Loss Landscape Visualization

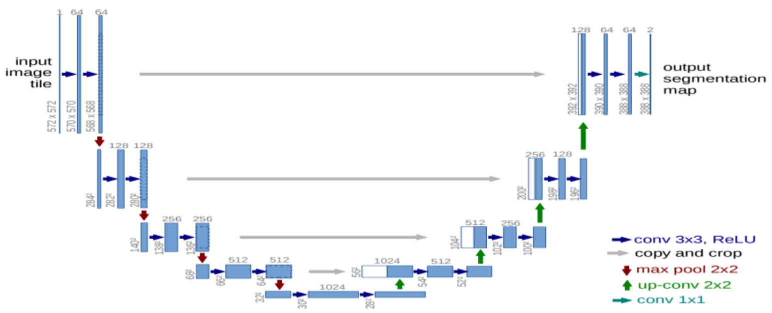
Arsh Upadhyaya Supervised By: Dr Shakes Chandra

INTRODUCTION

Motivation

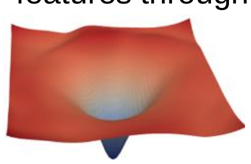
I was really fascinated by neural networks since my school days! But never really understood how exactly loss functions are evaluated during training stage. I am also very interested in U-Net model, it has a really cool structure, and major algorithm for biomedical segmentation. I felt being able to visualize loss landscape, which is a predominantly theoretical field for U-Net would be very interesting!

Background



U-Net Architecture[1]

The simple U-Net is able to transfer contextual features(through down sampling) and spatial features through skip connection concatenation



Loss Landscape[2]

$$\frac{1}{m}\sum_{i=1}^{m}\ell(x_i, y_i; \theta)$$

Cost Function[2]

During the training phase the objective is to find a set of parameters for which cost function is minimal across the entire training set.

In the context of the U-Net original images are the features, which are compared to original masks which are labels through some similarity based loss function.

METHADOLOGY

Initial Approach:

I wanted to visualize the individual parameters and their effect on loss through the training. Turned out to be too complex due to massive number of parameters existing in multiple dimensionalities

Effect of Certain conditions on Loss:

- Effect of multiple random initializations on loss
- Effect of multiple levels of data augmentation through added noise on loss

I ran into seemingly unexplainable divergent behavior for simple U-Net. After research identified scale invariance in neural networks as potential cause. I decided to create a second improved U-Net model with batch normalization and ReLU activation

Dataset Selection:

I decided to go with a skin lesion medical imaging dataset having 900 images and masks all of which were used in training

Original image Original mask

Loss Function

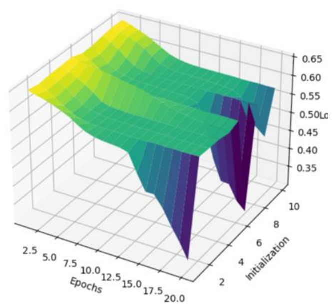
Since I was doing segmentation task, similarity based loss function like dice loss seemed great!

Optimizer Selection

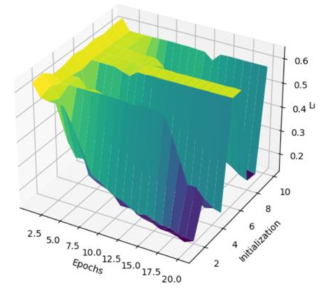
I felt for the purpose of loss visualization SGD would be a great optimizer due to it's greedy nature. For the results on this poster learning rate was set at 0.01and momentum at 0.9(when used)

RESULTS

Loss for random initializations in simple U-net

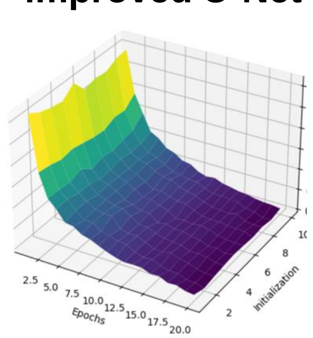


No momentum

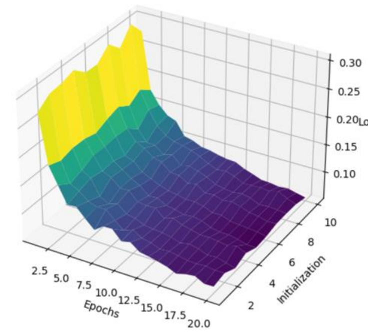


Added momentum

Loss for random initializations considering scale invariance in improved U-Net

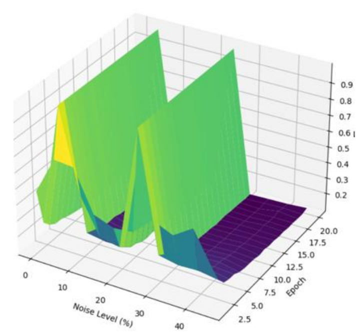


No Momentum

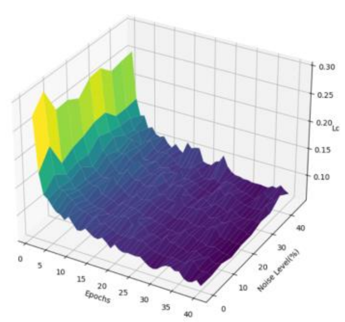


Added Momentum

Loss for different levels of noise without and with scale invariance



Simple U-Net



Improved U-Net

CONCLUSION

Major Learnings:

- Scale invariance is a very important consideration and batch normalization in improved U-Net gives normal behavior compared to inexplicable divergent nature of simple U-Net model in some cases
- Momentum addition to random initializations leads to instability to a greedy optimizer like SGD
- Higher levels of data augmentation through noise have more rough loss landscapes(expected behavior)

Future Directions:

- Rather than effect of conditions like random initialization and data augmentation, it would be great if I could analyze actual parameter based visualization of loss landscape
- Types of effect different skip connections have on loss landscape, since they are the identifying features of U-Net model

References:

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *arXiv pre-print server,* 2015-05-18 2015, doi: None arxiv:1505.04597.

[2] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the Loss Landscape of Neural Nets," *arXiv pre-print server,* 2018-11-07 2018, doi: None arxiv:1712.09913.

The University Of Queensland Australia

School of Information Technology & Electrical Engineering

INNOVATION EXPO