Homework 1
Photometric Stereo

Due date: 23:59 Wednesday December 7th (2025)

For each object, we provide 16-bit integer PNG images with resolution of 612x512 from 96 different lighting directions. All the given images are linearlized, i.e. the pixel value is the radiance predicted by the reflectance model. The mask image (‘mask.png’), lighting directions (‘light_directions.txt’, 3x96, with each row as a unit 3D vector), lighting intensities (‘light_intensities.txt’, 3x96, with each row represents the intensities in RGB channels; images are required to be normalized by dividing these intensity values perchannel before performing photometric stereo), and image file names (‘filenames.txt’) are provided within the subfolder of each object.

We have included a main script named mainBaseline.m that takes care of reading in images from a directory, calling the photometric stereo function (that you will be implementing) and generates images showing the output and some of the intermediate steps. You are free to modify the script as you want, but note that your script mainBaseline.m should be executable for our evaluation. Please make sure your code runs correctly with the original script and generates the required output images.

Every script and function you write in this section should be included in the matlab/ directory. Please include resulting images in your write-up.

[40 marks]
Estimation of normal: The intensity of a pixel (i.e. the radiance of the corresponding surface point) is decided by reflectance models. For example, according to the Lambert’s model, $I = \rho \vec{n} \cdot \vec{l}$. Here, ρ is the surface albedo, $\vec{n}, \vec{l}$ are the surface normal and illumination direction. According to the photometric stereo algorithm we studied in the class, ρ and $\vec{n}$ can both be uniquely determined when at least three images are provided with known $\vec{l}$. Implement this algorithm and test your implementation with the provided data.

[60 marks]
Dealing with shadows and highlights: Shadows and highlights break the linear Lambert’s model. A simple solution to this problem is to sort all the observations at each pixel respectively, and discard certain percentage of the darkest and brightest pixels to get rid of shadow and highlight respective. After discarding those noisy

observations, the Lambertian photometric stereo algorithm can be applied on the remaining data. Implement this method and test it with the provided data.

2. Grading

You are required to submit both your report and source code (Matlab or C/C++ or Python) to Learning in ZJU (学在浙大). Your report should be in the **pdf** format with no more than 6 pages.

Your report should include **at least three** images for each example data: a) a normal map linearly encoded in RGB; (You can use the RGB three channels to represent the x,y,z three components of a normal direction. However, xyz vary from -1 to 1, while RGB are only between 0 and 1. So we can store (x+1)/2 in R, (y+1)/2 in G and (z+1)/2 in B.) b) an albedo (ρ) map; c) a re-rendered picture of the object with your recovered normal and albedo under illumination direction that is the same as the viewing direction.

In the report, you are expected to discuss your findings through the experiment. For example, what makes trouble to the implemented algorithm? What kind of data works best/worst? How the implemented algorithm can be improved?

Please zip everything together in **a single file**, and **name it with your student id** before uploading.