EE5175 : Image Signal Processing

Lab - 8 : Photometric Stereo

In this assignment, you will be implementing a basic version of uncalibrated Photometric Stereo algorithm using SVD. Photometric stereo is a method for reconstruction of the surface of an object from several images of the object in a fixed position but under different lighting conditions. The images under the unknown lighting directions (`Beethoven1.png. Beethoven2.png, Beethoven3.png`) are given along with a binary mask `mask.mat` or `mask.png` indicating the region of interest. The objective is to estimate the surface normal and the lighting directions using the observed images.

Consider k images each of size $m \times n$. Construct an image matrix I of size $mn \times k$, whose columns are lexicographically arranged images. According to the image formation model, we can write, $I = N.L$, where N is the normal matrix of size $mn \times 3$ and L is the lighting matrix of size $3 \times k$; i^{th} row of N contains the surface normal at i^{th} location and j^{th} column of L denotes the lighting direction of the j^{th} light source.

(a) Decompose the matrix I using SVD. If $k > 3$ approximate the SVD representation for rank 3 by preserving only the largest 3 singular values (To avoid out of memory issues while using SVD in Matlab, use 'econ' option. [U,S,V] = svd(I,'econ')). Now with rank 3 approximation, $I = USV^T$, where U is of size $mn \times 3$, V is of size $k \times 3$ and S is of size 3×3. If you represent $N = US^{\frac{1}{2}}$ and $N = S^{\frac{1}{2}}V^T$ There is always a nine-parameter ambiguity in correctly determining N and L. For any 3×3 invertible matrix A, $N = US^{\frac{1}{2}}A$ and $L = A^{-1}S^{\frac{1}{2}}V^T$ will satisfy $I = NL$. From the given the ambiguity matrix A (in `A.mat` or `A.txt`), find the correct N and L.

(b) Find the albedo of the object from the estimate normals. Albedo at a particular location is defined as the magnitude of the normal at that location.

(c) Generate a new set of images of the same object from a different set of lighting directions L_{new} provided in `Lnew.mat` or `Lnew.txt` using the estimated normals.

(d) **(Optional)** Use the given MATLAB function `direct_weighted_poisson.m` to integrate the estimated normals to find the best fit surface of the object. Plot the surface using `surf` or `surfl` command in MATLAB. (Hint: Represent the surface normal at each pixel location as the surface gradients p and q. At each pixel, $p = -n_x/(n_z+\epsilon)$ and $q = n_y/(n_z+\epsilon)$, where ϵ is a very small positive number used to avoid divide by zero error. Use the surface gradients p and q for integration on the domain defined by `mask.mat` or `mask.png`. Caution: In your estimate of N, the second dimension will corresponds to n_x.)

-end-