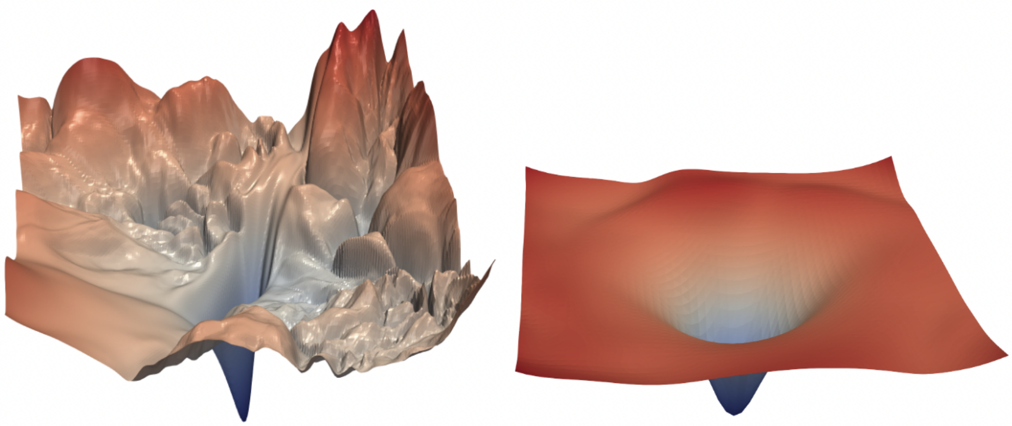
# Proposal: Visualizing the Loss Landscape of Neural Nets

Charles Ison*
Oregon State University

Matthew Morgan*
Oregon State University



(a) without skip connections (b) with skip connections

Figure 1: Example of two visualizations from the paper of interest: (a) the loss function of a neural network architecture without skip connections, and (b) with skip connections. We propose to implement these visualizations as the first step of our term project, then proceeding with creating other visualizations and enhancing them.

## Abstract

The abstract will be written as part of the final write-up...

**CR Categories:** I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems;

**Keywords:** rotational symmetry, field design, scalar field topology, surfaces, topology.

## 1 Introduction

For our term project, we propose to use the paper "Visualizing the Loss Landscape of Neural Nets" (`https://arxiv.org/pdf/1712.09913.pdf`) for Option 2 (implementing a published research paper). In this paper, the authors create scalar field visualizations of various neural network architectures that they call "Loss Landscapes." In these visualizations, the loss function's results serve as the scalar value and a two dimensional reduction of the model's weights serves the directional elements of the field. The specific loss functions used in the original paper are cross entropy (`https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html`) and mean squared error (`https://pytorch.org/docs/stable/generated/torch.nn.MSELoss.html`). The dimensionality reduction is done using two different approaches: principal component analysis and random vector selection. For the later approach, two random vectors within the input space are created by sampling a random Gaussian distribution. These vectors are used as the two dimensions of the scalar field input. The resulting loss landscapes generated can then provide insights into how certain architectural decisions can improve or worsen a network's trainability.

Just like the original authors, we will have to generate our own datasets by creating test models of multiple popular neural network architectures and recording their performance across a range of weights. The original authors tested ResNet and VGG architectures with various permutations, which we can recreate and also potentially extend to newer architectures such as a Vision Transformer (`https://arxiv.org/pdf/2010.11929v2.pdf`). To create our loss dataset, we will also use the CIFAR-10 dataset (`https://www.cs.toronto.edu/~kriz/cifar.html`) as input into our testing models like the original paper. The models will be created in Python using PyTorch and then our results exported to PLY files which we will use to generate corresponding visualizations in OpenGL.

To evaluate the correctness of our models we will use four separate criteria:

1. First, we will confirm that our results align with the original papers, which includes both quantitatively confirming our model's performance aligns with the expected results and visually confirming our loss landscapes align with those presented in the paper.
2. Second, we will extract all critical points from our field, classify them using a Hessian, and confirm they match the expected values based on the scalar field visualization.
3. Third, the original authors visualized their model's conver-

*e-mail: {isonc|morgamat}@eecs.oregonstate.edu

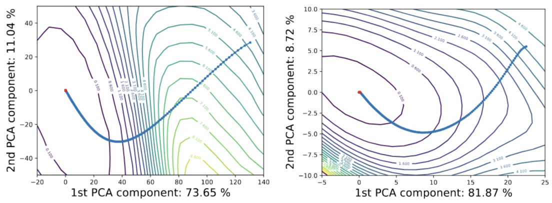



Figure 2: Plots from the original authors visualizing the model's convergence to a local minimum during gradient descent.

gence to a local minimum during gradient descent, as seen in Figure 2. We will compare our local minimum with theirs to verify our accuracy.

4. Finally, we will judge our visualization's overall usefulness by the ability to draw meaningful insights about neural network's architectures from the visualization. For example, the original authors were able to gain an intuition about the impact of increasing network depth and its impact on convexity. Our visualizations should be of a high enough quality to convey the same information.

## 2 Related Work

There have been a significant number of theoretical studies on the ability to optimize neural loss functions in order for neural networks to train faster and perform better against a generalized task. Less work has been conducted on the relationship between sharpness/flatness of local minima and their generalization ability. However, one paper by [Hochreiter and Schmidhuber 1997] defined "flatness" as the size of the connected region around the minimum where the training loss remains low which is visualized by this work.

Since the paper was released in 2018, numerous other works have cited it. However, many of the citations are not from works attempting to advance the visualizations, but rather to learn from the visualizations in their research on developing more generalizable and accurate models. The visualizations have come in handy for researchers studying reinforcement learning models such as [Bekci and Gümüş 2020] and [Plaat 2022].

There are some recent works which produce new visualizations of loss functions. One such by [Huang et al. 2020], "Understanding Generalization Through Visualizations", use visualization methods to make the mystery of neural network generalization more intuitive. They first visualize loss as a scalar field with height and color representing the output. But they also use a colored dot plot and a "Swissroll decision boundary" to show the difference in models that perform well versus models that perform poorly at generalizing.

An interesting extension of this work by [Linse et al. 2022] visualizes large neural networks in virtual reality. The approach allows for hgh interactivity, but requires a virtual reality headset and powerful computer to render.

## 3 Scalar Field Representation

In the final paper, we will display and discuss our results here…

### 3.1 Extracting Critical Points

To be written in the final report.

### 3.2 Tracing Stochastic Gradient Descent

To be written in the final report.

## 4 Applications

The predominant application of visualizing loss functions is for scientists to better understand how loss function geometry affects generalization in neural nets. We will continue this thought in the final report.

## 5 Conclusion

The conclusion will be written in the final report.

## Appendix

To be determined.

## Acknowledgements

To be determined.

## References

BEKCI, R. Y., AND GÜMÜŞ, M., 2020. Visualizing the loss landscape of actor critic methods with applications in inventory optimization.

HOCHREITER, S., AND SCHMIDHUBER, J. 1997. Flat minima. *Neural computation 9*, 1, 1–42.

HUANG, W. R., EMAM, Z., GOLDBLUM, M., FOWL, L., TERRY, J. K., HUANG, F., AND GOLDSTEIN, T. 2020. Understanding generalization through visualizations. In *Proceedings on "I Can't Believe It's Not Better!" at NeurIPS Workshops*, PMLR, J. Zosa Forde, F. Ruiz, M. F. Pradier, and A. Schein, Eds., vol. 137 of *Proceedings of Machine Learning Research*, 87–97.

LI, H., XU, Z., TAYLOR, G., STUDER, C., AND GOLDSTEIN, T. 2018. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc., S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31.

LINSE, C., ALSHAZLY, H., AND MARTINETZ, T. 2022. A walk in the black-box: 3d visualization of large neural networks in virtual reality. *Neural Computing and Applications*, 1–16.

PLAAT, A. 2022. Deep reinforcement learning. *arXiv preprint arXiv:2201.02135*.