ASSIGNMENT 13

In this assignment, we will load a textured mesh from an obj file, create a synthetic dataset by rendering the mesh from multiple viewpoints and use our multi-view dataset to infer properties of the scene with a differentiable renderer. First, we'll set up an optimization loop to fit a mesh to the observed dataset images based on a rendered silhouette loss; then we will augment this optimization loop with an additional loss based on rendered RGB images, aiming to predict both a mesh and its texture; and finally we'll learn to infer the camera position of a rendered image using the mesh geometry.

The goals of this practice are the following:

- Create a synthetic dataset by rendering a textured mesh from multiple viewpoints
- Understand more deeply a renderer pipeline
- Learn how a differentiable pipeline can be used to infer a variety of scene properties
- Train a model to infer 3D geometry using 2D images supervision
- Test and understand the limitations of a differentiable renderer

Instructions:

If you're using Google Colab, you just need to have a google account and an associated Google Drive. Make a copy of the notebook located below and modify it as requested.

In case you're choosing to work locally in your machine you must set Anaconda or a `venv` virtual environment, and install the necessary libraries. Create a folder in your Google Drive or in your machine's workspace. Copy to your drive folder or download the following notebook:

Assignment 13 Notebook 

1. Follow the instructions in the notebook for completing the assignment.
2. If you want, you can build auxiliary .py scripts and call them from your notebook, for organizational purposes.

Submission

The assignment is due on June 22nd, 2023 at 11:59am (GMT-3).

Students should send their assignments before the due date to hallpaz@impa.br with a copy to lvelho@impa.br. Late delivers will be consider subject to a lower score.

The submission email should be sent with the subject “Assignment 9 - [first-name] - [last-name]”. The assignment can be structured and sent in two ways:

If your whole solution is implemented in the same notebook as the one provided for the assignment, then you can send just the .ipynb file as the solution. If parts of your implementation were done in auxiliary .py scripts, then you must send both the final notebook and the scripts inside a .zip file. The organization of the code will also be considered in the evaluation.

References:

1. PyTorch3D: Getting Started with Renderer
2. Lab class on Differentiable Rendering
