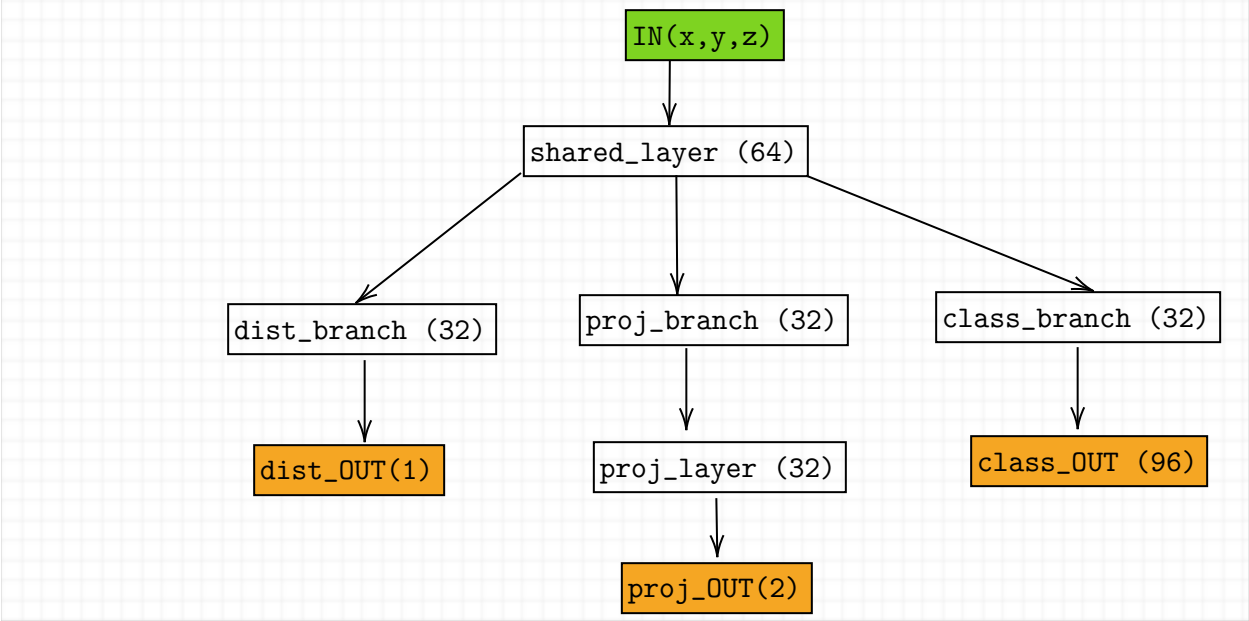
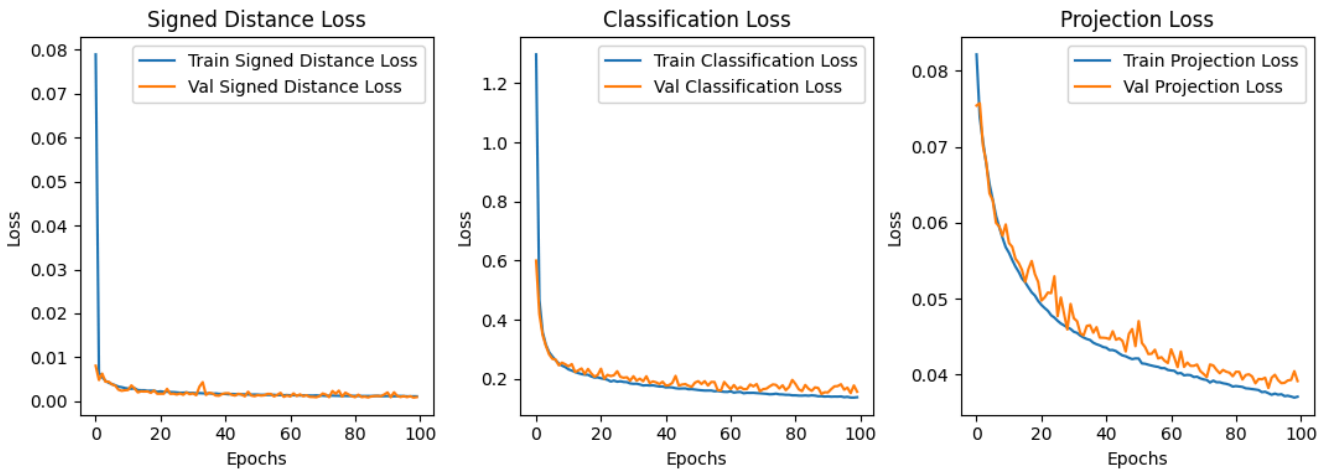
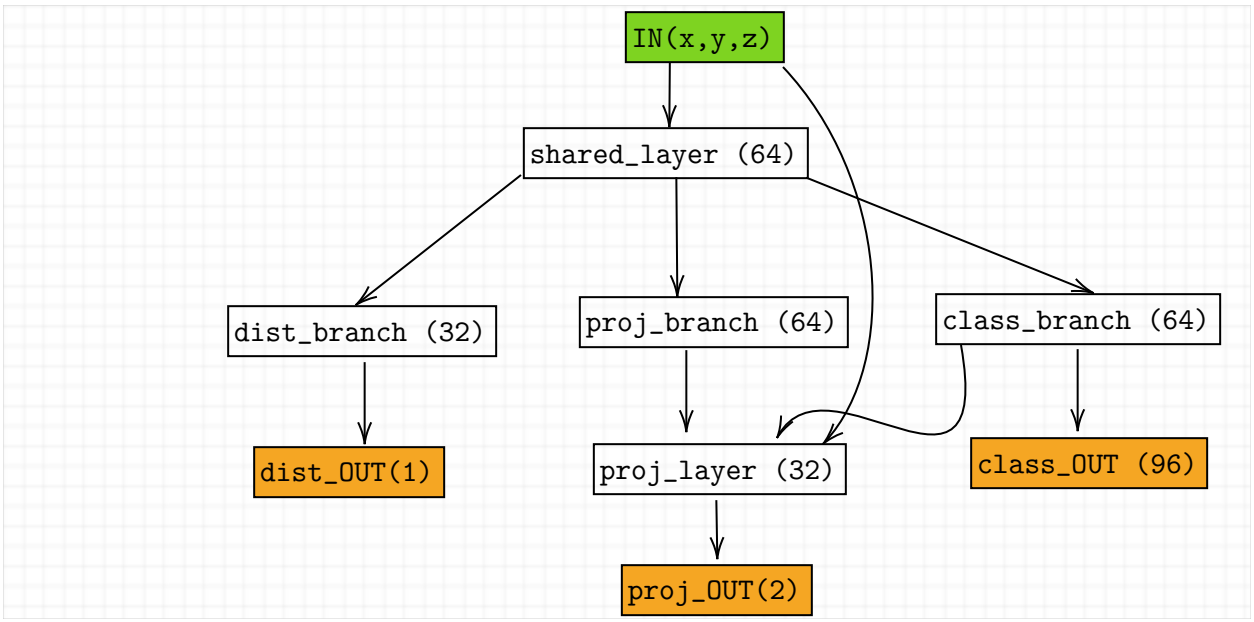
Here we try Multitask because it improves generalization, so potentially we could output just **one** pair (ξ_1,ξ_2) → proj_output.size = (2,)

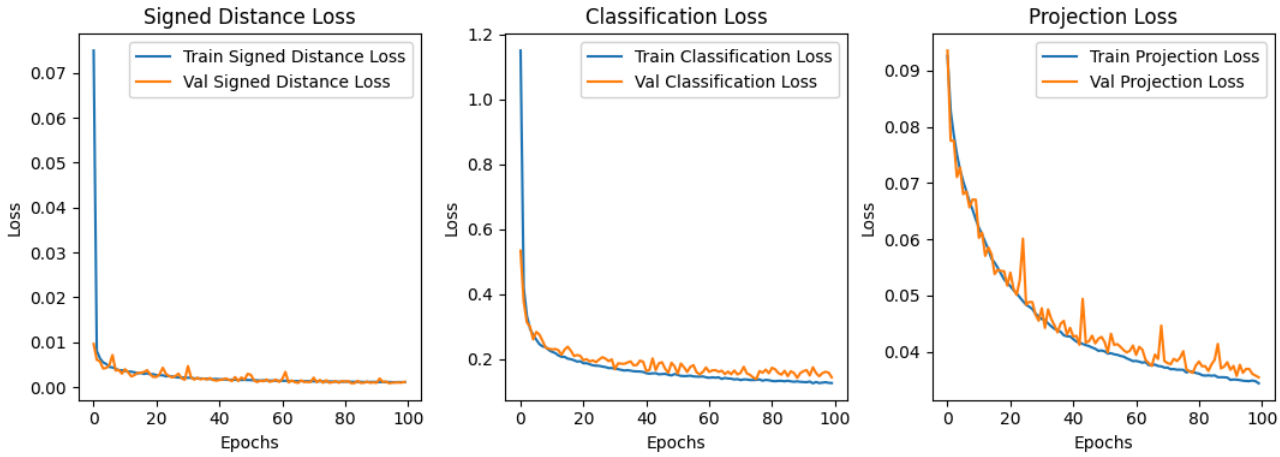
First working MultiTask



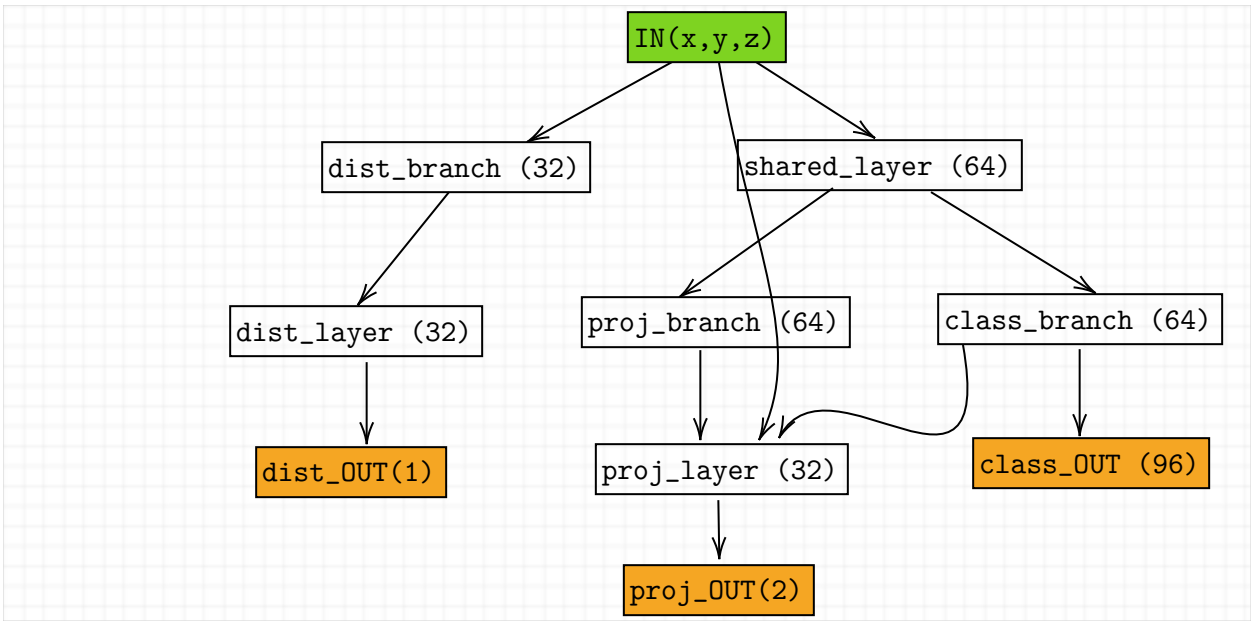


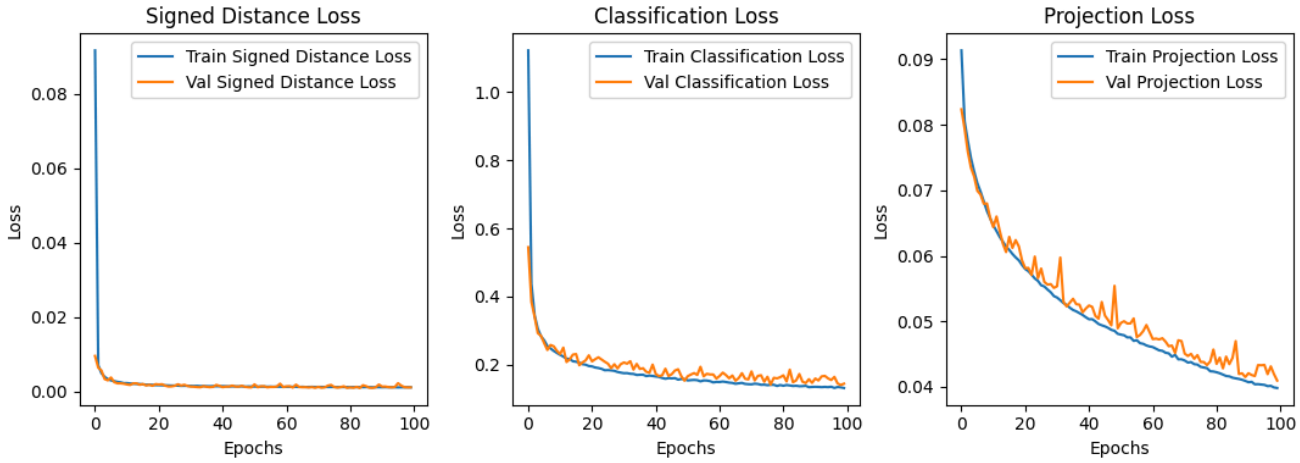
Calibration1



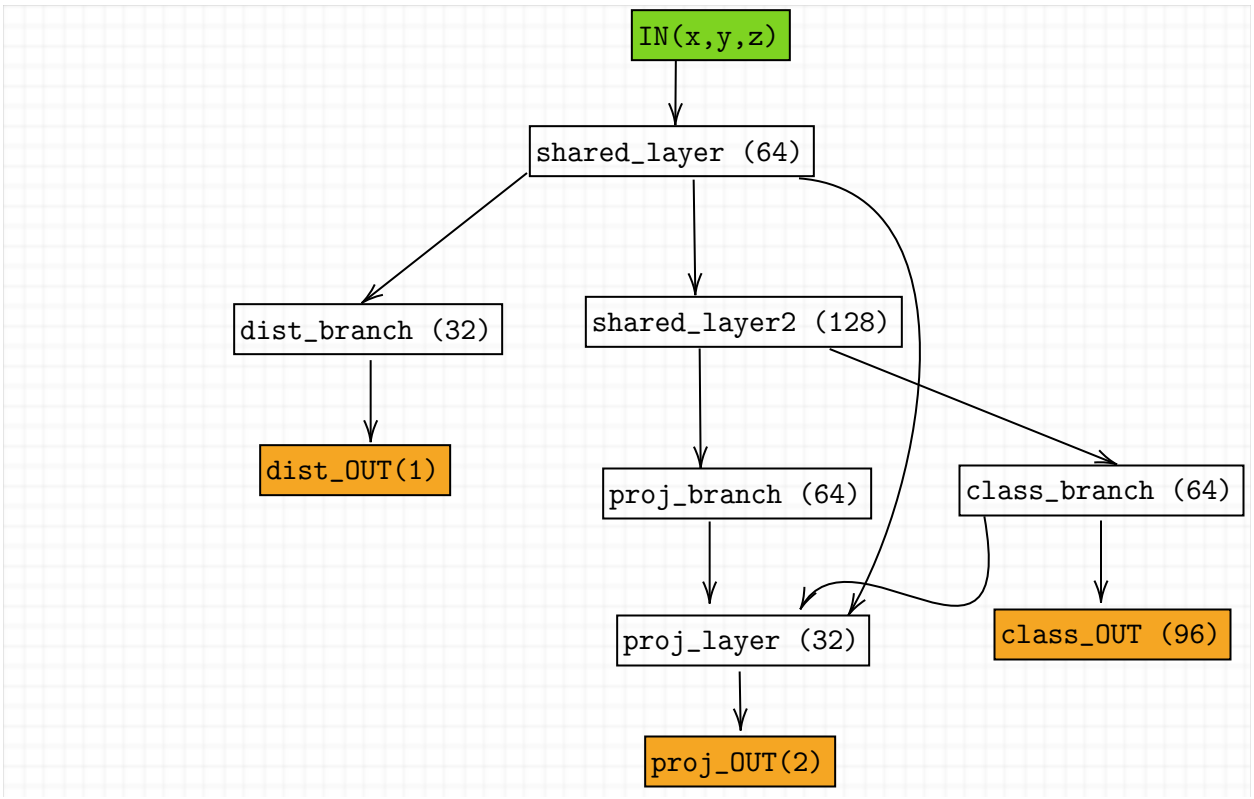


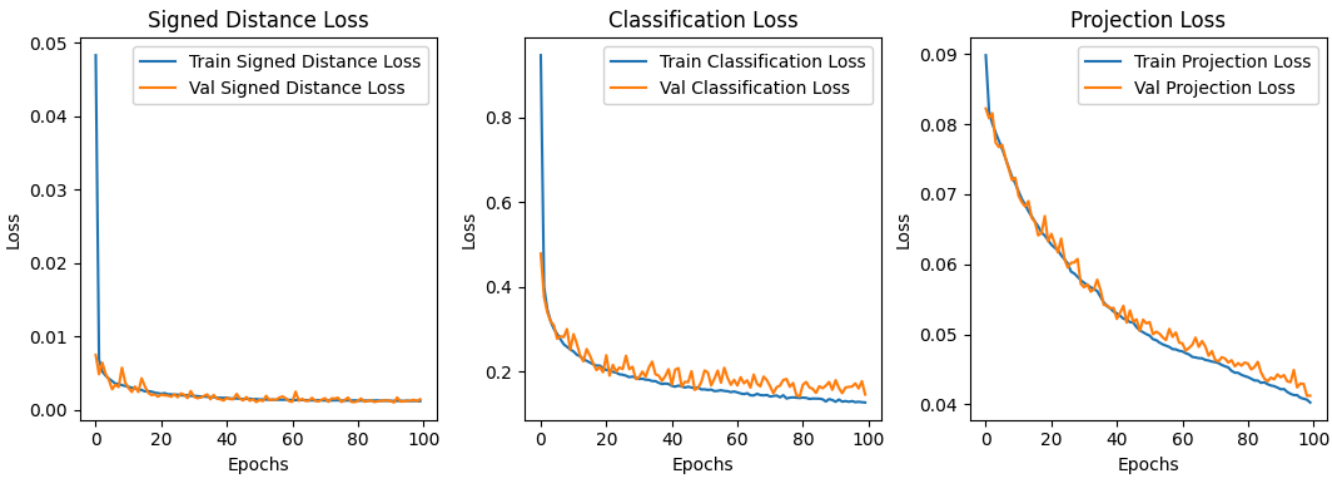
Calibration1_2



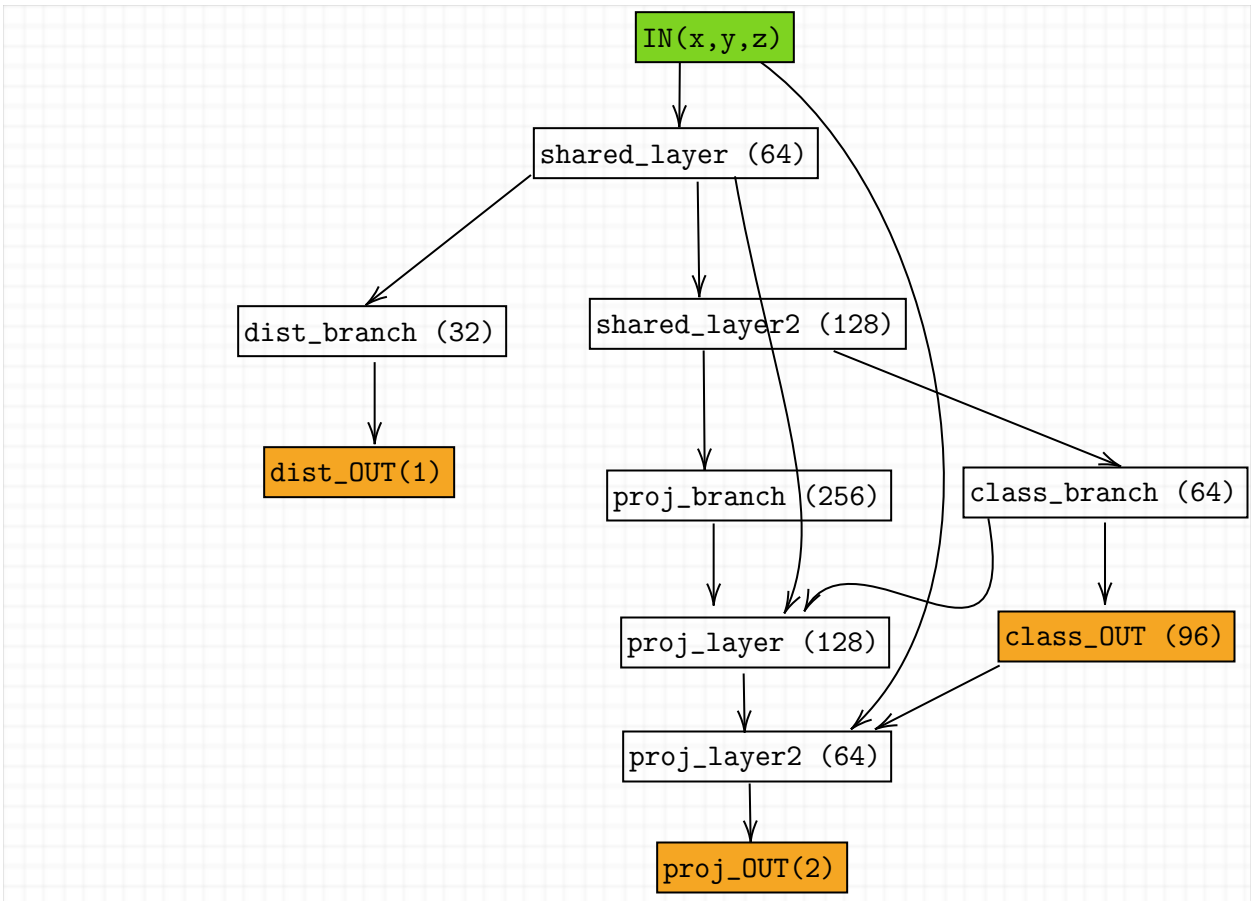


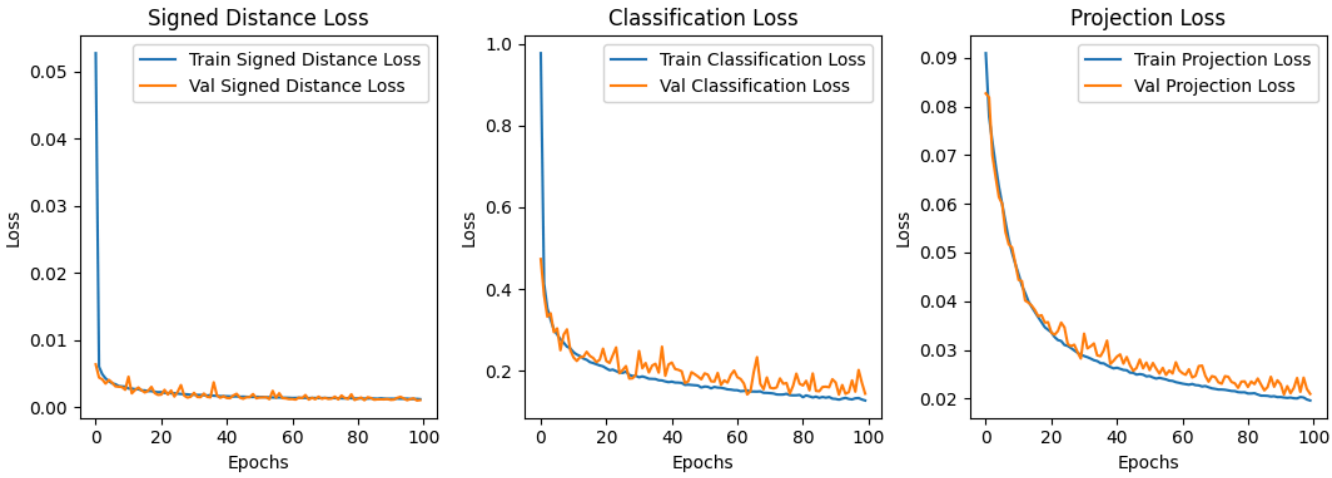
Calibration2



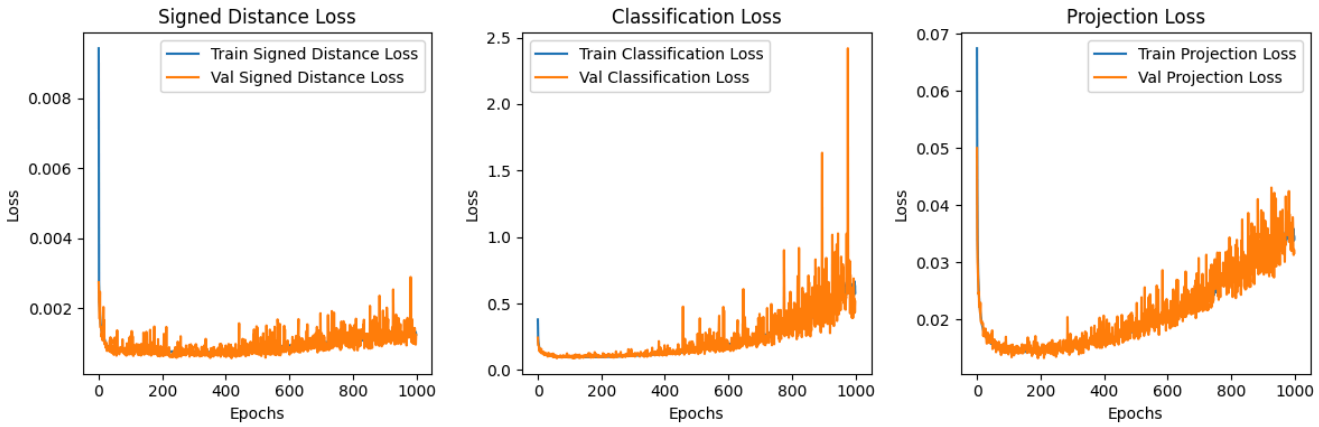


Calibration3

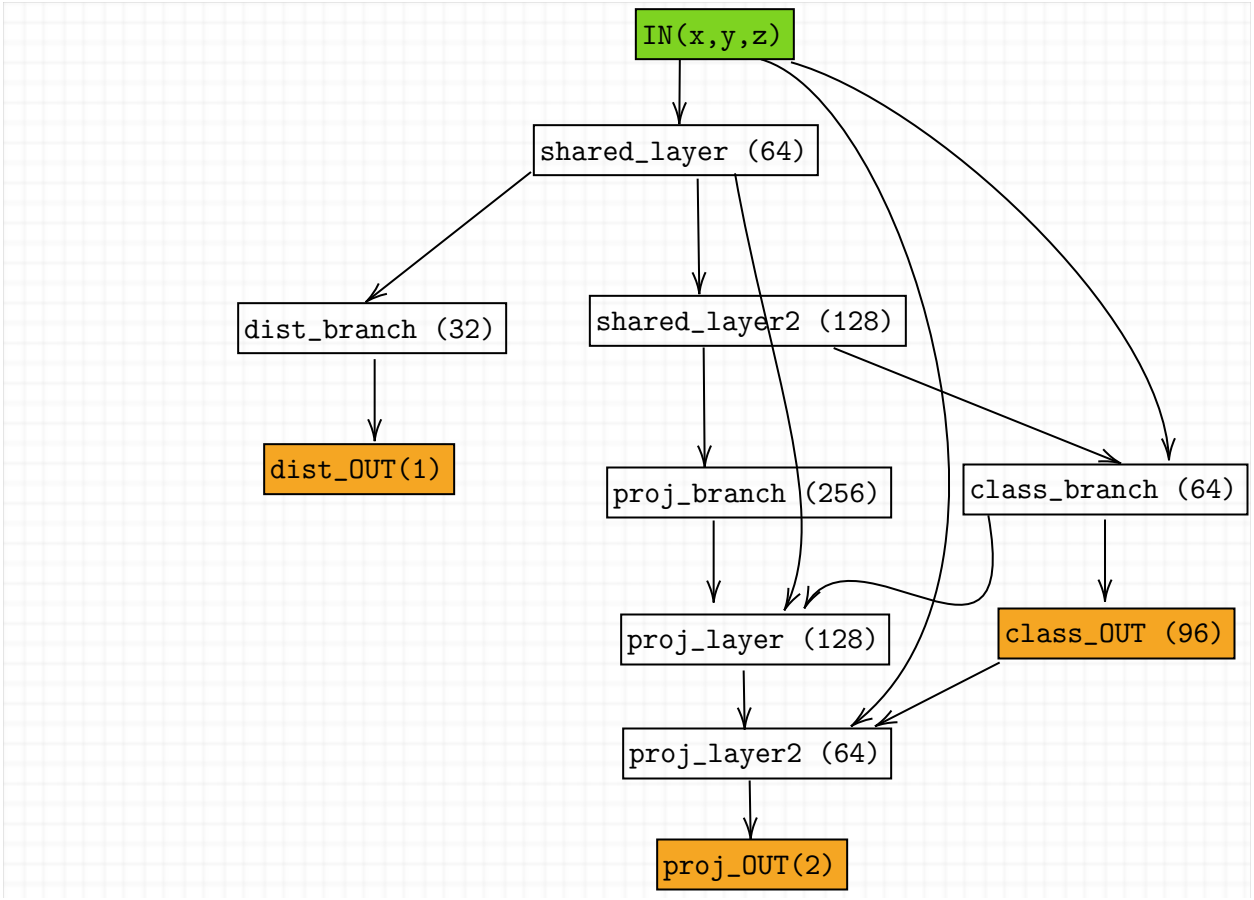


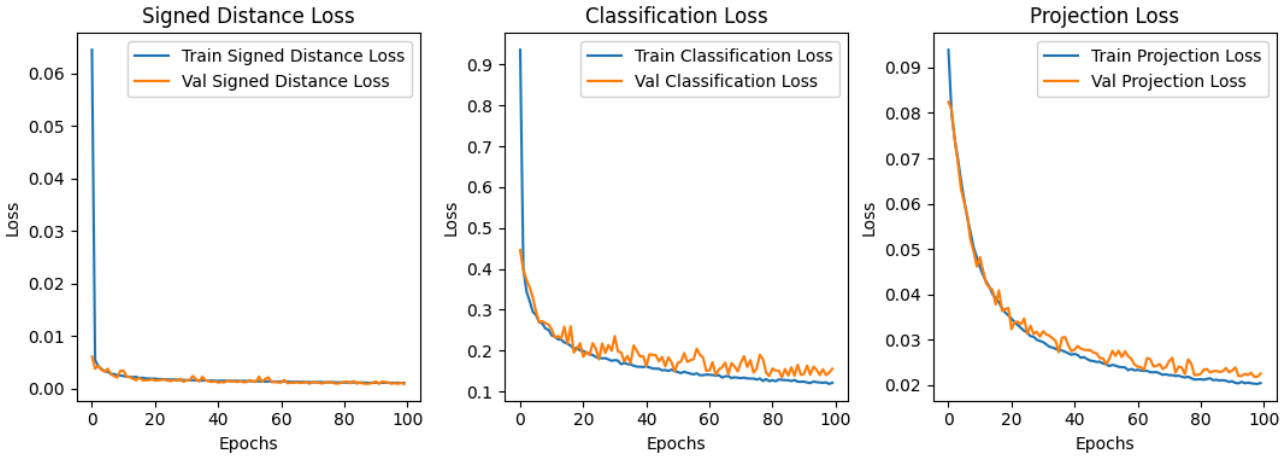


same but 10% of samples, 1000epochs...

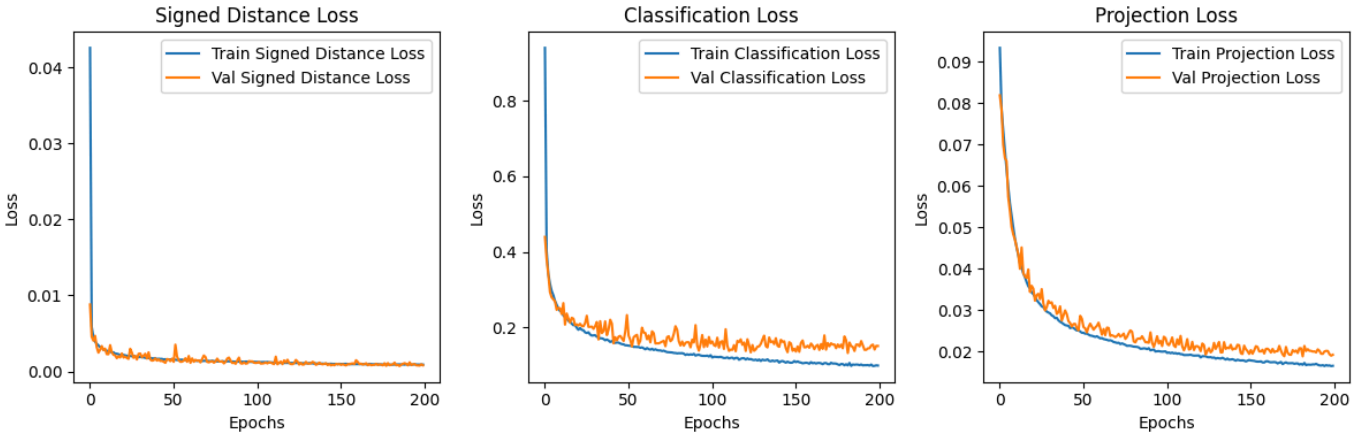


Calibration3_2

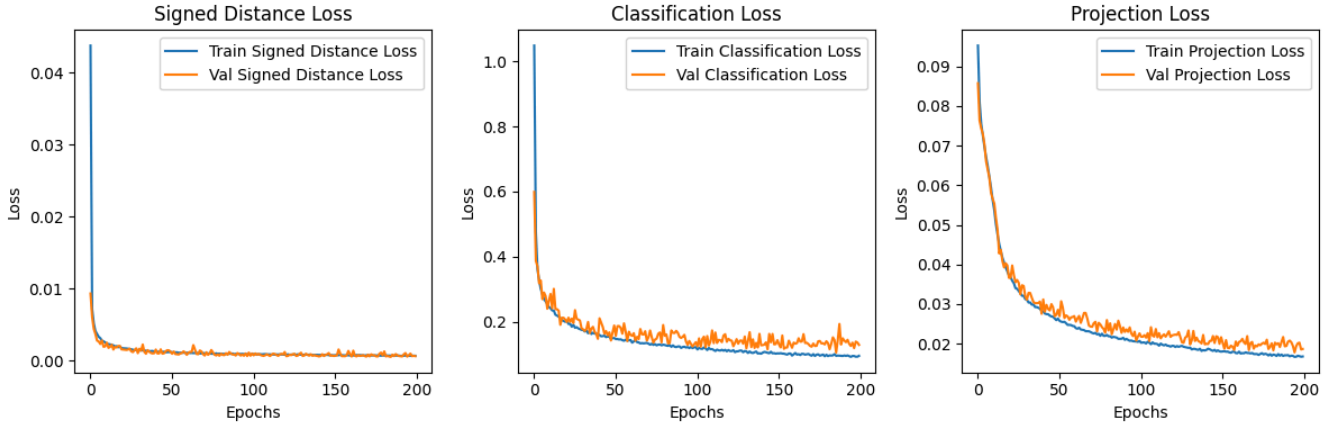




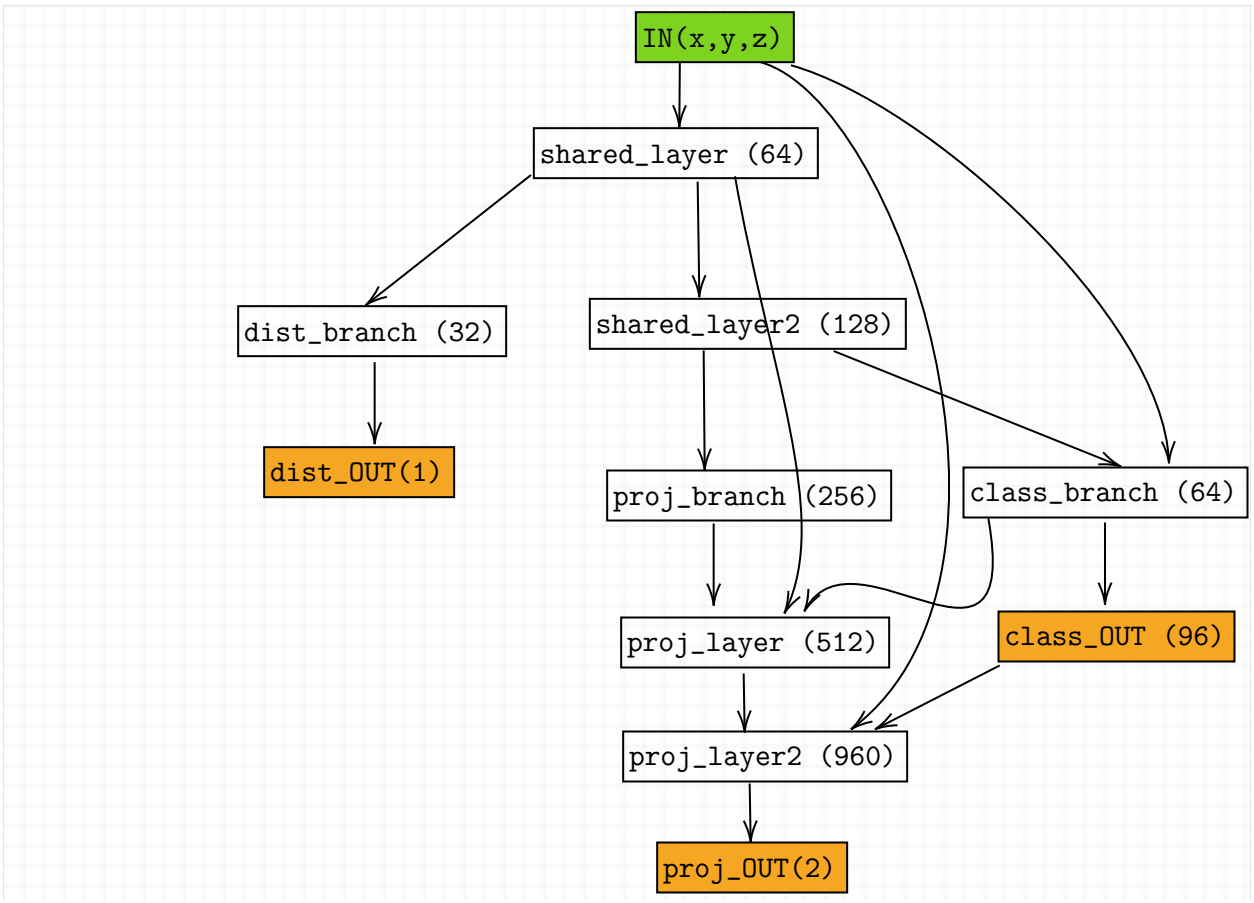
with 200 epochs...

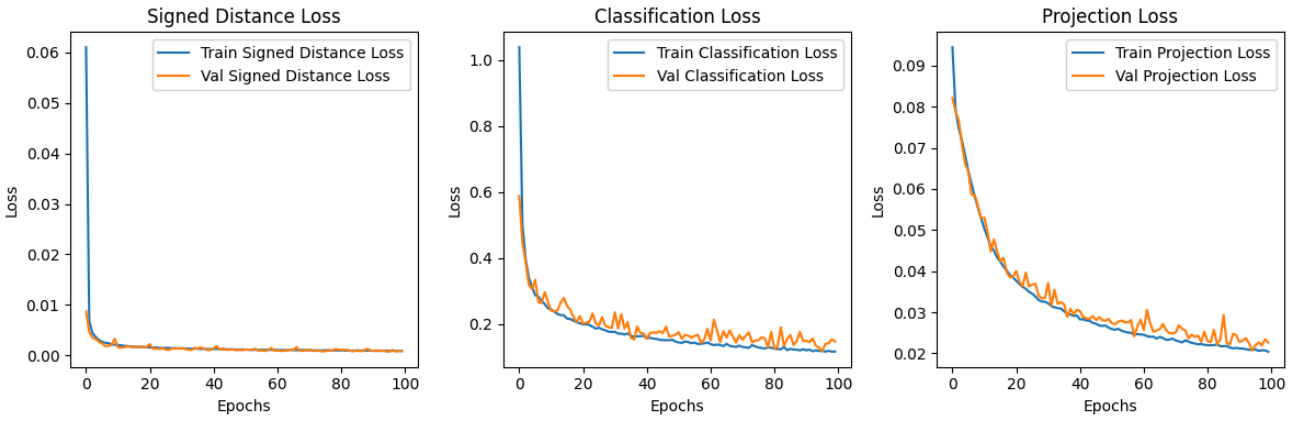


after pruning data (-0.5<gn<1.5)...



Calibration3_2_2 (increased units for regression, hoping to get selective use, when merging with classification output)





improves a lot the xi1,xi2 prediction. But close to the edge the accuracy ir **very** poor...

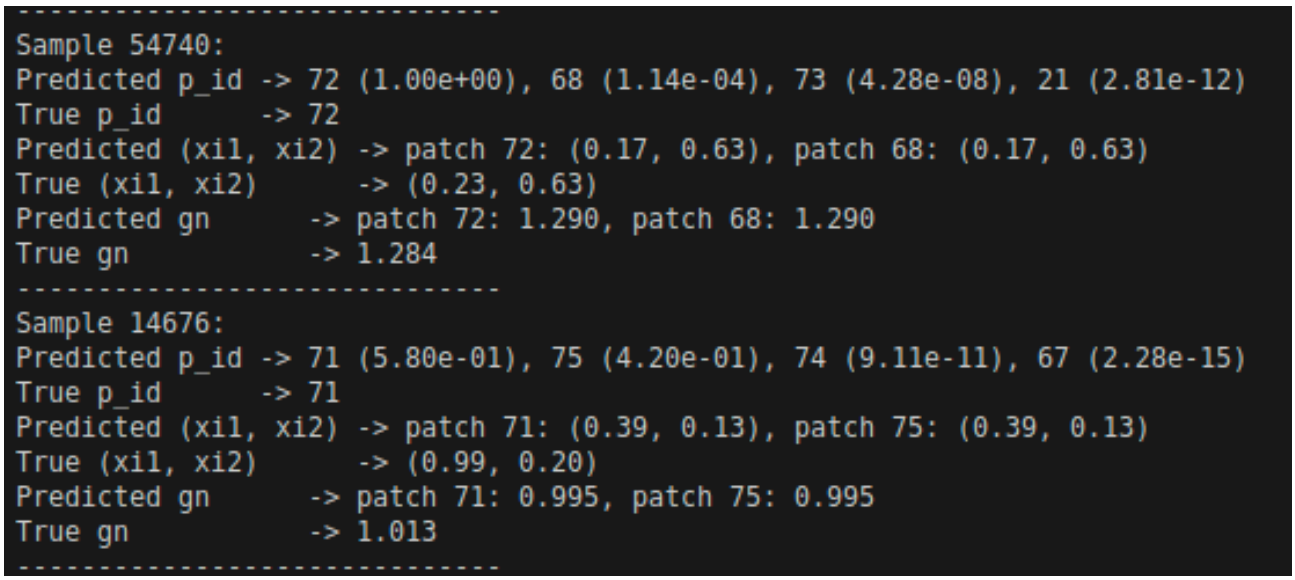
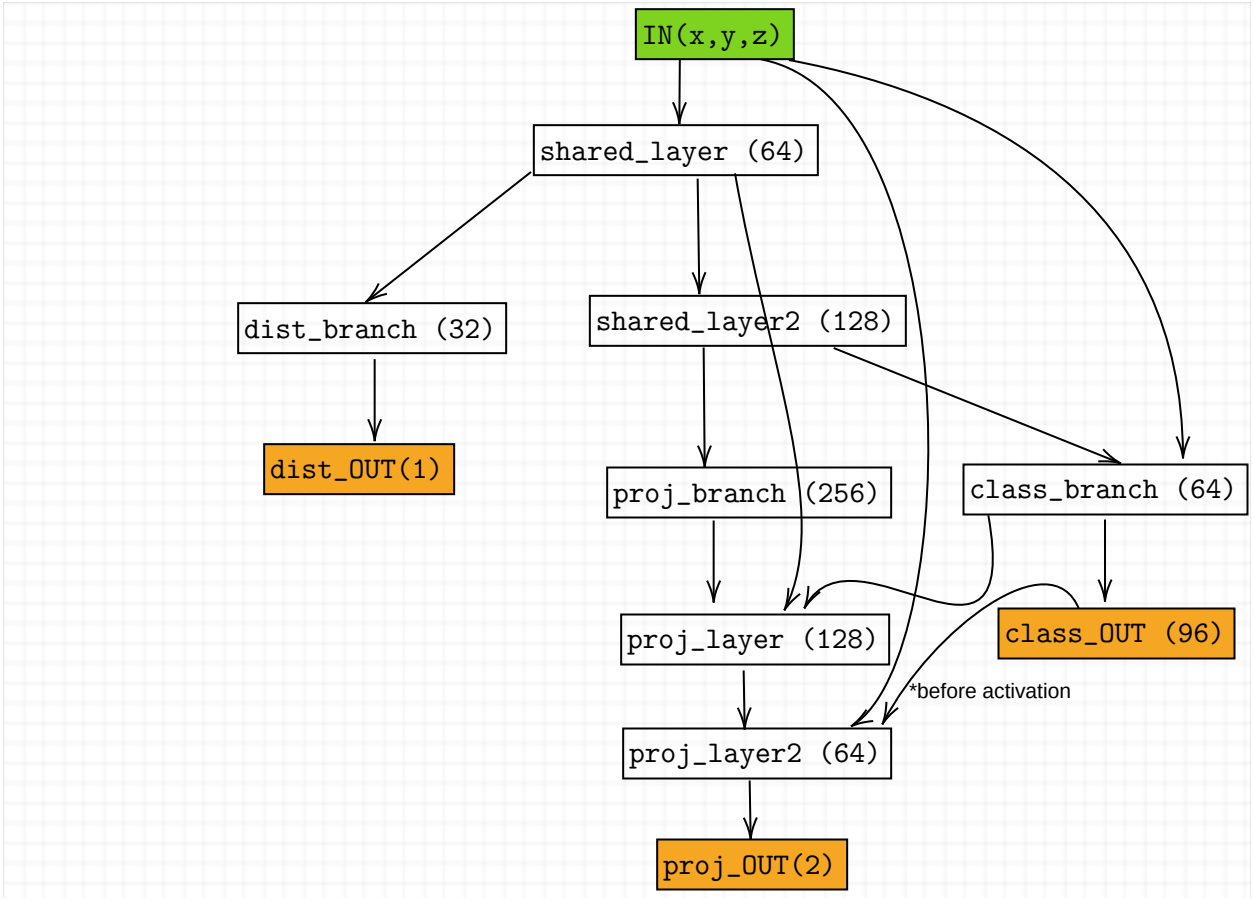
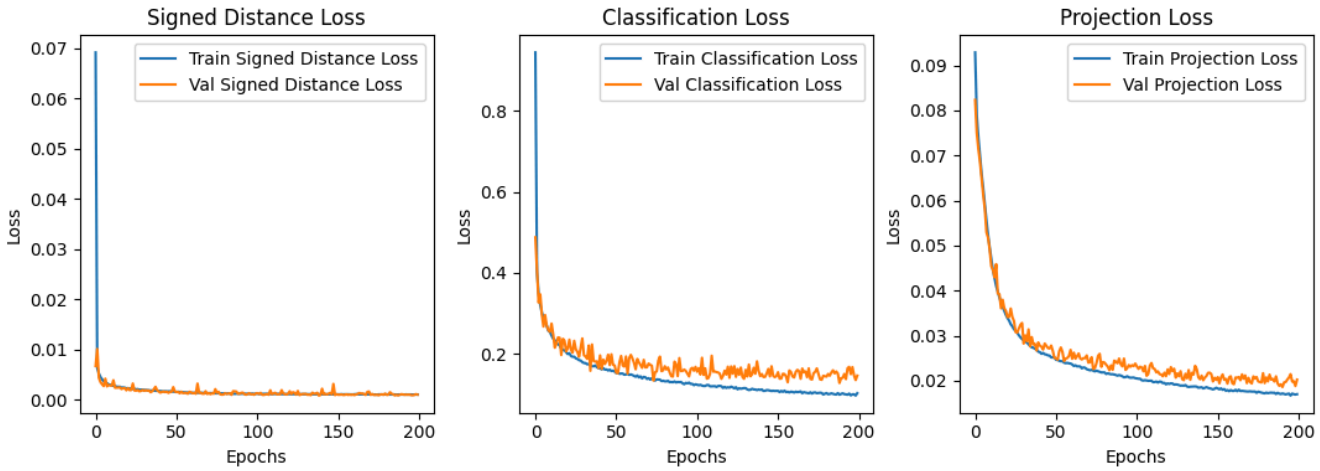


Figure 1: Sample 54740: Good projection. Sample 14676:Bad projection

Calibration3_3


IN(x,y,z)
shared_layer (64)
dist_branch (32)
shared_layer2 (128)
dist_OUT(1)
proj_branch (256)
class_branch (64)
proj_layer (128)
class_OUT (96)
*before activation
proj_layer2 (64)
proj_OUT(2)


Signed Distance Loss
Train Signed Distance Loss
Val Signed Distance Loss
Loss
Epochs
Classification Loss
Train Classification Loss
Val Classification Loss
Loss
Epochs
Projection Loss
Train Projection Loss
Val Projection Loss
Loss
Epochs