Implementation of Photometric Stereo with General, Unknown Lighting

Aishwarya Agrawal
Department of Electrical Engineering
Indian Institute of Technology Gandhinagar
Ahmedabad, India
aishwarya@iitgn.ac.in

Yash Goyal
Department of Electrical Engineering
Indian Institute of Technology Gandhinagar
Ahmedabad, India
yash.goyal@iitgn.ac.in

***Abstract*—This report discusses the implementation of the shape recovery in case of four harmonics as described in the paper titled *Photometric Stereo with General, Unknown Lighting (Basri, Jacobs, Kemelmacher), 2006*. The goal is to recover the harmonic images from a set of number of images of an object taken from the same viewpoint but with different illumination. These harmonic images are then used to infer the surface normal and albedos of the object being imaged.**

Keywords—photometric stereo; singular value decomposition(SVD); eigenvalue decomposition; albedo; surface normal; Lorentz transformation

I. INTRODUCTION

The paper titled *Photometric Stereo with General, Unknown Lighting*[1] shows how to perform photometric stereo under quite general lighting conditions that need not be known ahead of time. Objects considered are convex that are approximately Lambertian. It is assumed that lighting in each image may include arbitrary combination of diffuse, point and extended sources with the only constraint that lights are relatively distant and isotropic(no cast shadows or slide projectors). The main idea behind the work is that for Lambertian objects general lighting conditions can be represented using low order spherical harmonics. The shape of the object being imaged is then recovered using optimization in low-dimensional space. Shape ambiguities that arise in such representation have also been analyzed. The reconstructed shapes have been compared with the shapes obtained against laser scanner. Photometric stereo with general, unknown lighting finds application when modeling large outdoor structures where it may not be practical to completely control the lighting; reconstructing shape using previously taken photographs; understanding how humans perceive shape under similar conditions.

In [1] it is shown that the 4D space of an object's images corresponds to its albedo and its surface normals scaled by the albedo. Hence, the observed images are approximated with a 4D space in which one dimension equals the norm of the other three by solving an overconstrained linear system. It is also shown that with 4D approximation, normals are determined upto a subgroup of 4 x 4 linear transformations, called Lorentz transformations. This report discusses the algorithm and implementation of the above.

II. ALGORITHM AND IMPLEMENTATION

M denotes the matrix consisting of the images of the object being imaged from the same viewpoint, but with different illumination. M is f x n where f denotes the number of images and n denotes the number of pixels in each image. Then, M can be approximated by linear combination of harmonic images that is,

$$M \approx LS$$

where L (f x r) contains the low order coefficients of lighting and S (r x n) contains the harmonic images[1]. In the case of 4D approximation, $r = 4$. The goal is to recover the harmonic images S and thereafter infer albedos and surface normals from the harmonic images for each pixel. The algorithm and implementation for the same is given below. The program has been coded in C++ using Microsoft Visual Studio and OpenCV library.

1) Read the images into the matrix M where each row of M corresponds to the pixels in each image. For the implementation, the image dataset is taken from [2]. The images are sized down before reading to suit the memory availability of the system.

2) Decompose M using SVD as $M = U\Delta V^T$.

3) Factored M as $M = L'S'$, where $L' = U\sqrt{\Delta(f4)}$ $S = \sqrt{\Delta(4n)}V^T$. $\Delta(f4)$ and $(\Delta(4n))$ denote first 4 columns (and first 4 rows respectively) of Δ. $\sqrt{\Delta}$ denotes the non-negative square root of the components of Δ.

4) Normalize *S'* by diving each element of a row by the row's norm. This is done to ensure that all rows have equal norms.

5) Construct a matrix *Q* of dimension *n* x 10. Each row of *Q* is constructed with quadratic terms computed from a column of *S'*. For each column $q = (q_1, q_2, q_3, q_4)$ of *S'*, the corresponding row in *Q* is $(q_1^2, q_2^2, q_3^2, q_4^2, 2\,q_1\,q_2, 2q_1\,q_3, 2q_1\,q_4, 2q_2\,q_3, 2q_2\,q_4, 2q_3\,q_4,)$.

6) Construct *b* using SVD of *Q* (to solve $Qb = 0$). Construct *B'* from elements of *b* where *b* is a 10 dimensional vector (b_{11}, b_{22}, b_{33}, b_{44}, b_{12}, b_{13}, b_{14}, b_{23}, b_{24}, b_{34}). *B'* approximates the null space of *Q*.

7) Construct *A'* in the following way:

   a) If *B'* has exactly one positive eigenvalue, reverse its sign. ($B' \leftarrow -B'$).

   b) Then, *B'* has exactly one negative eigenvalue. Apply an eigenvalue decomposition $B' = WJΛW^T$ and compute $A' = \sqrt{Λ}W^T$.

   c) Otherwise, using gradient descent find *A'* that minimizes $\| \pm B' - A'^TJA' \|$

8) Compute $S = A'S'$.

9) Construct a matrix ρ of dimension of the original images from the 1st row of the matrix *S*. ρ is the recovered albedo matrix that gives albedo at each pixel of the original image.

10) Get n_x (x-component of surface normal), n_y (y-component of surface normal), n_z (z-component of surface normal) for each pixel from second, third and fourth rows of S respectively. (eg: value at (2,5) location of *S* is n_x for the fifth pixel of the original image).

III. RESULTS AND DISCUSSION

In this project, the algorithm written above was tested for four sets of images namely, elephant (fig.1), dinosaur (fig.2), hippopotamus (fig.3) and bear (fig.4). Each figure below shows one of several images used for reconstruction in gray scale (in left) and the recovered albedo (in right). It can be observed that the recovered albedo for each sample closely resembles the object being reconstructed.

The computation time taken by the program for a set of (7-14) images of size 80 x 60 was almost 1 hour. This time increases significantly with increase in number of pixels in the image. The program consumes time in computing SVD.

fig.1: Elephant- original image (left), albedo (right)

fig.2: Dinosaur- original image (left), albedo (right)

fig.3: Hippopotamus- original image (left), albedo (right)

fig.4: Bear- original image (left), albedo (right)

IV. CONCLUSION

In this project, we could recover albedos and surface normals successfully. However, the implementation of SVD needs to be improved to reduce computation time. We have used OpenCV built-in function SVD for computation of SVD.

V. FUTURE WORK

We have implemented the paper [1] till recovering the albedos and surface normals. A further step is to reconstruct surfaces form surface normals by solving overconstrained system of equations (24) and (25) in [1].

In addition to 4D approximation, [1] discusses 9D approximation as well. 9D approximation differs from 4D approximation in the sense that in 9D approximation the value of $r = 9$ instead of 4. 9D approximation makes weaker assumption that the first order harmonics lie somewhere in the 9D space spanned by nine principal components. Hence the scaled normals lying in the 9D space can be computed and harmonic images spanning a similar space can be generated.

Acknowledgment

We would like to thank Prof. Shanmuganathan Raman under whose guidance we carried this project. He helped us from time to time with the theoretical aspects of photometric stereo and technical aspects of programming.

References

[1] R. Basri, D. Jacobs, and I. Kemelmacher, “Photometric Stereo with General, Unknown Lighting,” International Journal of Computer Vision, 2006

[2] N. Alldrin, T. Zickler, and D. Kriegman, “Photometric Stereo with Non-Parametric and Spatially Varying Reflectance,” 2008 Conf. on Comp. Vision and Pattern Recognition (CVPR), Anchorage, AK, June 2008.

[3] OpenCV online documenttaion: http://opencv.org/