Muhammad Sohail, Toams Iser, Tim Weyrich

Generative Materials Project (Exercise 1)

BRDF Capture and Fitting.

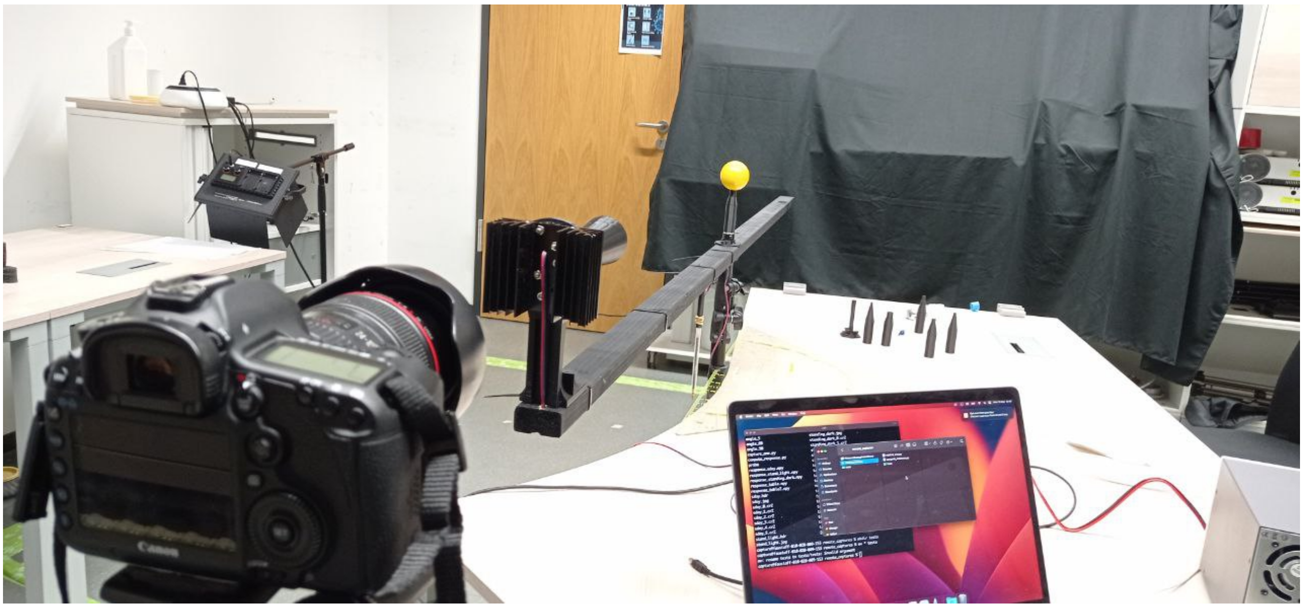
In this exercise you capture, fit, and render BRDFs in a controlled environment (with limited external factors). You will have the opportunity to analyze a material of your choice by using a capturing setup installed for this purpose. Guided by existing functions and tools, you will go through the entire pipeline: capture, pre-processing, fitting / inverse rendering, rendering, and evaluation. [2]

Please note that this exercise sheet only provide an overview of the assignments, and `wt_brdf.ipynb` is designed as a walk-through notebook with detailed instructions. This code framework is provided on StudOn.

On assignment submission you will be asked to show us your solution. Also, upload the completed jupyter notebooks to studon **beforehand**.

Assignment 1 [4 Points] (Data Acquisition)

The first task is to acquire the data. That is, you will capture a sphere that has a uniform, isotropic BRDF from several angles. The images need to attend to the high dynamic range of radiance values that will occur when observing a shiny material. To work efficiently with the data, we will detect the sphere in the images and make a tight crop around the area of interest as a preprocessing step. These crops will be used in the fitting step later in this exercise.



Capturing

For this assignment, you should use two different materials and process them with the rest of the pipeline. The images will be captured using the setup in the VR-Lab, employing the provided scripts, and following the relevant instructions.

You are free to come with your own spheres (uniform, isotropic, with a radius between 18 and 60 millimeters). Use a fitting support size to hold the sphere in position. If you are unable to come up with an adequate sphere, we provide a selection of spheres with various BRDFs that you can use.

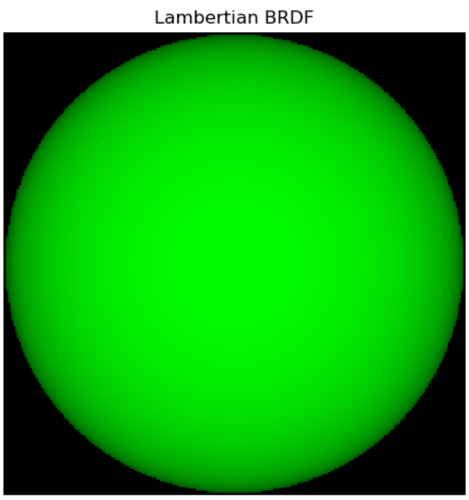
The result of the capture should be a folder `groupname_##/HDR` which contains an HDR image for each lighting angle. You should capture at least 10 images, uniformly covering the 5 to 175 degrees angular range. However, avoid the 0 angle, since for 0 the light source occludes the camera. The `groupname_##` tag is a simple identifier that you can freely choose, just make sure it is unique for each sphere for each group. After running the capture script for all the desired angles transfer the HDR folder to your personal machine.

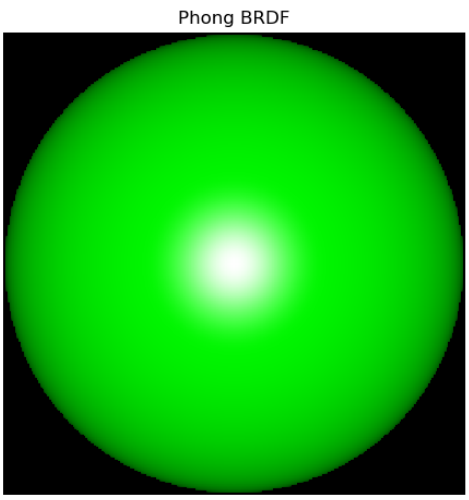
Processing

To obtain the sphere bounding box and crop the images use the provided `wt_process.ipynb` file. As most of the code is provided, you should just insert the information specific to your capture (sphere size, angles, etc.) and follow the instructions in the notebook.

In the end, you should have a set of equally-sized square images that are tightly cropped to contain the sphere.

Assignment 2 [6 Points] (BRDF Rendering)

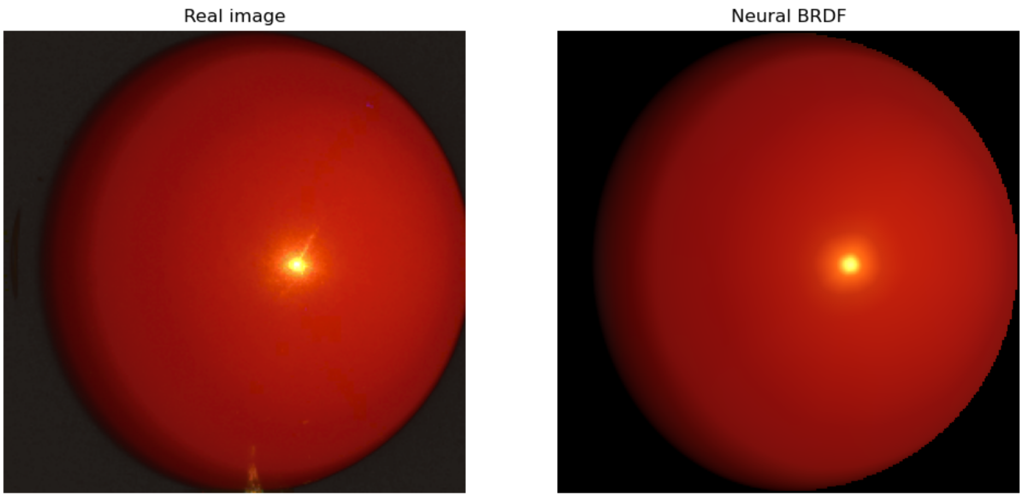




Using the data processed in the previous step, you will now follow the second guiding notebook: `wt_brdf.ipynb`. If you did not complete the first assignment, and you are willing to forfeit the points, you can use a default capture set provided by us as a backup for the rest of the assignments.

You will first write the code for simple, static BRDFs following the Lambertian and Phong models respectively. Then, you have to implement a simple rendering function that takes each point on the virtual sphere and computes the incoming light direction and the outgoing direction to the camera. We rotate the directions to a canonical system where the normal lies along the z-axis. The rendering routine then calls the BRDF with the computed directions to obtain the final output.

Assignment 3 [6 Points] (Fitting)



For this assignment, you should continue using the `wt_brdf.ipynb` notebook and follow the instructions for implementing BRDF fitting. We first load the data and using an analog to the rendering function we record the observed data as a mapping from light and view direction in the canonical system and the observed radiance. To fit the Phong model you can use a simple optimization procedure from scipy. In the second part, you should write a NeuralBRDF and can use the linked paper as a reference [2]. The model is a simple multi-layer perceptron defined in PyTorch, and the fitting is done here using gradient descent.

You should implement the Rusinkiewicz parametrization that can improve the neural BRDF fitting. For more information see the original paper about this change of variables from Rusinkiewicz [1]. You should also implement the log-based loss function (Equation 1 from [2]).

Optionally, you can also try different parametrizations to enforce isotropy and the Helmholtz reciprocity principle to observe the effect of these contraints.

Assignment 4 [4 Points] (Evaluation)

As the final assignment you should evaluate the quality of the BRDFs obtained from your captures, how faithful are the renderings to the real material?

You should compute the MSE and SSIM metrics, comparing the renderings to the original images. Compute the metrics for two models: Phong fitted BRDF, and the NeuralBRDF with Rusinkiewicz parametrization and logarithmic loss. Briefly discuss in free form your observations, what model is better? In your opinion do the metrics correlate well with your perception?

As described in the first assignment, each group should capture at least 2 different spheres. What can you say about the two materials? How do their BRDFs differ?

Have fun :)

References

[1] Szymon M Rusinkiewicz. A new change of variables for efficient brdf representation. *Rendering techniques*, 98:11–22, 1998.

[2] Alejandro Sztrajman, Gilles Rainer, Tobias Ritschel, and Tim Weyrich. Neural BRDF representation and importance sampling. *Computer Graphics Forum*, 40(6):332–346, September 2021.