 01:47:55 d2.util.events]: eta: 00:08:12 iter: 439 total_loss: 0.7674 loss_cls: 0.1355 loss_box_reg: 0.2064 loss_rpn_cls: 0.0887 loss_rpn_loc: 0.1285 time: 1.4928 last_time: 1.4057 data_time: 0.9888 last_data_time: 0.0022 lr: 8.00811964 max_mem: 4961M
[11/19 01:48:13 d2.util.events]: eta: 00:08:24 iter: 459 total_loss: 0.6544 loss_cls: 0.1106 loss_box_reg: 0.1821 loss_rpn_cls: 0.0824 loss_rpn_loc: 0.1144 time: 1.4931 last_time: 1.5763 data_time: 1.0456 last_data_time: 0.0123 lr: 8.00811964 max_mem: 4961M
[11/19 01:48:55 d2.util.events]: eta: 00:08:24 iter: 479 total_loss: 0.6544 loss_cls: 0.1106 loss_box_reg: 0.1821 loss_rpn_cls: 0.0824 loss_rpn_loc: 0.1144 time: 1.4931 last_time: 1.5763 data_time: 1.0456 last_data_time: 0.0123 lr: 8.00811964 max_mem: 4961M
[11/19 01:49:29 d2.util.events]: eta: 00:08:00 iter: 499 total_loss: 0.6528 loss_cls: 0.1334 loss_box_reg: 0.2325 loss_rpn_cls: 0.070909 loss_rpn_loc: 0.1367 time: 1.4078 last_time: 1.2649 data_time: 1.1099 last_data_time: 2.0959 lr: 8.00811964 max_mem: 4961M
[11/19 01:49:29 d2.engine.hooks]: Overall training speed: 88 iterations in 0:12:29 (1.4978 s / it)
[11/19 01:49:29 d2.engine.hooks]: Total training time: 0:32:28 (4.0980 s on hooks)
```

Final loss plot:

Because tensorboard was not working for me, I plotted this using matlab from the data below the plot:



```

roi_heads.box_predictor.bbox_pred.bias, weight
roi_heads.box_predictor.cls_score.bias, weight
[11/19 02:06:37 d2.engine.train_loop]: Starting training from iteration 0
[11/19 02:06:37 d2.engine.train_loop]: iter: 15 total_loss: 2.686 loss_cls: 0.5776 loss_box_reg: 0.3394 loss_rpn_cls: 1.384 loss_rpn_loc: 2.284 time: 2.1071 last_time: 1.6757 data_time: 0.8221 last_data_time: 0.6064 lr: 1.9481e-06 max_mem: 5301M
[11/19 02:13:13 d2.util.events]: eta: 01:11:31 iter: 39 total_loss: 0.4942 loss_cls: 0.2861 loss_box_reg: 0.2774 loss_rpn_cls: 0.4281 loss_rpn_loc: 1.2088 last_time: 2.5772 data_time: 0.4476 last_data_time: 0.8166 lr: 1.9481e-06 max_mem: 5081M
[11/19 02:13:13 d2.util.events]: eta: 01:11:31 iter: 59 total_loss: 0.4041 loss_cls: 0.2746 loss_box_reg: 0.2891 loss_rpn_cls: 0.4551 loss_rpn_loc: 1.2088 last_time: 2.5772 data_time: 0.4476 last_data_time: 0.8166 lr: 1.9481e-06 max_mem: 5081M
[11/19 02:13:13 d2.util.events]: eta: 01:11:31 iter: 79 total_loss: 0.1711 loss_cls: 0.3617 loss_box_reg: 0.1399 loss_rpn_cls: 0.19 loss_rpn_loc: 0.1626 time: 2.0589 last_time: 1.0397 data_time: 0.4052 last_data_time: 0.0037 lr: 7.0424e-05 max_mem: 5083M
[11/19 02:13:13 d2.util.events]: eta: 01:11:31 iter: 99 total_loss: 0.9668 loss_cls: 0.1652 loss_box_reg: 0.1266 loss_rpn_cls: 0.2097 loss_rpn_loc: 0.2786 time: 2.0080 last_time: 1.3315 data_time: 0.2371 last_data_time: 0.0009 lr: 9.0401e-05 max_mem: 5083M
[11/19 02:13:13 d2.util.events]: eta: 00:08:24 iter: 119 total_loss: 0.8464 loss_cls: 0.1482 loss_box_reg: 0.1195 loss_rpn_cls: 0.1819 loss_rpn_loc: 0.2002 time: 2.0081 last_time: 1.3315 data_time: 0.2371 last_data_time: 0.0009 lr: 9.0401e-05 max_mem: 5083M
[11/19 02:13:13 d2.util.events]: eta: 00:08:24 iter: 139 total_loss: 0.8464 loss_cls: 0.1482 loss_box_reg: 0.1195 loss_rpn_cls: 0.1819 loss_rpn_loc: 0.2002 time: 1.9795 last_time: 1.6386 data_time: 0.3651 last_data_time: 0.1812 lr: 8.008153934 max_mem: 5080M
[11/19 02:13:13 d2.util.events]: eta: 00:08:24 iter: 159 total_loss: 0.7953 loss_cls: 0.1765 loss_box_reg: 0.1263 loss_rpn_cls: 0.1765 loss_rpn_loc: 0.2153 time: 1.9795 last_time: 1.6386 data_time: 0.3651 last_data_time: 0.1812 lr: 8.008153934 max_mem: 5080M
[11/19 02:13:13 d2.util.events]: eta: 00:08:24 iter: 179 total_loss: 0.7655 loss_cls: 0.1354 loss_box_reg: 0.1354 loss_rpn_cls: 0.1765 loss_rpn_loc: 0.2153 time: 1.9792 last_time: 1.7999 data_time: 0.3708 last_data_time: 0.1668 lr: 8.00817932 max_mem: 5080M
[11/19 02:13:13 d2.util.events]: eta: 00:08:24 iter: 199 total_loss: 0.6721 loss_cls: 0.1145 loss_box_reg: 0.1263 loss_rpn_cls: 0.1874 loss_rpn_loc: 0.2153 time: 1.9792 last_time: 1.7999 data_time: 0.3708 last_data_time: 0.1668 lr: 8.00817932 max_mem: 5080M
[11/19 02:13:13 d2.util.events]: eta: 00:08:31 iter: 219 total_loss: 0.6721 loss_cls: 0.1145 loss_box_reg: 0.1263 loss_rpn_cls: 0.1874 loss_rpn_loc: 0.2153 time: 1.9792 last_time: 1.7999 data_time: 0.3708 last_data_time: 0.1668 lr: 8.00817932 max_mem: 5080M
[11/19 02:13:13 d2.util.events]: eta: 00:08:31 iter: 239 total_loss: 0.5739 loss_cls: 0.101 loss_box_reg: 0.1265 loss_rpn_cls: 0.0672 loss_rpn_loc: 0.1614 time: 1.9630 last_time: 2.1663 data_time: 0.4309 last_data_time: 0.4551 lr: 8.00827922 max_mem: 5087M
[11/19 02:13:13 d2.util.events]: eta: 00:08:43 iter: 259 total_loss: 0.5739 loss_cls: 0.101 loss_box_reg: 0.1265 loss_rpn_cls: 0.0672 loss_rpn_loc: 0.1614 time: 1.9630 last_time: 2.1663 data_time: 0.4309 last_data_time: 0.4551 lr: 8.00827922 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:02 iter: 259 total_loss: 0.6754 loss_cls: 0.1224 loss_box_reg: 0.1263 loss_rpn_cls: 0.1623 loss_rpn_loc: 0.1477 time: 1.9626 last_time: 1.6599 data_time: 0.3708 last_data_time: 0.0012 lr: 8.0080992 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:02 iter: 279 total_loss: 0.6754 loss_cls: 0.1224 loss_box_reg: 0.1263 loss_rpn_cls: 0.1623 loss_rpn_loc: 0.1477 time: 1.9626 last_time: 1.6599 data_time: 0.3708 last_data_time: 0.0012 lr: 8.0080992 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:56 iter: 299 total_loss: 0.5414 loss_cls: 0.1167 loss_box_reg: 0.1262 loss_rpn_cls: 0.0938 loss_rpn_loc: 0.1248 time: 1.9627 last_time: 1.2369 data_time: 0.4077 last_data_time: 0.0052 lr: 8.00830196 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:13 iter: 309 total_loss: 0.5179 loss_cls: 0.1126 loss_box_reg: 0.1258 loss_rpn_cls: 0.0957 loss_rpn_loc: 0.1248 time: 1.9558 last_time: 4.2205 data_time: 0.2442 last_data_time: 2.298 lr: 8.0083514 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:13 iter: 329 total_loss: 0.5179 loss_cls: 0.1126 loss_box_reg: 0.1258 loss_rpn_cls: 0.0957 loss_rpn_loc: 0.1248 time: 1.9558 last_time: 4.2205 data_time: 0.2442 last_data_time: 2.298 lr: 8.0083514 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:13 iter: 339 total_loss: 0.5973 loss_cls: 0.1163 loss_box_reg: 0.1251 loss_rpn_cls: 0.0915 loss_rpn_loc: 0.1247 time: 1.9668 last_time: 1.8028 data_time: 0.4387 last_data_time: 1.2378 lr: 8.0083911 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:13 iter: 349 total_loss: 0.4091 loss_cls: 0.08894 loss_box_reg: 0.1661 loss_rpn_cls: 0.06331 loss_rpn_loc: 0.1085 time: 1.9781 last_time: 1.9752 data_time: 0.4700 last_data_time: 0.0071 lr: 8.00841908 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:26 iter: 419 total_loss: 0.4091 loss_cls: 0.08894 loss_box_reg: 0.1661 loss_rpn_cls: 0.06331 loss_rpn_loc: 0.1085 time: 1.9781 last_time: 1.9752 data_time: 0.4700 last_data_time: 0.0071 lr: 8.00841908 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:26 iter: 439 total_loss: 0.4091 loss_cls: 0.08894 loss_box_reg: 0.1661 loss_rpn_cls: 0.06331 loss_rpn_loc: 0.1085 time: 1.9781 last_time: 1.9752 data_time: 0.4700 last_data_time: 0.0071 lr: 8.00841908 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:36 iter: 439 total_loss: 0.4091 loss_cls: 0.08894 loss_box_reg: 0.1661 loss_rpn_cls: 0.06331 loss_rpn_loc: 0.1085 time: 1.9781 last_time: 1.9752 data_time: 0.4700 last_data_time: 0.0071 lr: 8.00841908 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:36 iter: 479 total_loss: 0.4742 loss_cls: 0.0911 loss_box_reg: 0.189 loss_rpn_cls: 0.08282 loss_rpn_loc: 0.1086 time: 1.9651 last_time: 1.6916 data_time: 0.3910 last_data_time: 0.0046 lr: 8.00847982 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:40 iter: 479 total_loss: 0.4742 loss_cls: 0.0911 loss_box_reg: 0.189 loss_rpn_cls: 0.08282 loss_rpn_loc: 0.1086 time: 1.9651 last_time: 1.6916 data_time: 0.3910 last_data_time: 0.0046 lr: 8.00847982 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:40 iter: 499 total_loss: 0.4492 loss_cls: 0.088748 loss_box_reg: 0.1753 loss_rpn_cls: 0.082629 loss_rpn_loc: 0.1072 time: 1.9642 last_time: 1.7994 data_time: 0.3994 last_data_time: 0.0166 lr: 8.008499 max_mem: 5087M
[11/19 02:13:43 d2.util.events]: eta: 00:08:40 iter: 499 total_loss: 0.4492 loss_cls: 0.088748 loss_box_reg: 0.1753 loss_rpn_cls: 0.082629 loss_rpn_loc: 0.1072 time: 1.9642 last_time: 1.7994 data_time: 0.3994 last_data_time: 0.0166 lr: 8.008499 max_mem: 5087M

```

COCOevaluator of the validation set:

This is the cocoevaluator of the validation set with the poor learning rate and improved learning rate respectively:

```

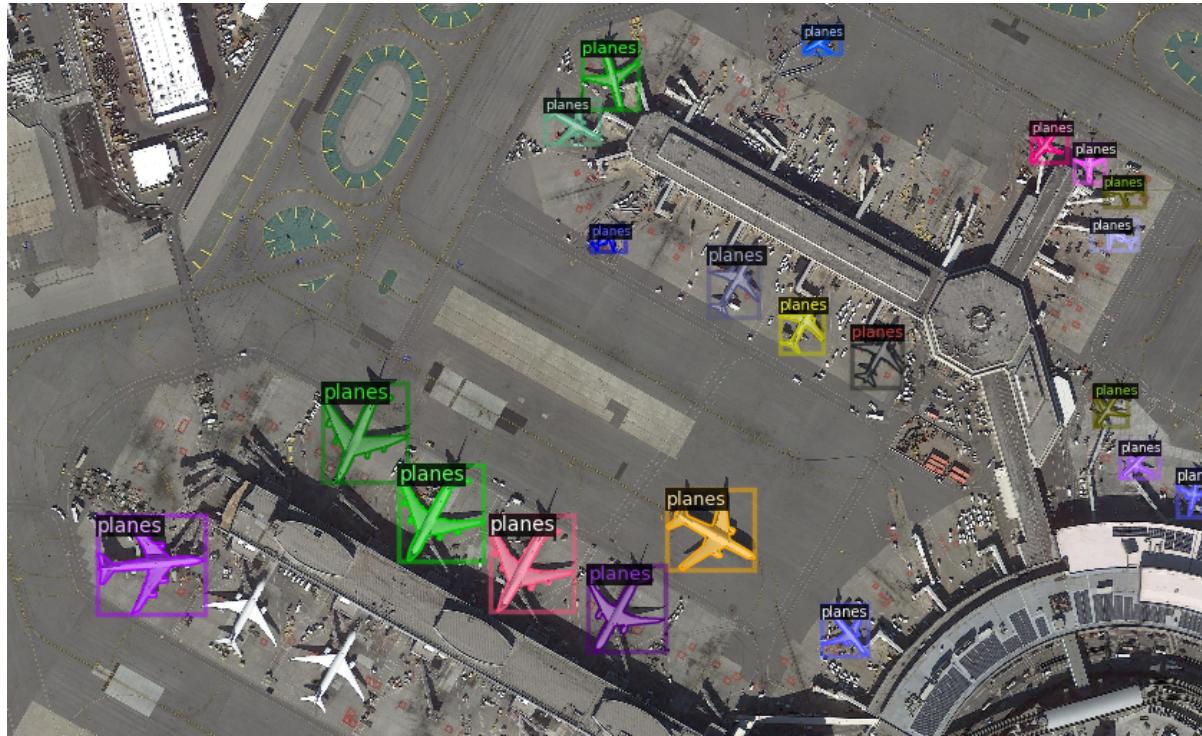
creating index...
index created!
[11/09 23:29:29 d2.evaluation.fast_eval_api]: Evaluate annotation type *bbox*
[11/09 23:29:29 d2.evaluation.fast_eval_api]: COCOeval_opt.evaluate() finished in 0.07 seconds.
[11/09 23:29:29 d2.evaluation.fast_eval_api]: Accumulating evaluation results...
[11/09 23:29:29 d2.evaluation.fast_eval_api]: COCOeval_opt.accumulate() finished in 0.01 seconds.
Average Precision (AP) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.331
Average Precision (AP) @[ IoU=0.50 | area= all | maxDets=100 ] = 0.657
Average Precision (AP) @[ IoU=0.75 | area= all | maxDets=100 ] = 0.293
Average Precision (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.294
Average Precision (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.430
Average Precision (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.409
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 1 ] = 0.019
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 10 ] = 0.169
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.492
Average Recall (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.302
Average Recall (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.512
Average Recall (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.783
[11/09 23:29:29 d2.evaluation.coco_evaluation]: Evaluation results for bbox:
| AP | AP50 | AP75 | APs | APm | AP1 |
|-----|-----|-----|-----|-----|-----|
| 33.150 | 65.669 | 29.264 | 29.362 | 42.972 | 40.938 |
OrderedDict([('bbox', {'AP': 33.149942948868386, 'AP50': 65.66940259858933, 'AP75': 29.264126382299

```

3 Train dataset visualizations:







3 Test dataset visualizations:





Part 2

Hyperparameter settings

batch_size=32, learning_rate=0.0005, num_epochs=50, weight_decay=1e-5

Model Architecture Modifications

New architecture:

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 16, 128, 128]	448
BatchNorm2d-2	[-1, 16, 128, 128]	32
ReLU-3	[-1, 16, 128, 128]	0
conv-4	[-1, 16, 128, 128]	0
Conv2d-5	[-1, 32, 128, 128]	4,640
BatchNorm2d-6	[-1, 32, 128, 128]	64
ReLU-7	[-1, 32, 128, 128]	0
conv-8	[-1, 32, 128, 128]	0
MaxPool2d-9	[-1, 32, 64, 64]	0
down-10	[-1, 32, 64, 64]	0
Conv2d-11	[-1, 64, 64, 64]	18,496
BatchNorm2d-12	[-1, 64, 64, 64]	128
ReLU-13	[-1, 64, 64, 64]	0
conv-14	[-1, 64, 64, 64]	0
MaxPool2d-15	[-1, 64, 32, 32]	0
down-16	[-1, 64, 32, 32]	0
Conv2d-17	[-1, 128, 32, 32]	73,856
BatchNorm2d-18	[-1, 128, 32, 32]	256
ReLU-19	[-1, 128, 32, 32]	0
conv-20	[-1, 128, 32, 32]	0
MaxPool2d-21	[-1, 128, 16, 16]	0
down-22	[-1, 128, 16, 16]	0
Conv2d-23	[-1, 256, 16, 16]	295,168
BatchNorm2d-24	[-1, 256, 16, 16]	512
ReLU-25	[-1, 256, 16, 16]	0
conv-26	[-1, 256, 16, 16]	0
MaxPool2d-27	[-1, 256, 8, 8]	0
down-28	[-1, 256, 8, 8]	0
Conv2d-29	[-1, 512, 8, 8]	1,180,160
BatchNorm2d-30	[-1, 512, 8, 8]	1,024
ReLU-31	[-1, 512, 8, 8]	0
conv-32	[-1, 512, 8, 8]	0
ConvTranspose2d-33	[-1, 512, 16, 16]	1,049,088
Conv2d-34	[-1, 256, 16, 16]	1,179,904
BatchNorm2d-35	[-1, 256, 16, 16]	512
ReLU-36	[-1, 256, 16, 16]	0
conv-37	[-1, 256, 16, 16]	0
up-38	[-1, 256, 16, 16]	0
Conv2d-39	[-1, 256, 16, 16]	884,992
BatchNorm2d-40	[-1, 256, 16, 16]	512
ReLU-41	[-1, 256, 16, 16]	0
conv-42	[-1, 256, 16, 16]	0
ConvTranspose2d-43	[-1, 256, 32, 32]	262,400
Conv2d-44	[-1, 128, 32, 32]	295,040
BatchNorm2d-45	[-1, 128, 32, 32]	256
ReLU-46	[-1, 128, 32, 32]	0
conv-47	[-1, 128, 32, 32]	0
up-48	[-1, 128, 32, 32]	0
Conv2d-49	[-1, 128, 32, 32]	221,312
BatchNorm2d-50	[-1, 128, 32, 32]	256
ReLU-51	[-1, 128, 32, 32]	0
conv-52	[-1, 128, 32, 32]	0
ConvTranspose2d-53	[-1, 128, 64, 64]	65,664
Conv2d-54	[-1, 64, 64, 64]	73,792
BatchNorm2d-55	[-1, 64, 64, 64]	128
ReLU-56	[-1, 64, 64, 64]	0
conv-57	[-1, 64, 64, 64]	0
up-58	[-1, 64, 64, 64]	0
Conv2d-59	[-1, 64, 64, 64]	55,360
BatchNorm2d-60	[-1, 64, 64, 64]	128
ReLU-61	[-1, 64, 64, 64]	0
conv-62	[-1, 64, 64, 64]	0
ConvTranspose2d-63	[-1, 64, 128, 128]	16,448
Conv2d-64	[-1, 32, 128, 128]	18,464
BatchNorm2d-65	[-1, 32, 128, 128]	64
ReLU-66	[-1, 32, 128, 128]	0
conv-67	[-1, 32, 128, 128]	0
up-68	[-1, 32, 128, 128]	0
Conv2d-69	[-1, 32, 128, 128]	13,856
BatchNorm2d-70	[-1, 32, 128, 128]	64
ReLU-71	[-1, 32, 128, 128]	0
conv-72	[-1, 32, 128, 128]	0
Conv2d-73	[-1, 1, 128, 128]	289
conv-74	[-1, 1, 128, 128]	0

Total params: 5,713,313
Trainable params: 5,713,313
Non-trainable params: 0

Input size (MB): 0.19
Forward/backward pass size (MB): 125.50
Params size (MB): 21.79
Estimated Total Size (MB): 147.48

Modifications for the new architecture:

- More Channels: The model can learn better feature representations when there are more channels at each layer.
- More Layers: The model can capture data at a wider range of scales due to additional layers of upsampling and downsampling.
- Bottleneck: A bottleneck layer between the encoder and decoder pathways manages the most abstracted components, enhancing the network's capacity for generalization.
- Skip Connections: Accurate localization for segmentation is facilitated by concatenating encoder features to decoder layers.
- Post-Upsample Convolution: Convolution layers help to enhance features and restore detail after upsampling.
- No Activation in Output: Logits are produced by the last convolution layer and are appropriate for segmentation tasks that call for probability map post-processing.

Training alterations:

I switched the optimizer from SGD to Adam.

I added a learning rate scheduler to reduce the learning rate whenever the loss plateaus.

I saved the model checkpoints periodically so that the training can restart from checkpoints if it trends in a poor direction.

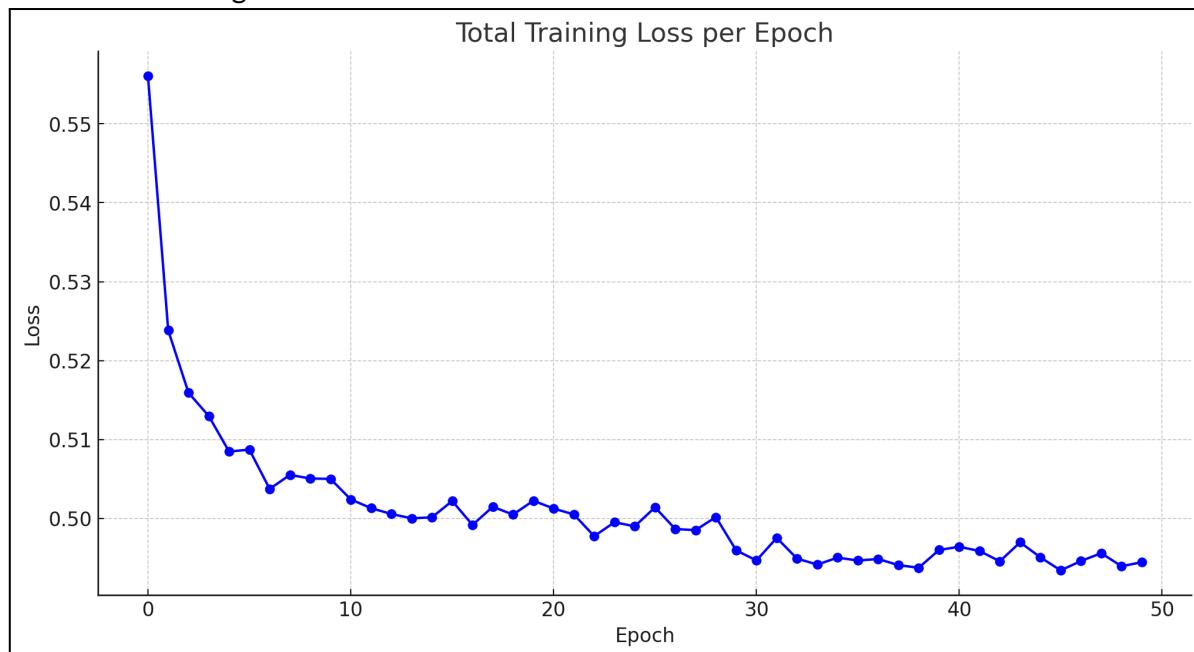
I implemented mixed precision training to speed up training and decrease the memory usage.

I also switched ‘total_loss += loss.cpu().data’ to ‘total_loss += loss.detach().item()’ to only add the value and thus decrease the memory usage.

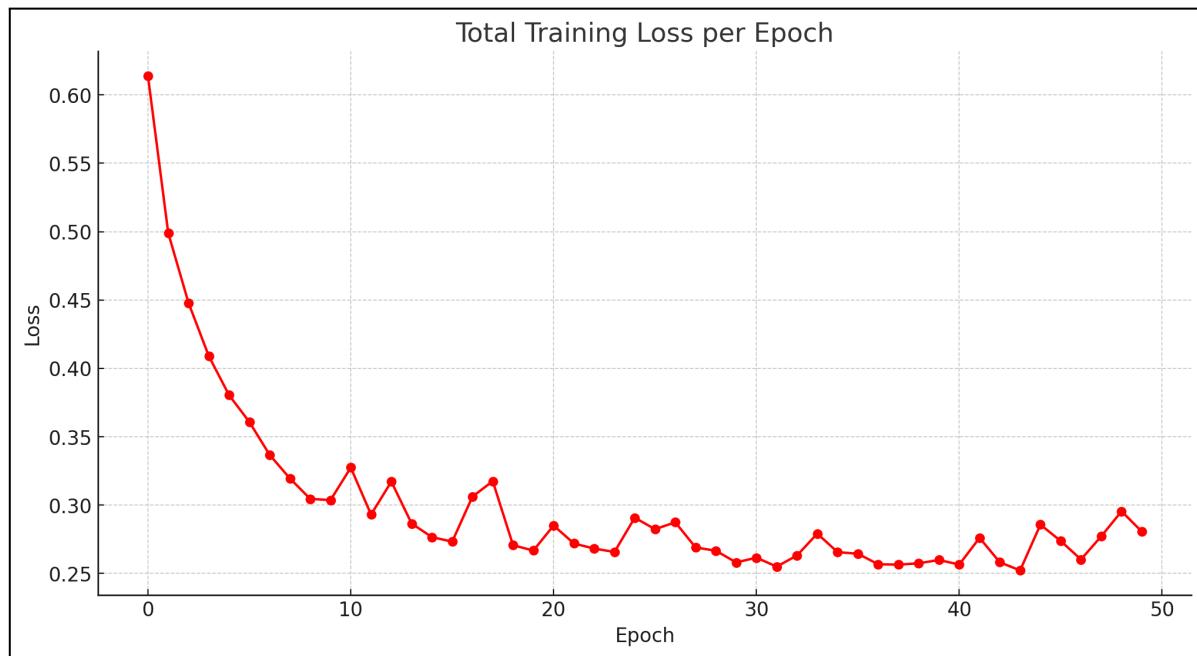
Training loss plot:

As mentioned before, I could not get tensorboard working, so I have plotted it with matlab from the text data.

Trained on 20 images:



When I train the model on fewer images, the loss is lower, but the output performance is worse (possibly due to overfitting). Trained on 3 images:



Mean IoU:

Mean IoU: 0.0997

```
Images index: 3/3      bbox index: 26/30
Images index: 3/3      bbox index: 27/30
Images index: 3/3      bbox index: 28/30
Images index: 3/3      bbox index: 29/30
Images index: 3/3      bbox index: 30/30
/usr/local/lib/python3.10/dist-packages/torch/util
  warnings.warn(_create_warning_msg(
100% [██████████] 9/9 [00:00:00]
#images: 70, Mean IoU: 0.0997
```

Visualize 3 images from the test set and the corresponding predicted masks.

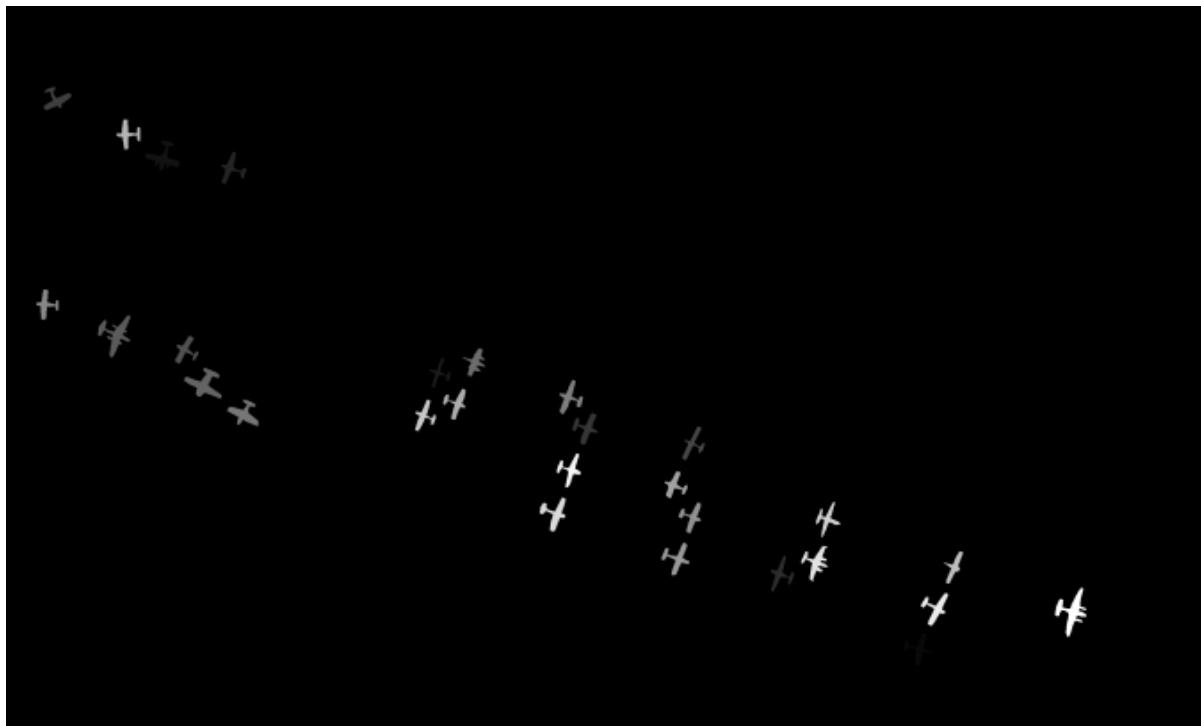
I am not sure why the true masks do not seem to match.

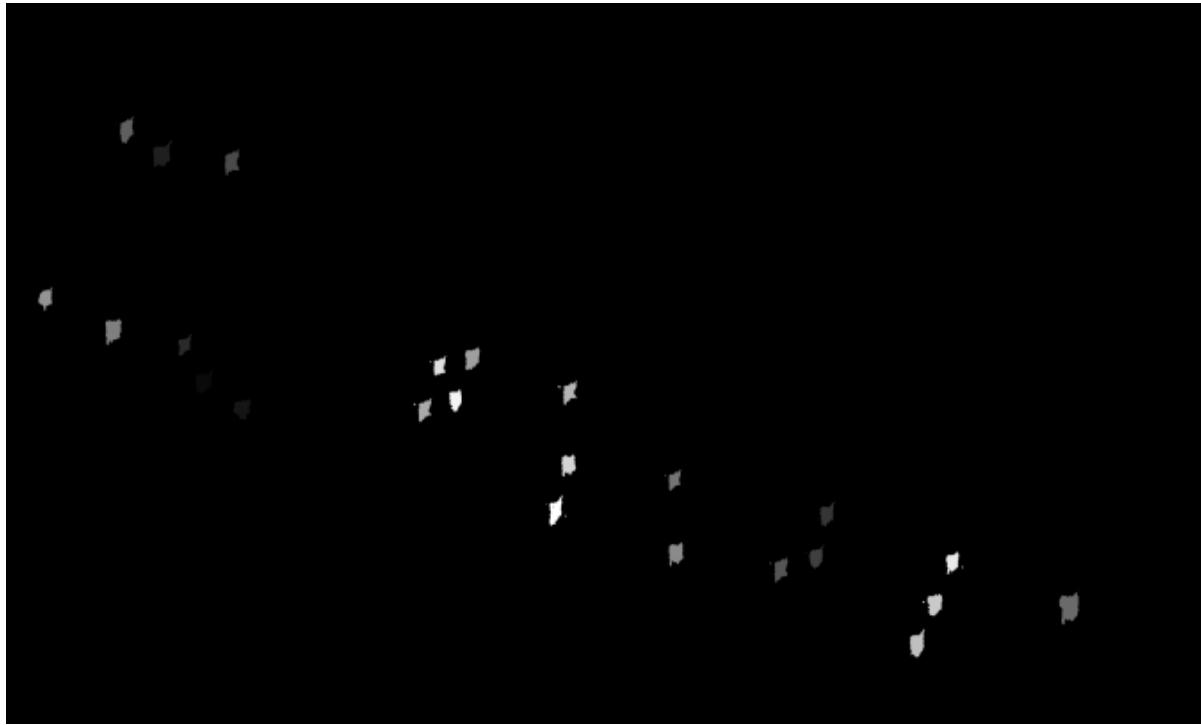
Original image	Predicted mask	True mask



Part 3

I submitted my results as **Jared Tweed** on kaggle with a score of 0.00034. I am not in a group.





Part 4

Here is a visualization produced from part 4. The model performs significantly worse than part 1. Using the Mask R-CNN from Detectron2 in part 4, this model gives us less flexibility, but can be implemented faster than part 3 which uses instance segmentation.

