Hybrid 3D Reconstruction Using Photometric Stereo and Photogrammetry

Yi Cheng Zhu, Cheng Xu, Miaoqi Zhang

Abstract

3D reconstruction of objects is a fundamental task in computer vision, with applications in areas such as robotics, entertainment, and cultural heritage preservation. In this paper, we propose a hybrid approach that combines photogrammetry and near-light photometric stereo to achieve high-quality 3D reconstructions. Our method leverages a reference camera for acquiring images of the object under varying illumination conditions and a floating camera for capturing images from different angles. The pose of the floating camera is estimated using the Scale-Invariant Feature Transform (SIFT), Fast Library for Approximate Nearest Neighbors (FLANN), and Random Sample Consensus (RANSAC) algorithms. The 3D reconstruction is then achieved by integrating the surface normal map generated using the near-light photometric stereo approach and the point cloud obtained from the photogrammetry pipeline. Our experiments demonstrate the feasibility and potential of such a method.

1. Introduction

3D reconstruction of objects has been a topic of interest in computer vision for many years. Traditional methods for 3D reconstruction include photogrammetry, which uses multiple overlapping images to estimate the structure and geometry of an object, and photometric stereo, which estimates surface normals from images taken under different illumination conditions. Both of these methods have their strengths and weaknesses; photogrammetry can provide accurate geometry but may struggle with textureless or reflective surfaces, while photometric stereo can recover fine surface details but is sensitive to lighting conditions and may not provide accurate depth information.

In this work, we propose a novel hybrid approach that combines the strengths of both photogrammetry and near-light photometric stereo to achieve high-quality 3D reconstructions. Our method employs a reference camera for acquiring images of the object and a floating camera for capturing images under varying illumination conditions. We estimate the pose of the floating camera and predict the point cloud of the scene at the same time with the structure-from-motion algorithm. The photogrammetry pipeline focuses on using floating camera images for the point cloud generation, while the near-light photometric stereo approach generates a surface normal map from acquired data of the reference camera and floating camera's light source positions.

The resulting surface normal map and depth maps are then integrated to generate a complete 3D reconstruction of the object. This hybrid approach benefits from the combined strengths of both photometric stereo and photogrammetry, achieving higher quality reconstructions than either method alone. In this paper, we present the details of our approach and try to demonstrate its effectiveness through a series of experiments on real-world objects.

2. Related Works

It has been shown by our sister field of computer graphics that having both surface normal and point cloud data has numerous benefits, including better interpolation of the model [7] and greater robustness to perturbation in measurements. [2]

In the field of computer vision, however, we usually use one to calculate the other for ease of setup, either with Poisson Surface Reconstruction [5] or the reverse process. [6].

This project is intended to build upon the prior works of Yeh, Chia-Kai, et al. [10]. Where their team proposed the use of a stationary camera to both act as a grounded coordinate frame for obtaining the pose for photogrammetry and also the camera for photometric stereo. With this setup, and assuming that the point light source(flashlight) is close to the moving camera, the measurement process would be simplified to taking a few photos with the flash on. We plan on utilizing this process and exploring the potential of combining both surface normal and depth maps to obtain a more accurate measurement.

3. Theory

3.1. overall setup

The aim of the project is to create a workflow that demands as little input and setup as possible from the user. We have simplified the hardware requirement to two phones, a tripod, and a printed Aruco marker, and the workflow from

the user's perspective is as follows: the user will need to set up one phone on the tripod facing downwards and start recording the video. Then the user should set up the scene with an Aruco marker in the frame. The user then only needs to take 7-10 pictures with the other phone with the flashlight on at various different angles. The pipeline should then be able to extract all the information needed to create a high-quality 3d reconstruction.

3.2. calibration

The calibration matrices(the camera intrinsic matrix and the distortion coefficients) are needed at various points in the calculations. Such matrices can usually be accessed through APIs on majorities of Android and Apple smartphones, but in the event that they are not available, we also built in a camera calibration function that requires the user to take multiple pictures of ChAruCo boards at various angles.

3.3. Photogrammetry

Photogrammetry is a technique that derives 3D information from multiple overlapping images of an object or scene. The underlying principles of photogrammetry are rooted in projective geometry and the concept of structure-from-motion (SfM). The key steps involved in a typical photogrammetry pipeline include:

1. Camera calibration: Determining the intrinsic and extrinsic parameters of the camera, such as the focal length, distortion coefficients, and camera pose. Accurate calibration is crucial for obtaining reliable 3D reconstructions.

2. Feature detection and matching: Identifying key points in the images and establishing correspondences between them to determine their relative positions and orientations. This process usually involves the use of robust feature detectors and descriptors, such as SIFT or SURF, and efficient matching algorithms like FLANN.

3. Camera pose estimation: The pose of a camera is defined by its position and orientation in the 3D world. Estimating the camera pose involves determining these parameters for each image in the input sequence. This is usually done by applying the essential matrix or the fundamental matrix to estimate the relative pose between the camera pairs. The absolute pose of the cameras is obtained by chaining the relative poses through the sequence.

4. Triangulation: Once the camera poses are known, the next step is to estimate the 3D position of the points in the scene. This is done through triangulation, which involves finding the intersection of the viewing rays from multiple camera viewpoints for a given point correspondence. The intersection of these rays gives the 3D position of the point in the world. Several methods can be used for triangulation, such as linear triangulation, direct linear transformation (DLT), or iterative methods like Levenberg-Marquardt.

5. Bundle adjustment: The camera pose estimation and triangulation steps might have errors due to noise in the feature matching or inaccuracies in the pose estimation. Bundle adjustment is a global optimization technique that refines the camera poses and 3D points simultaneously to minimize the reprojection error, which is the difference between the observed 2D positions of the points in the images and the positions obtained by projecting the 3D points onto the images using the estimated camera poses. This optimization is typically performed using a non-linear least squares method like Levenberg-Marquardt.

6. Dense reconstruction: Estimating depth information for each pixel in the images by finding correspondences between overlapping images. This step can be achieved using multi-view stereo techniques, such as semi-global matching (SGM), patch-based methods, or depth map fusion algorithms.

3.4. Near-Light Photometric Stereo

The Near-Light Photometric Stereo approach is based on the work of Woodham and further developed by other researchers. This method estimates the surface normal map of an object using acquired data from the reference camera and the floating camera's light source positions. The key steps involved in the near-light photometric stereo pipeline are:

1. Illumination vector computation: For each floating camera image, the light source position is approximated using the camera's location or a pre-calculated adjustment factor. Accurate estimation of the illumination vectors is essential for reliable surface normal estimation.

2. Image preprocessing: The reference camera images are preprocessed to reduce noise, correct for illumination inconsistencies, and enhance the photometric stereo process. This step may involve normalization, filtering, or other techniques, depending on the quality of the images.

3. Pixel-wise intensity measurements: For each pixel in the reference camera images, the intensity measurements are collected to form a vector $I_i = (I_{i1}, I_{i2}, ..., I_{in})$, where n is the number of images and I_{ij} is the intensity of the i-th pixel in the j-th image.

4. Surface normal estimation: Using the illumination vectors L_i and the intensity measurements I_i, the surface normal $\mathbf{N}$ for each pixel can be estimated using a least-squares approach. The surface normals can then be combined to create a surface normal map for the object.

5. Integration and fusion: The surface normal map and the depth maps obtained from the photogrammetry pipeline are integrated to generate a complete 3D reconstruction of the object. This may involve a multi-view stereo approach or other fusion techniques to ensure consistency and accuracy in the final output. The resulting 3D model benefits from the combined strengths of both photometric stereo and photogrammetry, achieving higher quality reconstructions than either method alone.

4. Analysis

4.1. Photogrammetry Analysis

The photogrammetry process can be analyzed based on the principles of projective geometry and structure-from-motion (SfM). SfM is a technique for estimating the 3D structure of a scene and the camera motion using a set of 2D images. In the SfM approach, camera poses and 3D points are estimated simultaneously, making it a non-linear optimization problem. This problem can be solved iteratively using bundle adjustment, which refines the camera poses and 3D points by minimizing the reprojection error between observed and predicted image features.

4.1.1 Feature extraction and matching

The first step in SfM is to extract distinctive features (points, corners, or edges) from each image in the sequence. Common feature extraction algorithms are the Scale-Invariant Feature Transform (SIFT), Oriented FAST and Rotated BRIEF (ORB), or Speeded-Up Robust Features (SURF). Once features are extracted from each image, we need to find correspondences between the features across the images. This is done using feature descriptors, which are vectors that describe a feature's local neighborhood. Feature matching can be performed using methods like Nearest Neighbor Search or Lowe's ratio test.

4.1.2 Camera motion estimation

We can estimate the camera motion (rotation and translation) between two images using the Essential Matrix (E) or the Fundamental Matrix (F). The Essential Matrix relates the corresponding points in two images when the camera's intrinsic parameters are known, while the Fundamental Matrix is used when the camera's intrinsic parameters are unknown.

For the Essential Matrix, let's denote two corresponding points in the images as p_1 and p_2. We have:

$$p_2^T * E * p_1 = 0$$

where p1 and p2 are homogeneous coordinates of the corresponding points, and E is the Essential Matrix.

The Fundamental Matrix, F, is defined similarly:

$$p_2^T * F * p_1 = 0$$

If the camera's intrinsic parameters are known, we can compute the Essential Matrix (E) from the Fundamental Matrix (F) as:

$$E = K^T * F * K$$

where K is the camera's intrinsic matrix.

The Essential Matrix can then be decomposed into the camera's rotation matrix (R) and translation vector (t) using the Singular Value Decomposition (SVD) method.

4.1.3 Triangulation

Once the camera motion is estimated, we can find the 3D position of the matched feature points in the scene using triangulation. Given the camera matrices P_1 and P_2 for the two views, and their corresponding points x_1 and x_2, we want to find the 3D point X that minimizes the reprojection error:

$$argmin||x_1 - P_1 * X||^2 + ||x_2 - P_2 * X||^2$$

The solution can be found using the Direct Linear Transformation (DLT) algorithm or the Least Squares Triangulation method.

4.1.4 Bundle adjustment

After obtaining initial estimates of the camera motion and 3D points, bundle adjustment is used to refine these estimates by minimizing the reprojection error across all images and points. The reprojection error is defined as the difference between the observed image point and the projection of the corresponding 3D point onto the image plane:

$$E = sum(||x_{ij} - P_i * X_j||^2)$$

where x_ij is the observation of the j-th point in the i-th image, P_i is the projection matrix of the i-th camera, and X_j is the 3D position of the j-th point. This is a non-linear optimization problem and can be solved using the Levenberg-Marquardt algorithm.

The accuracy of the photogrammetry pipeline depends on various factors, such as the quality and resolution of the

images, the baseline (i.e., the distance between cameras), the number of images, and the quality of the feature matching and dense reconstruction algorithms used. By analyzing the performance of the photogrammetry process on different datasets and scenarios, one can identify the strengths and limitations of the method and explore potential improvements.

4.2. Near-Light Photometric Stereo Analysis

The near-light photometric stereo approach can be particularly advantageous when dealing with objects that have textureless or reflective surfaces, as these characteristics can pose challenges for traditional photogrammetry methods. By incorporating the near-light photometric stereo technique into the 3D reconstruction pipeline, the resulting model can achieve improved surface details, accurate geometry, and robust performance on a wider range of object types and surface properties.

The Near-Light Photometric Stereo approach is based on the work of Woodham [9] and further developed by other researchers [3, 4]. This section outlines the process of generating a surface normal map using the acquired data from the reference camera and the floating camera's light source positions.

4.2.1 Illumination Vectors

For each floating camera image, the light source position is approximated using the camera's location or a precalculated adjustment factor. This results in a set of illumination vectors, $L_i = (l_{ix}, l_{iy}, l_{iz})$, representing the direction and intensity of the light source.

4.2.2 Image Preprocessing

The reference camera images corresponding to each floating camera image will be preprocessed to reduce noise and enhance the photometric stereo process. This may involve normalization, filtering, or other techniques, depending on the quality of the images.

4.2.3 Pixel-wise Intensity Measurements

For each pixel in the reference camera images, the intensity measurements are collected to form a vector $I_i = (I_{i1}, I_{i2}, ..., I_{in})$, where n is the number of images and I_{ij} is the intensity of the i-th pixel in the j-th image.

4.2.4 Surface Normal Estimation

We estimate the surface normals from photometric stereo following the methods proposed by [8].

We first re-scale the world coordinates to pixel coordinates by multiplying by the constant

$$d = \frac{\mu \hat{z}}{f}$$

where f is the focal length, μ is the camera pixel size, and $\hat{z}$ the approximate mean depth of the object from the camera.

Let a single light source be at e_k, we describe a single light pencil from light source k at pixel p as

$$\hat{L}_{pk} = \frac{L_k - X_p}{||L_k - X_p||^2} e_k$$

where X_p is the 3D location of the pixel. The depth component of X_p is initially set to 1, and we iteratrively solve the depth map.

Using the illumination vectors L_i and the intensity measurements I_i, the surface normal $\mathbf{N}$ for each pixel can be estimated using a least-squares approach:

The image intensity at a pixel is then described as

$$I_{pk} = B_p^T \hat{L}_{pk}$$

where B_p is the surface normal at the location.

We solve for B_p by computing the pseudo-inverse of $\hat{L}_{pk}$.

Using B_p, we construct the gradient field by dividing out the Z component B_p.

This is often a non-integrable gradient field.

Following the methods of [1], we attempt to solve a plausible solution to this non-integrable gradient field.

Let the two components of the gradient field be (p, q). At each pixel we construct the 2×2 matrix H by gaussian blurring component-wise the matrix $\begin{bmatrix} p^2 & p \times q \\ p \times q & q^2 \end{bmatrix}$. Let $u_1 u_2$ be the eigenvalues of H, construct new eigenvalues as

$$\lambda_1 = \begin{cases} 1 & u_1 = 0 \\ \beta + 1 - \exp -3.315/u_1^4 & u_1 > 0 \end{cases}$$

$$\lambda_2 = 1$$

Construct the matrix D using eigenvectors and eigenvalues λ_1, λ_2.

Then let

$$D = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix}$$

$$u_D = \text{div}(d_{11}p + d_{12}q, d_{21}p + d_{22}q)$$

We solve for the new depth map Z as

$$\nabla_D^2 Z = u_D$$

Then, by iteratively updating the normal map, then integrating the resulting gradient field for a new depth map, it converges into the accurate final depth map.

4.2.5 Integration and Fusion

The surface normal map and the depth maps obtained from the photogrammetry pipeline will be integrated to generate a complete 3D reconstruction of the object. This may involve a multi-view stereo approach or other fusion techniques to ensure consistency and accuracy in the final output. The resulting 3D model will benefit from the combined strengths of both photometric stereo and photogrammetry, achieving higher quality reconstructions than either method alone.

The accuracy of the near-light photometric stereo approach is influenced by several factors, including the quality and consistency of the illumination vectors, the precision of the surface normal estimation, and the robustness of the integration and fusion techniques. By examining the performance of the method on various datasets and under different conditions, one can assess its capabilities and identify areas for improvement.

5. experimental results

5.1. Frame extraction

In order for the hybrid method to function, we need to establish a mapping between the pictures from the overhead camera and the portable camera. We have decided to record the overhead camera as a video so as to not worry about the temporal syncing of the shutter until the preprocessing step. At the preprocessing step, we first evaluate the light level at each frame (see figure. 1). Then we extract the highest n amount of frames with the restrain that the frames will not be within 0.2 seconds of each other and n being the number of photos the portable camera took. We then convolve the neighboring two frames with the selected frame to account for the rolling shutter effect to finalize our preprocessing step. The corresponding pairs of images can be seen in figure. 8.

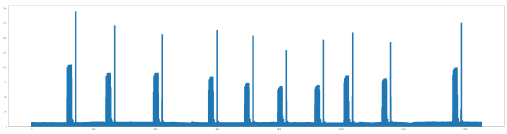


Figure 1. light level over each frame of the video

5.2. calibration

To provide a tool for calibration for cases where the internal matrix and the distortion coefficients are not accessible, we created a supplementary process to calibrate the cameras. We used the existing implementation of charuco boards in opencv to perform the calibration. The result and validations can be seen in figure. 2.

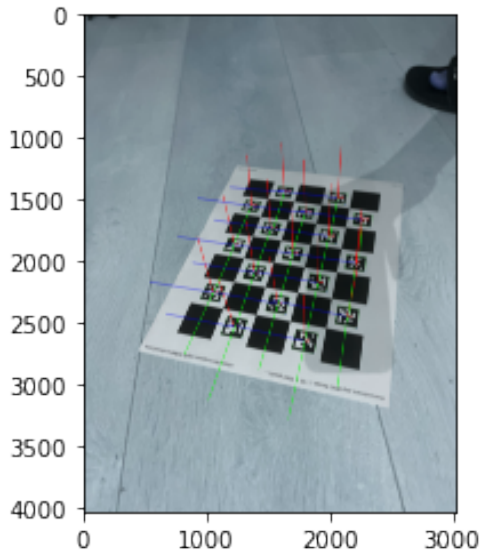


Figure 2. calibration done with charuco boards

5.3. pose estimation

Although the poses can be obtained by Structure from Motion, to increase robustness, we provided to option to use an aruco board as the poses for the photometric stereo step. We chose a 2x2 aruco board to ensure a more robust measurement when compared to a single marker. We once again leverage the libraries provided by opencv for this function. The unit test for the submodule can be seen in figure. 3.

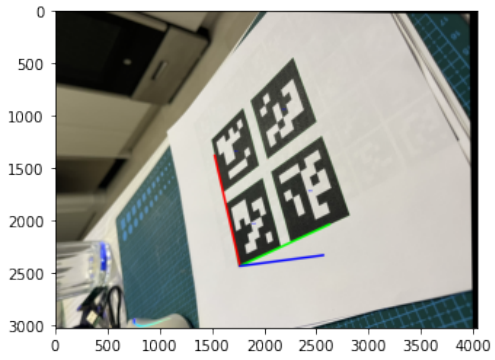


Figure 3. pose estimation using aruco board

5.4. photogrammetry

Our photogrammetry pipeline managed to extract the correct point cloud for the scene in the dataset(see figure. 4). Unfortunately we are unable to implement the dense estimation step in time. as a result, our depth map is not able to get per-pixel depth value. We will touch on that in the depth estimation section.

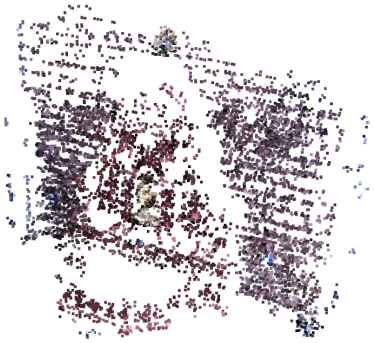


Figure 4. the point cloud output of our photogrammetry algorithm

5.5. depth estimation

As previously mentioned, our point cloud is not dense enough to direct convert to a depth map. As a result, we had to linearly interpolate the depth map for pixels where there are sufficient neighboring data. The result of this step can be seen in figure. 5.

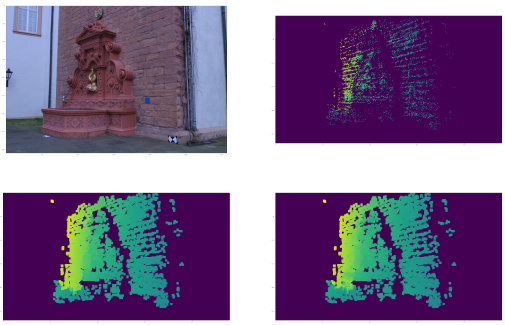


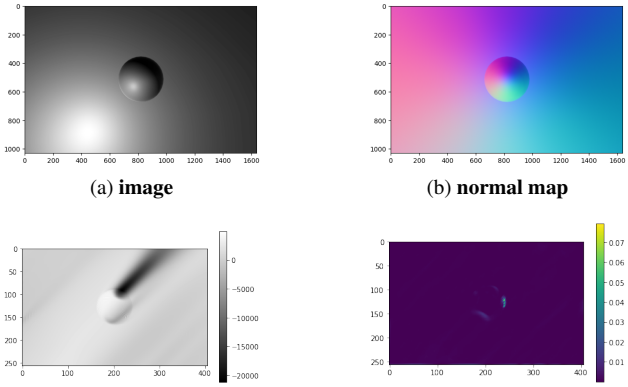
Figure 5. the depth map directly from projecting and after interpolation.

5.6. photometric stereo

The result for photometric stereo can be seen in Figure 6. and Figure 7.

6. concluding remarks

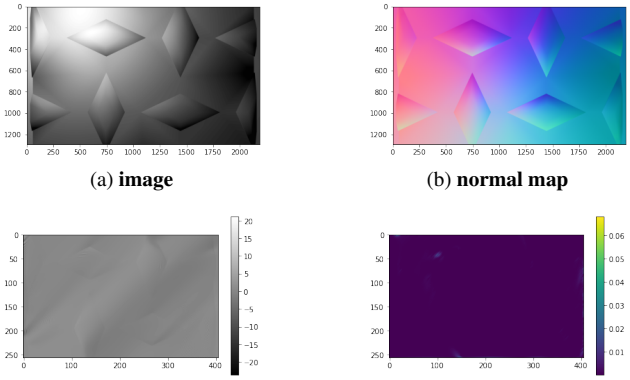
We have shown throughout this project that it is plausible to create a pipeline that incooperates both photogrammetry and photometric stereo while still maintaining the simplicity of only photogrammetry. Unfortunately, we were unable to implement the dense point cloud estimation part of photogrammetry, which created a rift in the entire pipeline and prevented us from testing it end to end. We were also unable to correlate the scale of the scene with its real-world measurements without the help of aruco markers. Those are the areas that warrant further exploration.



(a) **image** (b) **normal map**

(c) **depth map** (d) **error**

Figure 6. The result from photometric stereo



(a) **image** (b) **normal map**

(c) **depth map** (d) **error**

Figure 7. The result from photometric stereo

References

[1] Amit Agrawal, Ramesh Raskar, and Rama Chellappa. What is the range of surface reconstructions from a gradient field? In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, pages 578–591. Springer, 2006. 4

[2] Doris Antensteiner, Svorad Štolc, and Thomas Pock. A review of depth and normal fusion algorithms. *Sensors*, 18(2), 2018. 1

[3] R. Basri and D.W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):218–233, 2003. 4

[4] Heng Guo, Hiroaki Santo, Boxin Shi, and Yasuyuki Matsushita. Edge-preserving near-light photometric stereo with neural surfaces, 2022. 4

[5] Michael M. Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In Alla Sheffer and Konrad Polthier, editors, *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, volume 256 of *SGP '06*, pages 61–70, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association. 1

[6] Niloy J. Mitra and An Nguyen. Estimating surface normals in noisy point cloud data. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, SCG '03, page 322–328, New York, NY, USA, 2003. Association for Computing Machinery. 1

[7] Takashi Nagata. Simple local interpolation of surfaces using normal vectors. *Computer Aided Geometric Design*, 22(4):327–347, 2005. 1

[8] Thoma Papadhimitri and Paolo Favaro. Uncalibrated near-light photometric stereo. 2014. 4

[9] Robert Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19, 01 1992. 4

[10] Chia-Kai Yeh, Nathan Matsuda, Xiang Huang, Fengqiang Li, Marc Walton, and Oliver Cossairt. A streamlined photometric stereo framework for cultural heritage. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part I 14*, pages 738–752. Springer, 2016. 1