IPR Project Report

ARNAV JAIN

210186

1 Implementation Detail

Data Generation Steps

1. **Generate Surface Normals**: Generate random upward-facing, normalized surface normals.
2. **Generate Lighting Directions**: Generate random lighting directions to avoid shadows.
3. **Compute Intensity Measurements**: Simulate Lambertian reflection to compute light intensity due to varying light sources.
4. **Generate Reflectance Maps**: Create spatially varying or uniform reflectance maps for multiple wavelengths.
5. **Render Synthetic Images**: Combine surface normals, lighting, and reflectance to render synthetic images.

Code files and folders

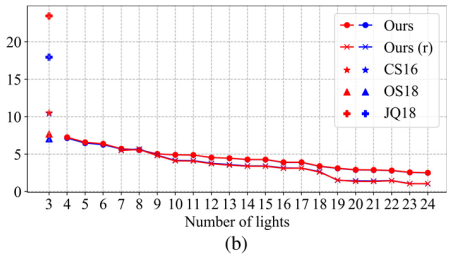
1. **MPS_SCPS.py and MPS_SCPS_robust.py:** Estimates surface normals and reflectance using normal and robust method respectively.
2. **generateLight.py:** Simulates and visualizes lighting in a 3D space.
3. **render.py:** Generates synthetic data based on Lambertian reflection models. Renders final image with the lighting and albedo conditions.
4. **eval.py:** Evaluates the accuracy of estimated surface normals compared to ground truth normals.
5. **albedo.py:** Generates spatially varying albedo.
6. **err_maps.py and testplot.py:** Testing files to output results.
7. **Folders - bunny and sphere:** These contain files to generate bunny and sphere synthetic images.
8. **Folder - lights:** Stores different lighting condition files.

2 Results

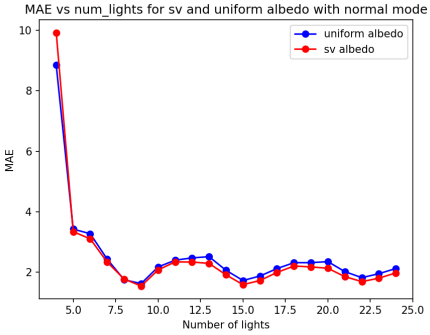
Issues with the existing code

1. The code involves generating random chromacities thus the results change on different runs. No particular seed value or any strategy of averaging over different runs has been provided to replicate results. I averaged over 100 runs to get close to paper results.
2. The code to generate spatially varying albedo was not provided (albedo.py). I wrote that myself using the albedo image provided in paper.

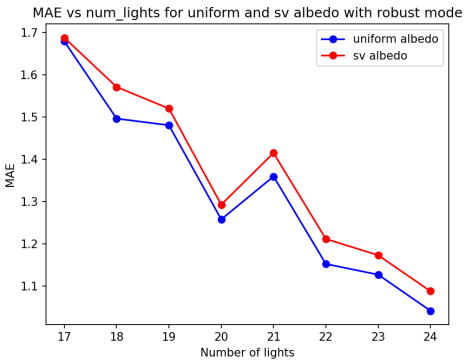
3. The parameters of lighting conditions (angle, isHalfRequired, lightzmargin) not given. In the lighting conditions which gave close results, robust mode only gave stable results for number of lights greater than 16.



Paper results for both norm and rob mode.



My implementation results for norm

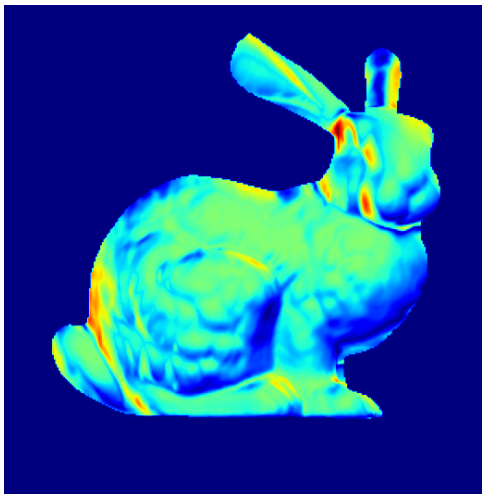


My implementation results for rob

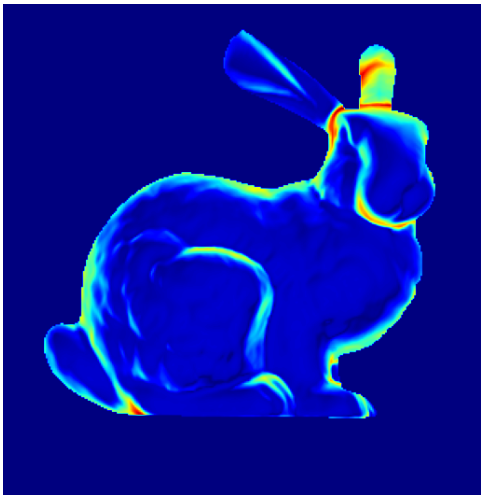
Graphs

Mode	num_lights	MAE
norm	4	9.91
norm	24	1.98
rob	24	1.08

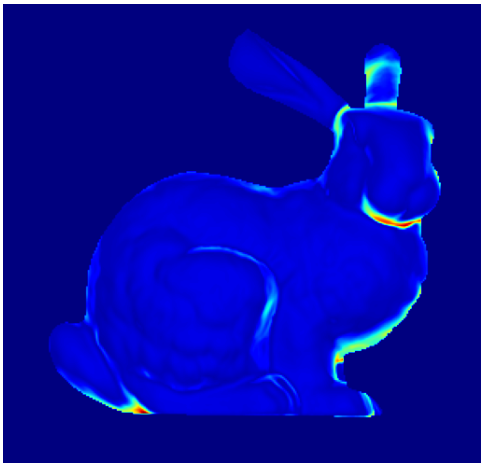
MAE Values on spatially varying albedos.



norm,4

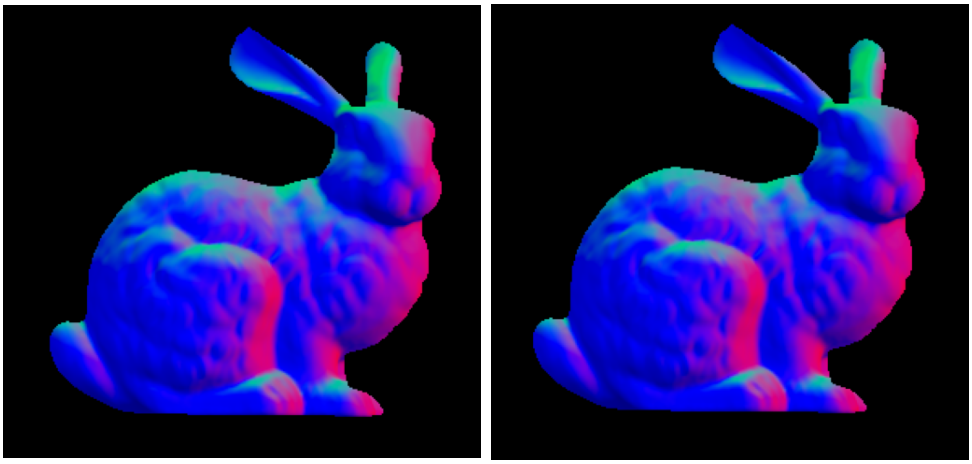


norm,24



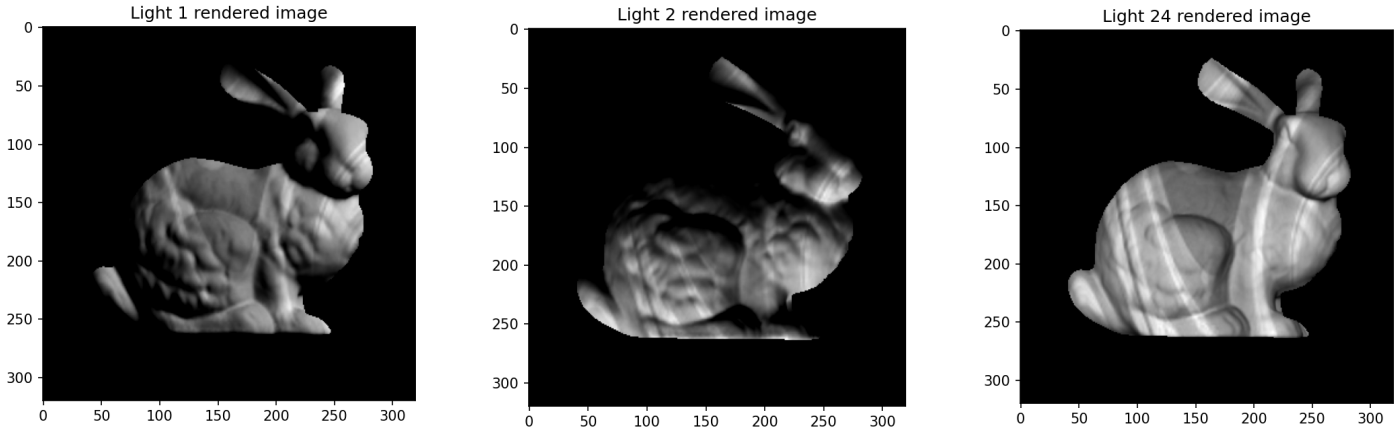
rob,24

Error Maps on sv albedo (Mode, num_lights)



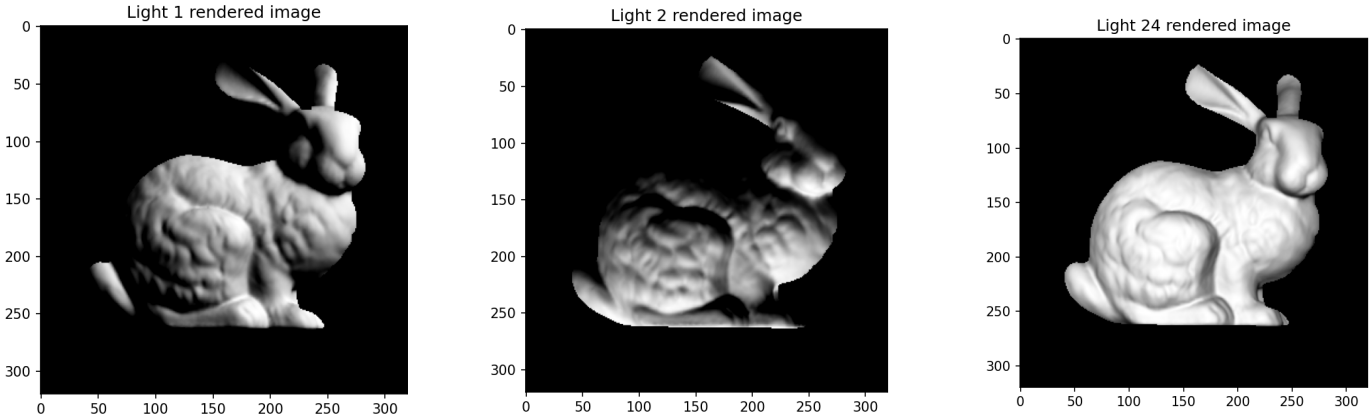
uniform albedo sv albedo

Normal Estimates (4 lights)



light 1 light 2 light 24

Rendered Images (spatially varying albedo)



light 1 light 2 light 24

Rendered Images (uniform albedo)

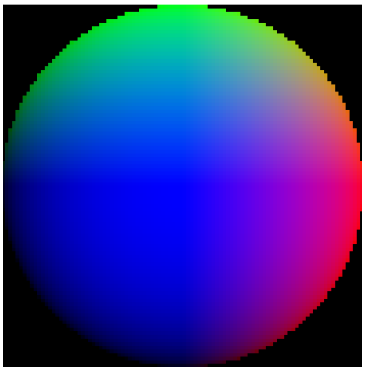
3 Dataset Description

Generates synthetic dataset. We generate two images - sphere and bunny. The normal.npy and mask.npy files are provided for both. To render final image we further provide albedo and lighting conditions. Shown below are the inputs to synthesize the images -

bunny mask

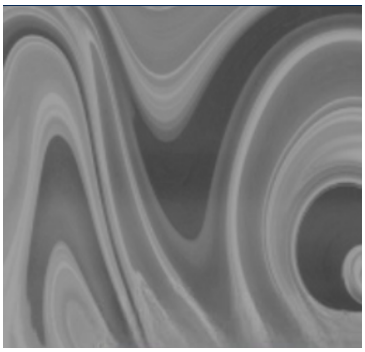
bunny normals

sphere mask

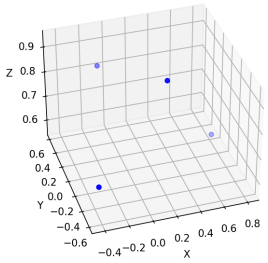


sphere normals

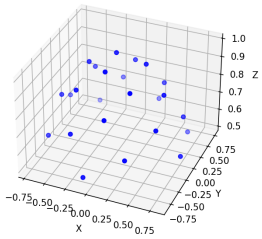
uniform albedo



spatially varying



4 lights positions



24 lights positions