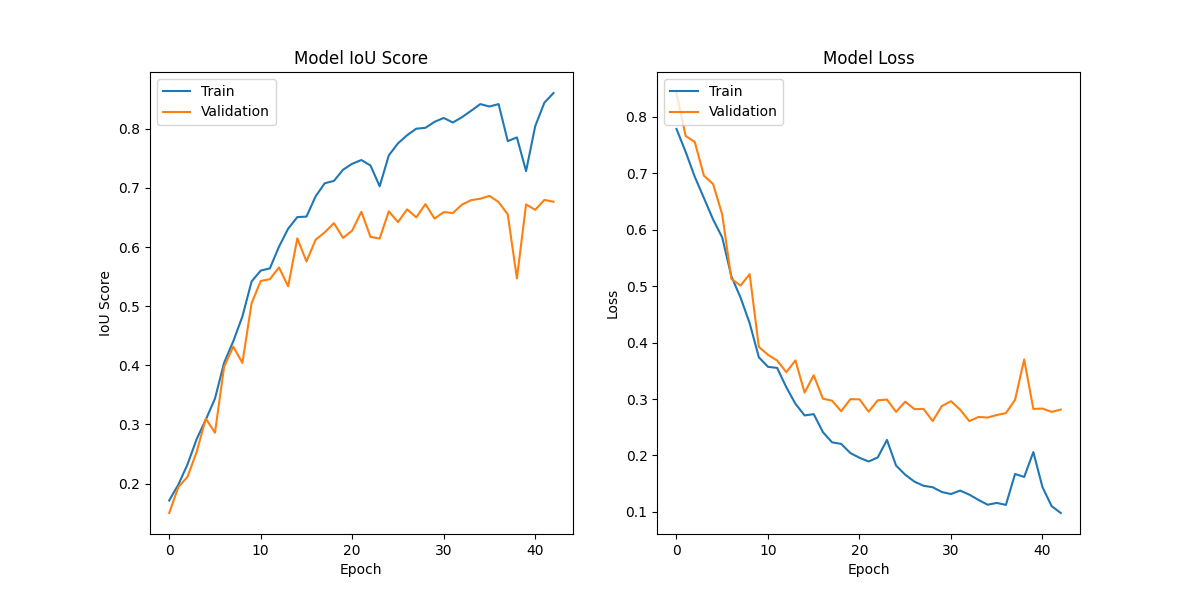
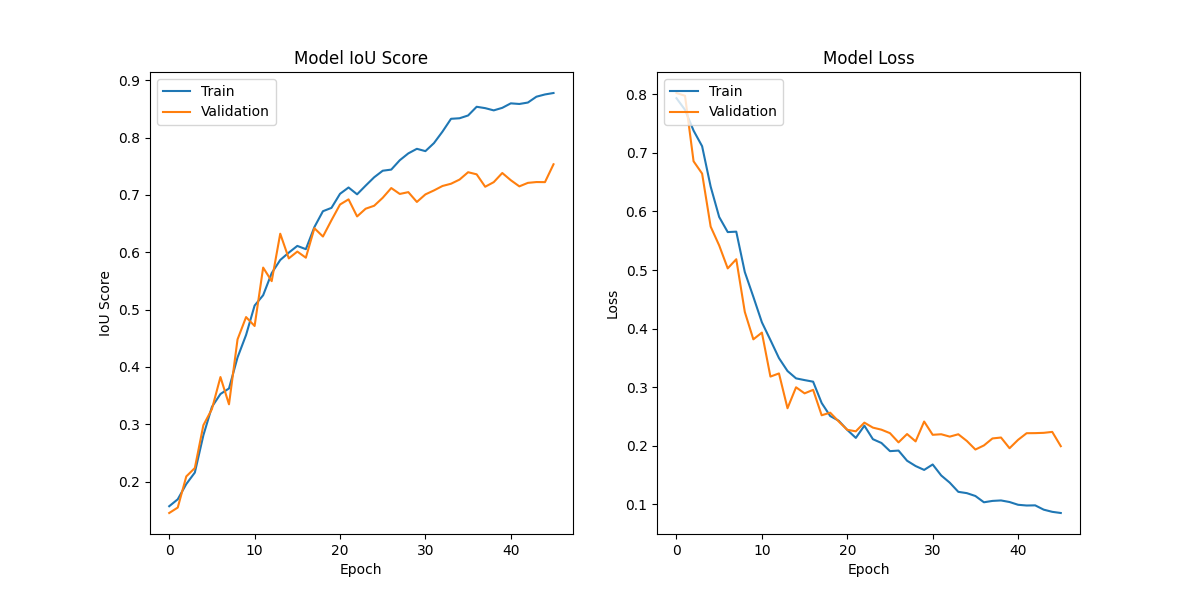
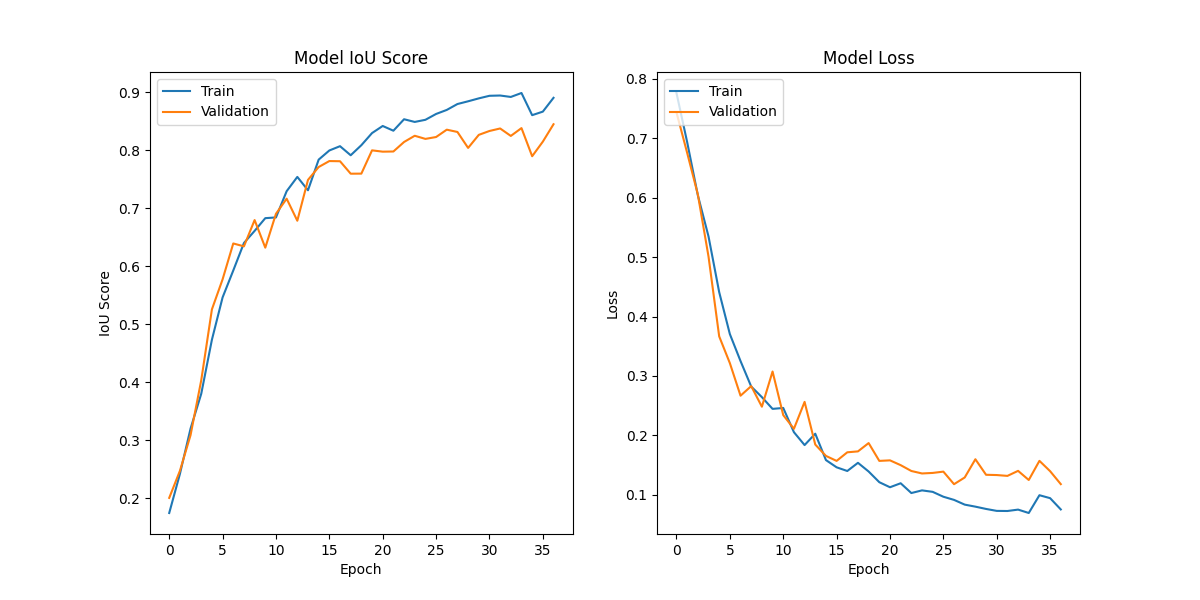
Patched images to 200x200  
  
loss: 0.0614 - iou\_score: 0.9101 - val\_loss: 0.3040 - val\_iou\_score: 0.6919  
  
  
the val iou is poor.  
  
I want to boost it.  
imma get a first 16 filter layer line  


31ms/step - loss: 0.0981 - iou\_score: 0.8605 - val\_loss: 0.2813 - val\_iou\_score: 0.6767  
  
  
which iis much faster but actually less precise.  
  
  
maybe the size is too small.  
  
lets try with the original 400 images, but keep the num filters reduced  
  
  
  
loss: 0.0852 - iou\_score: 0.8779 - val\_loss: 0.1993 - val\_iou\_score: 0.7535  
  
  
much better !

IoU= True Positive (TP)​ / True Positive (TP)+False Positive (FP)+False Negative (FN)

lets try to change the unet a bit (play with num of layers and num filters  
  
  
going from 32 to 256 is worse. Dropping the iou by 3 points  
  
  
  
  
Ok I figured out the output images are 608x608  
  
1rst experiement was to train on 400x400, then resize the test image before and after the predicitons.  
  
F1 score : 0695 : Accuracy : 0.848  
  
  
2nd experiment will involve using a patching technique.  
Images to be predicted will be 256  
  
  
  
loss: 0.0756 - iou\_score: 0.8906 - val\_loss: 0.1182 - val\_iou\_score: 0.8451  
  
WOOOOH !  
  
  
  
  
  
We now need data augmentation techniques in order to attain   
  
3rd will be to use best case and try to reduce the unet depth and tweak the drop out rates, and play with the number of filters  
  
  
  
  
  
what are the last updates:  
  
we added 90\*k rotation and increased the dataset size by 4. It helped cranked up the F1 score from 82 to 86,7 §  
  
Next improvements will concern :  
  
- different loss, metrics, learning rates and optimizers

* Smaller batch size  
  shallower unet and deeper unet (try different depth)
* Add post processing

Trying to reduce : no incidence on val iou : 83, loss 0.128  
trying to increase : a bit worse :val\_iou : 0,82 loss 0,135   
  
  
  
  
  
  
  
  
at this point we have a f1 score of 86.8 !! which is very good. But I see a degree of improvement !  
  
instead of rotating randomly. Let rotate every 45° ! (this is test image 50 after prediction (mred mask), we can clearly see the model trying to find horizontal and vertical roads more successfully than 45° angles roads.  
lets try it  
  
  
  
IOU is said to be lower !  
  
but road classification seems well better !!  
  
but F1 on submission did not increase…  
  
next : add hue variation or some saturation variation, brightness.. (randomization)  
  
IOU IS A LIAR !