

Interpolation of Light Source Direction on a Subject

EEE515 Project Proposal

Michael Liang Li

March 14, 2021

Introduction and Background

I propose using a neural network architecture to find the light transport interpolation between two or more angular lighting samples. In situations for capturing 4D reflectance fields, light stages are often used to gather this data of a subject. However, the light stages used for this are constrained by the size of the object, as well as the construction of the lighting stage required. To capture these fields, individual lighting from different directions is required, ideally covering the full sphere of illumination. To capture an accurate and detailed reflectance field dataset, a large number of these lighting sources is required. As an alternative, it may be possible to instead train a neural network architecture to learn the relationship between the lighting angle (θ, ϕ) , and an input image. From there, training the neural network on two or more images accompanied by its respective lighting angle, the network may learn the correlation between these two values. Then, given a novel lighting angle in the range of the input lighting angles, the model would be able to synthesize an image lit from that angle. This would alleviate some of the complexity of the light stage setup, allowing for similar levels of quality, while reducing the necessity of gathering as much initial data. This could also have additional application in other situations where multiple images of single source lighting is required. To approach this problem, a dataset with 4D radiance fields is required. One of the difficulties in synthesizing this novel lighting viewpoint is the consideration of the geometry that effects the light. For this light angle, finding an average of two lighting angles or summing them would not work due to a singular point having not as wide of a spread across the entire subject. In addition, depending on the geometry, some points of light will not be occluded in these scenarios. Therefore, some level of recognition of the geometry of the subject is required. In this dataset, a consistent viewpoint of a subject along with a changing lighting direction are needed. Two possible neural network architectures to achieve this goal include periodic activation functions for neural representations, and neural radiance fields.

Approach/Method

The Implicit Neural Representations with Periodic Activation Functions' model could be an avenue of synthesizing the novel lighting. Also known as Siren due to it's use of

sinusoidal representation networks, Siren works to map 2D pixel coordinates to an output color to parameterize images [1]. While the goal of this research is to synthesize a novel viewpoint, Siren mainly works in recreating already known viewpoints. However, due to the clarity of Siren’s image parameterization, and its ability to take in the pixel coordinates in addition to another coordinate, Siren could be a promising way for the network to interpolate between the secondary coordinate, in this case the lighting angle, to synthesize an image. Modifying the sinusoidal representation network to accept the lighting angle in addition to multiple image coordinates, and attempting to train it on these inputs could possibly result in the lighting synthesis. Some possible issues include trying to ensure that the network learns the correct correlation between the input image and the input theta, as since theta is only one parameter, it may not be weighted heavily enough in order to effect the output image. In addition, adding in multiple training images, whether randomized or in a pattern could cause problems with the learning of coordinate to color mapping as the initial image siren parameterization with an additional input ,may only parameterize a concatenated image, rather than taking in multiple separate inputs. This could cause the output to be blurry, due to a changing ground truth image over time.

Another possibility is the NeRF (neural radiance field) architecture [2]. While the implementation of NeRF focuses on volume rendering, it has also been shown to be applicable in relighting situations. One example of this is the Deep Reflectance Volumes paper [3], where the author utilizes NeRF’s ability to model a real scene, but uses it instead to relight the appearance under novel conditions. By comparison their method does not rely on viewing and lighting directions in order to compute their shading. Modification of the NeRF framework however could provide a good avenue for the model to learn and synthesize novel lighting angles, as the initial goal of NeRF is to learn the volumetric density representation from a photo collection. While their collection is unstructured, they also rely on camera parameters which provides a similar parameter to the lighting angle (θ, ϕ) . One possible consideration for the usage of NeRF is disentangling the rendering portion of the architecture from the Fourier features and the multilayer perceptron portions.

The initial step is to modify the networks to accept two input images along with their respective lighting angle, and test to see if the networks can reproduce the two input images with sufficient clarity. After modifying these networks to accept the correct parameters and training them, the results will need to be evaluated to discern what possible tuning would be required, dependent on the output of the network when passing in a lighting angle in the range between the training angles, and to compare the resultant output to the ground truth found in the dataset. Some possible avenues of fine-tuning the results would be modifying the number of hidden layers, changing the input dimensions, changing the network parameters, or modifying the network structure itself. The input dimensions of the network have many possible options for change, as both networks structure the input differently. For Siren, the input dimensions of the network directly correlate to the output dimensions. Two possible options for modifying the input dimensions in this case include either appending the lighting angle to every single 2D pixel coordinate, or adding a pixel coordinate specifically as a (θ, ϕ) value. In addition to this, other factors such as the

angle between the lighting in the images, the number of input images, and the structure of the network itself must be considered. If neither of the base models work out well for test results, another option would be a two branch neural network [4], to account for the minimum of two inputs in order to synthesize the viewpoint. If time permits, I will compare and contrast the outputs of both modified networks against a ground truth image from the dataset of a lighting angle in the range between the input lighting angles. This will provide a basis of comparison for how the performance of the neural network is. From there, testing the network on multiple different datasets to ensure consistency is necessary.

Datasets and Software

This problem will be implemented and tested using Python [5], initially through Google Colab for fast prototyping, then through Python scripts ran through Arizona State University’s Agave computing cluster [6]. While Python is not as fast as other programming languages, its higher overhead is mitigated by the wide variety of specialized machine learning libraries available for it. To keep consistency, PyTorch is the machine learning framework that will be used to implement all of the models. PyTorch is the native framework for the implementation of implicit neural representations with periodic activation functions, allowing for easier debugging in notebook form, whereas TensorFlow’s debugging system requires a much more difficult workflow [7]. For the short duration of this project, PyTorch has the ease of access. While the neural radiance fields implementation was written in TensorFlow, there are several models developed in PyTorch that replicate NeRF’s model structure. This will allow for both of the base models to be easily modifiable and developed from. The other portions of the Python processing will involve libraries such as the OpenCV [8] to assist with image processing and loading, as well as NumPy[9] as an accompaniment for OpenCV. In terms of the transitioning of usage from Google Colab to the Agave cluster, while Colab has the benefits of ease of access, it runs into memory limitations for processing larger images. To alleviate this problem, Agave is necessary.

For the dataset, the initial one used will be the light stage data provided by the ICT Vision & Graphics Laboratory from the University of Southern California [10]. This dataset provides several examples of 4D reflectance fields, along with the values necessary for gamma correction. This dataset fulfills the requirement of having a fixed camera angle, while varying the angle of the incoming light source. These images will need to be converted to radiometrically linear values before input into the model for training, as the provided images are already gamma-corrected. In addition, this dataset provides a good basis of comparison for the target image synthesis, as it has a large number of possible angle variations as inputs to test out. Due to the large number of sample points in the dataset, this gives a larger variety of possible light angle changes to sample from, and to compare against as a ground truth image, to see whether or not the model is learning and parameterizing the correct outputs. These subject images also have complex geometry which makes it easier to determine whether or not the network is learning a different operation of light for the image, rather than interpolating the viewing angle.

Timeline

March 31st - Literature Review

- Review at least 3 papers, assessing the strengths and weaknesses of each paper, and contextualizing it with this current project as well as other related projects. In addition, propose extensions or future ideas for these papers

April 2nd - Report v1 Rough Draft

- Introduction
- Background and Related Work - In this case, information on NERF, SIREN and their predecessors Methods/Approach - Explaining the model
- Experiments - information about the framework and datasets used

April 5-9th - Midterm Presentation

- Create a 10-15 minute presentation on the progress of the project, note critical feedback for possible revisions and leads to follow
- Introduction to the problem statement
- Note technologies used
- Describe current methods/approach to the project
- Detail machine learning models being explored as well as the dataset being used
- Slide show demonstrating visuals of the results so far
- Detail current plans for the rest of the semester

April 19-23rd - Report v2

- Revision of the rough draft of the report, improve sections based on feedback
- Inclusion of a preliminary results section

April 26-30th - Final Report

- Description of every member's contributions - review of tasks and progress over the project time period

References

- [1] V. Sitzmann, J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein, “Implicit neural representations with periodic activation functions,” 2020.
- [2] R. Martin-Brualla, N. Radwan, M. S. M. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, “Nerf in the wild: Neural radiance fields for unconstrained photo collections,” 2021.
- [3] S. Bi, Z. Xu, P. Srinivasan, B. Mildenhall, K. Sunkavalli, M. Hašan, Y. Hold-Geoffroy, D. Kriegman, and R. Ramamoorthi, “Neural reflectance fields for appearance acquisition,” 2020.
- [4] L. Wang, Y. Li, and S. Lazebnik, “Learning two-branch neural networks for image-text matching tasks,” *CoRR*, vol. abs/1704.03470, 2017.
- [5] “Python.” <https://www.python.org/>. Accessed: 2021-03-14.
- [6] “Research computing.” <https://asurc.atlassian.net/wiki/spaces/RC/overview>. Accessed: 2021-02-25.
- [7] “Tensorflow.” <https://www.tensorflow.org/>. Accessed: 2021-03-13.
- [8] “Opencv.” <https://opencv.org/>. Accessed: 2021-03-5.
- [9] “Numpy.” <https://numpy.org/>. Accessed: 2021-03-14.
- [10] “Light stage data gallery.” <https://vgl.ict.usc.edu/Data/LightStage/>. Accessed: 2021-01-10.