Photometric Stereo via Computer Screen Lighting for Real-time Surface Reconstruction

Grant Schindler
Georgia Institute of Technology
College of Computing
Atlanta, GA, USA
schindler@cc.gatech.edu

Abstract

We introduce a method which uses the light emitted by a computer screen to illuminate an object such as a human face from multiple directions, simultaneously capturing images with a webcam in order to perform photometric stereo. Dominant eigenvectors of the captured images provide surface normals, which are integrated into a 3D surface using Gauss-Seidel relaxation. The system runs at 10 frames per second on a consumer laptop computer.

1 Introduction

Despite the existence of numerous methods for capturing 3D models, some of which require only a projector and a camera, most home computer users lack the necessary hardware setup to make such methods widely practical for anyone but professionals. We introduce a method which uses the light emitted by a computer screen to illuminate an object such as a human face from multiple directions, simultaneously capturing images with a webcam in order to perform photometric stereo. The significance of this method is that it relies only on widely adopted consumer hardware, a computer screen and a webcam, both of which are standard components sold with millions of laptop and desktop computer systems in the past few years. Thus, with software alone, we can bring 3D model acquisition systems into homes across the world, leading to previously unforeseen uses of the technology. Furthermore, the method works in real-time, enabling applications like 3D video conferencing which previously would have required specialized hardware.

The two primary contributions of this work are: (1) a software-based photometric stereo solution that exploits commodity hardware already present in millions of homes across the world and (2) a real-time photometric stereo method for capturing and displaying 3D models in real-time.

Figure 1. Using only the light from a computer screen to illuminate a face from multiple directions (top), we use photometric stereo techniques to recover a textured 3D surface (bottom) in real-time from live video. This technique is viable on any laptop or desktop computer with a screen and webcam.

1.1 Related Work

Photometric stereo [18] is a shape-from-shading technique first introduced by Woodham [18] to estimate surface curvature from multiple images of an object under varying illumination. It was later realized that images under varying illumination lie in a low-dimensional subspace [17][10],

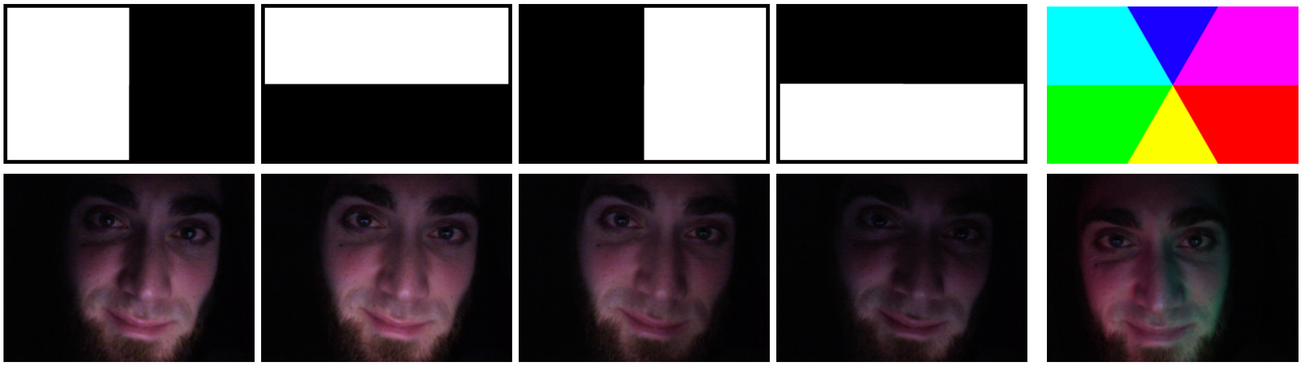

Figure 2. Here we see binary lighting patterns for $N = 4$ images (left) and a multispectral lighting pattern for $N = 3$ images (right). Note that in the standard case on the left, the lighting patterns are separated in time, while in the multispectral case on the right, the patterns are separated in wavelength, allowing three images to be captured simultaneously.

and that therefore surface normals could be extracted directly from input images via Singular Value Decomposition (SVD) [13][9][20]. These surface normals can be used not only to relight the scene [2] but also to reconstruct the 3D surface itself [1].

Various lighting hardware setups have been proposed for photometric stereo systems. Multispectral photometric stereo [19][14] systems have been used to simultaneously capture multiple illuminations from differently colored lights. In addition, real-time photometric stereo systems [15] running on a GPU have been used for inspecting objects using specialized multi-head lamps. Large outfitted environments like the light stage [8] have also been used to capture extremely high quality reflectance information for a human or other object sitting at the center of a sphere of programmable lights. A number of techniques use projectors to illuminate a scene in a structured light [16][3][21] scenario. In a structured light setup, dense 3D models are acquired by projecting known high-resolution light patterns onto the surface an object and observing the surface with one or more cameras. In [3] a simple structured light setup involving a lamp and a shadow-casting pencil was used for 3D model acquisition on the desktop.

Several recent papers [12, 6] explore the idea of using computer screen illumination to perform photometric stereo. In [6], a theoretical justification is provided for using nearby rectangular illuminants rather than distant point light sources for photometric stereo. In [12], photometric stereo is performed in an enclosed apparatus using a computer screen as a light source. Compared to our work, [12] uses a different strategy for activating screen pixels which necessitates long camera exposure times of 10 to 30 seconds per image. In addition, the environment matting work of [4] uses video of an object in front of a computer screen to model the light-transport paths through the object.

Finally, the idea of using a computer screen as a light source has several precedents outside of the computer vision and graphics literature. In the field of biochemistry, Filippini et al. [11] developed the computer screen photo-assisted technique (CSPT) which treats a computer screen as a programmable light source in determining the visible absorption spectra of samples placed in front of the screen. Commercially, the Photo Booth software bundled with Apple brand computers [7] simulates a camera flash by briefly turning all pixels on the computer screen completely white when capturing static images.

2 Photometric Stereo Approach

Photometric stereo is a shape-from-shading technique which takes as input a number of images of a static scene illuminated from various directions by a point light source. Assuming a lambertian reflectance function, the varying responses of individual pixels across the multiple images enables the recovery of the surface normal at each pixel. These surface normals are then integrated to recover a 3D surface, represented as a depth map or a height field. We discuss each of these steps in detail below.

2.1 Computer Screen Lighting

We capture N images $\{I_j : j = 1..N\}$ of an object, each under a different set of lighting conditions L_j. The lighting conditions L_j are defined by the set of illuminated pixels on the computer screen as follows:

$$L_j(x,y) = sign(x \cdot \cos(\frac{2\pi j}{N}) + y \cdot \sin(\frac{2\pi j}{N}))$$

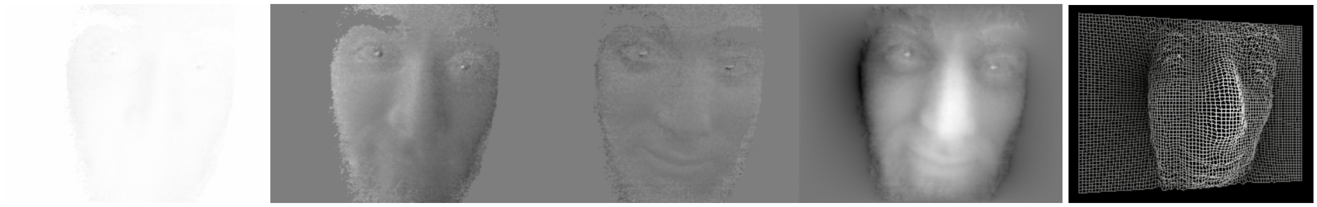

Figure 3. Surface Normals and Depth Map. Above we see, from left to right, the z, x, and y components of the surface normal vectors, the depth map that results from Gauss-Seidel relaxation, and a perspective view of a uniform mesh with vertices displaced according to the depth map. The four input images from which these results were computed can be seen in Figure 2.

where values -1 and 1 map to black and white pixels respectively and coordinate $(0,0)$ is at the center of the computer screen. This formulation leads to a rotating set of binary images as depicted in Figure 2.

2.2 Recovering Surface Normals

Following a standard approach [2], we treat each image I_j (of width W and height H) as a 1-dimensional column vector of intensity values of size $WH \times 1$ and place these N column vectors side by side into a matrix A of size $WH \times N$. It is well established [20] that the images live in a 3-dimensional subspace defined by the eigenvectors corresponding to the three largest eigenvalues of the matrix AA^T. These eigenvectors, which can be extracted using SVD, correspond to the z,x, and y components of the surface normal at every pixel in the image (see Figure 3).

2.3 Surface Reconstruction

Though there exist numerous methods of reconstructing a depth map from surface normals [1], we here adopt a coarse-to-fine Gauss-Seidel relaxation approach similar to [14]. In this incremental approach we update the depth estimate at each pixel based on measured surface normals and previously estimated values for the pixel's neighbors in the depth map. Starting from an initially flat depth map, this approach converges in less than 20 iterations (see Figure 3). We show in the next section why this method is even more beneficial in a real-time setting using an incrementally updated depth map.

3 Real-time Photometric Stereo

The primary obstacles to real-time photometric stereo are the time it takes to capture images under varying lighting conditions and the computational costs associated with recovering surface normals from images, and depth from surface normals. We address all three of these issues by adopting an approach that lets us compute continuous incremental updates to the model as we go, rather than recomputing the entire model at each step. We address each of these issues in turn.

3.1 Real-time Image Capture

We do not want to wait for a new set of N images to be captured before updating the 3D model during real-time photometric stereo. Instead we capture images continuously using the following scheme: At time t, we capture image I_j using light pattern L_j where $j = t \textit{ modulo } N$. Thus, at any time, we always have the most recently captured N images in memory, with each successive image I_j overwriting the previously captured image in position j. Every time we get a single new image, we can recompute the 3D model as described below. Another solution to the capture-speed problem is to use a multi-spectral approach, which we discuss in Section 3.4.

3.2 Real-time Surface Normals

It is well known that the expensive SVD computation of the $WH \times WH$ matrix AA^T can be avoided by instead computing the SVD of the much smaller $N \times N$ matrix A^TA. However, the construction of A^TA itself is computationally expensive as it involves taking the dot product of every image with every other image – an $O(WHN^2)$ operation. We make two key observations. First, in our incremental capture scheme, only one of the N images is updated at each timestep, meaning that only $2N-1$ of the entries need to be updated. Furthermore, the matrix A^TA is of course symmetric, so that of its N^2 entries, only N entries must be updated every time a new image comes in – an $O(WHN)$ operation.

Under an assumption of small motion, the resulting surface normals will correctly describe the surface under observation. However, even in the presence of temporary large motion which violates this assumption, the model will have caught up with reality in a maximum of N frames when a

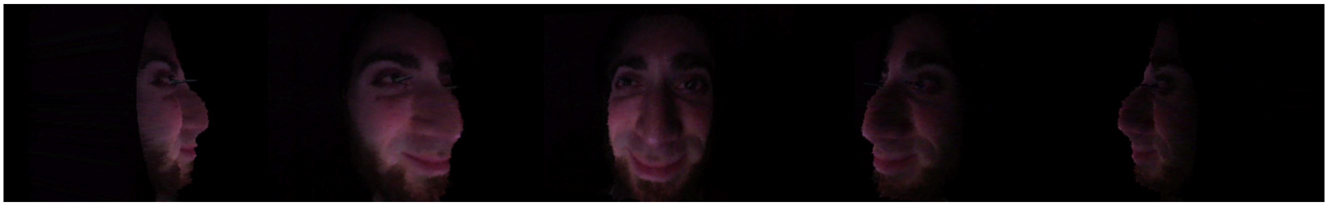

Figure 4. A recovered 3D model rotated through 180 degrees. The geometry of this model was recovered from the four images shown in Figure 1, while the texture was taken directly from only the first of the four images. Note that the virtual camera is placed close to the model to emphasize depth, leading to the extreme perspective distortion in these images.

completely new set of images has been captured. Thus, the normal operation of a system which does not use our incremental method becomes the worst-case operation of our method.

3.3 Real-time Surface Reconstruction

As noted above, to compute depth from surface normals, we use Gauss-Seidel relaxation. Normally, we begin this incremental approach from a completely flat estimate of the depth map. However, a further benefit of this approach in our incremental setting is that we can use the final depth map recovered from the previous relaxation as the initial state for the current relaxation procedure. This not only leads to faster convergence of the depth map for objects under small motion, but if the subject remains stationary, the depth map should converge to more and more accurate values over time.

3.4 Multi-Spectral Illumination, Ambient Illumination, and Extensions

There are a number of extensions to photometric stereo which our system can take advantage of. As noted above, multispectral photometric stereo [19][14] systems can be used to simultaneously capture multiple illuminations from differently colored lights. Because the pixels on a computer screen emit light in red, green, and blue channels independently, we can simultaneously display all lighting patterns for $N = 3$ at different wavelengths and therefore capture all images simultaneously with a color camera (see Figure 2). We note that the models resulting from normal $N = 4$ illumination are qualitatively more acceptable than those acquired under multispectral conditions, mainly due to a stronger resulting y component of the surface normals as a result of the additional constraints provided by the fourth image under different illumination conditions.

As noted in [5], constant ambient illumination can be removed from images by background subtraction. Because our scene is not static, a single background image will not suffice. We also don't wish to waste time capturing an unlit background image during every cycle of N images. Instead, we observe that every pixel should be illuminated by the computer screen in some images and not in others. If we consider the minimum intensity value of each pixel across the N most recent images to be due to ambient lighting, we can subtract it from all N images and the remaining intensity values should be attributable solely to the light emitted by the computer screen. In this way, we can capture 3D models even in the presence of ambient illumination.

Finally, we can trivially perform relighting on the 3D models we capture as described in [2]. In the above procedure, we already must normalize the x, y, and z components of the surface normals. The albedo at each pixel is just the unnormalized length of the surface normals. Combining albedo, surface normals, and a synthetic light source, the captured object may be illuminated in arbitrary fashion.

4 Results

We implemented the approach described above in Java on a 2.33 GHz MacBook Pro and achieved framerates of 10 Hz. Captured images are 320×240 pixels, and all subsequent operations are performed at this resolution.

We compute surface normals and depth maps as described above and texture the resulting 3D model with one of the original images of the object. An example of a resulting textured model can be seen in Figure 4, which shows the model from multiple view-points at a single instant of time during a real-time capture with the subject undergoing small motion. In addition, Figure 5 shows an assortment of images captured live by a webcam and the corresponding 3D reconstructions computed in real-time (where each reconstruction was computed from the depicted video frame and three additional video frames not shown in the figure).

We make no quantitative claims about the accuracy of the resulting 3D models, though [12] has shown that a similar system is capable of acquiring measurably accurate mod-

Figure 5. Example frames of live video (left) and corresponding real-time 3D reconstructions (right) displayed as textured wire-frame meshes.

els of static objects. Rather, we claim that these results are qualitatively acceptable for an application such as a real-time 3D chat system.

The average reported timings for each step of the algorithm during a single cycle are: surface normal computation (9 ms), depth map computation (32 ms), combined lighting, 3D display, and video capture (26 ms). Thus, the most expensive step in the computation is the Gauss-Seidel relaxation used to compute the depth map from surface normals, which is more than three times as expensive as computing the surface normals themselves.

4.1 Limitations

The method we have described is applicable in a wide variety of situations and is not limited to capturing human faces, though the surface normal computation implicitly assumes a lambertian surface and does not account specifically for specularities. The three largest limitations of the system are related to lighting, motion, and synchronization.

The largest limitation of the current method pertains to the ambient lighting conditions during capture. For best performance, the surroundings should be as dark as possible such that the computer screen is the predominant illuminant during capture. Though we can subtract out ambient illumination (Section 3.4), the resulting models are of lower quality than those captured in complete darkness.

Our system works in real time, enabling the capture of scenes in motion. However, because the method includes no explicit motion compensation, the result is a form of 3D motion blur in which the surface normals and depth maps may be incorrect during large motion. As described in Section 3.2 however, the system is able to recover completely just N frames after the large motion stops. Note that the motion problem disappears completely in the multispectral case (Section 3.4).

Finally, synchronization between lighting and capture is an important issue. As frame rates increase, issues of camera latency and LCD response time become important for synchronization and to prevent aliasing in real-time capture. The most important factor is the variability in image capture latency, which may lead to multiple images being captured under the same lighting conditions, and simultaneously to a failure to capture any image at all during another set of lighting conditions. However, at the $10\,hz$ speeds we deal with here, synchronization has not yet been a large issue.

5 Conclusion

We have a presented a real-time photometric stereo solution for capturing textured 3D models using only the illumination provided by a computer screen. The two primary contributions of this work have been: (1) a software-based photometric stereo solution that exploits commodity hardware already present in millions of homes across the world and (2) a real-time photometric stereo method for capturing and displaying 3D models in real-time. In future work, we expect to increase the frame rate of the system by making use of the GPU.

References

[1] R. Basri and D. Jacobs. Photometric stereo with general, unknown lighting. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 374–381, June 2001.

[2] P. Belhumeur and D. Kriegman. What is the set of images of an object under all possible illumination conditions? *Intl. J. of Computer Vision*, 28(3):245–260, 1998.

[3] J. Bouguet and P. Perona. 3d photography on your desk. In *Intl. Conf. on Computer Vision (ICCV)*, 1998.

[4] Yung-Yu Chuang, Douglas E. Zongker, Joel Hindorff, Brian Curless, David H. Salesin, and Richard Szeliski. Environment matting extensions: Towards higher accuracy and real-time capture. In *Proceedings of ACM SIGGRAPH*, pages 121–130, 2000.

[5] J. J. Clark. Active photometric stereo. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 29–34, 1992.

[6] James J. Clark. Photometric stereo with nearby planar distributed illuminants. In *Canadian Conference on Computer and Robot Vision*, 2006.

[7] Apple Computer Inc. Photo Booth, 2005.

[8] P. Debevec, T. Hawkins, C. Tchou, H. Duiker, W. Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. In *SIGGRAPH*, pages 145–156, 2000.

[9] R. Epstein, A. L. Yuille, and P. N. Belhumeur. Learning object representations from lighting variations. *Object Representation in Computer Vision II*, pages 179–199, 1996.

[10] Russel Epstein, P. W Hallinan, and Alan L Yuille. 5+/-2 eigenimages suffice: An empirical investigation of low-dimensional lighting models. Technical report, Division of Applied Sciences, Harvard University, 1995.

[11] D. Filippini, S. P. S. Svensson, and I. Lundstrom. Computer screen as a programmable light source for visible absorption characterization of (bio)chemical assays. *Chem. Commun.*, (2):240–241, 2003.

[12] Nathan Funk and Yee-Hong Yang. Using a raster display for photometric stereo. In *Canadian Conference on Computer and Robot Vision*, pages 201–207, 2007.

[13] H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *J. Opt. Soc. Am. A*, 11:3079, 1994.

[14] C. Hernandez, G. Vogiatzis, G. J. Brostow, B. Stenger, and R. Cipolla. Non-rigid photometric stereo with colored lights. In *Intl. Conf. on Computer Vision (ICCV)*, 2007.

[15] T. Malzbender, B. Wilburn, D. Gelb, and B. Ambrisco. Surface enhancement using real-time photometric stereo and reflectance transformation. In *Eurographics Symposium on Rendering*, pages 245–250, 2006.

[16] M. Proesmans, L. Van Gool, and A. Oosterlinck. One-shot active 3d shape acquisition. In *Intl. Conf. on Pattern Recognition (ICPR)*, page 336, 1996.

[17] A Shashua. *Geometry and Photometry in 3D Visual Recognition*. PhD thesis, MIT, 1992.

[18] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.

[19] R. J. Woodham. Gradient and curvature from photometric-stereo method including local confidence estimation. *J. Opt. Soc. Am. A*, 11(11):3050, 1994.

[20] A. Yuille, D. Snow, R. Epstein, and P. Belhumeur. Determining generative models of objects under varying illumination: Shape and albedo from multiple images using svd and integrability. *Intl. J. of Computer Vision*, 35(3):203–222, 1999.

[21] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *SIGGRAPH*, 2004.