3D Stereo Reconstruction Using Multiple Spherical Views

Ifueko Igbinedion
Deaprtment of Electrical Engineering
Stanford University
ifueko@stanford.edu

Harvey Han
Department of Electrical Engineering
Stanford University
hanhs@stanford.edu

***Abstract*—360 degree stereoscopic image capture makes it easier to capture full scenes in a limited number of images. Because of this quality, it is useful to utilize the these spherical images in computer vision and virtual reality applications such as depth estimation and 3D scene reconstruction. Based on the principles of disparity map generation, previously explored by a Spring 2015 EE368 project, we aim to improve 3D stereo reconstruction by using multiple spherical views. The spherical images are captured by two vertically displaced Ricoh Theta cameras. Each pair of spherical images allows us to generate a disparity map and depth information that can be used for 3D reconstruction. Utilizing multiple viewpoints during scene reconstruction can allow for more robustness when creating translated views. In this paper, we discuss our method for improving these depth maps by utilizing multiple spherical views to improve the 3D reconstruction of scenes.**

Keywords–3D, stereoscopic, scene reconstruction, disparity map, depth map, multiple viewpoints, spherical images

I. Introduction

Stereoscopic image rectification is a widely studied topic in image processing that provides the ability to estimate 3D depth from 2D input images. Because of this popularity, depth-based stereoscopic image rendering and 3D reconstruction receives a great deal of attention in areas of multimedia research, more recently because of the potential applications in 3D television [3][5]. Using a single pair of spherical images allows us to perform depth estimation from one perspective. This allows to achieve depth accuracy to a certain degree, but if we utilize multiple viewpoints for the same scene during 3D reconstruction, we can allow for more robustness when creating 3D Views. Figure 1 shows the two-viewpoint (epipolar) vs. three-viewpoint geometry for spherical cameras. The small displacement inaccuracies that arise when using epipolar geometry can have a large aggregate effect on disparity maps. Having another reference camera allows for more accurate triangulation during reconstruction, and consequently more accurate depth and disparity estimation.. Motivated by this fact, we aim to use multiple view points of spherical images to improve 3D stereo reconstruction.

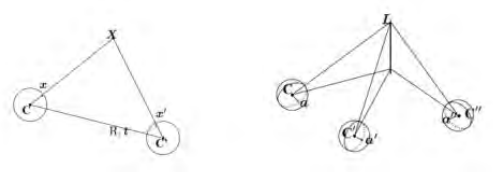


Fig. 1. Epipolar vs. three-viewpoint geometry.

The remainder of this report is structured as follows. First, we discuss previous research that allow us to build a deeper understanding of the fundamentals behind spherical camera capture and depth map estimation. Next we, we describe our setup and algorithm for utilizing multiple spherical views for scene reconstruction. Finally, we show experimental results, discuss some of the challenges related to 3D reconstruction of spherical scenes, and explore potential applications in virtual reality research and industry.

II. Related Work

Understanding spherical geometry is the first step to reconstructing depth from spherical images. Previous work has been done to derive geometry for two and three viewpoint configurations. The symmetric nature of a sphere changes both the representation of pixel locations and calculation of depth from disparity, and thus should be considered in reconstruction [9].

Many have explored generating depth maps from stereoscopic views. Kim et.al. succeed in generating depth maps using the principles of spherical geometry, and employ various smoothing and averaging functions in order to remove noise from the generated depth map [Kim]. While depth can be estimated using just spherical geometry, some have shown that optical flow is useful in disparity estimation from stereo views [2]. Others have achieved 3D reconstruction using omni-directional cameras, much like the Ricoh Theta cameras utilized in this project, but rather than using perspective views, a slanted-plane Markov model is utilized for depth map generation [7] Additionally, while utilizing undistorted pinhole modeled cameras, some advocate for the use of multiple (up to hundreds) of viewpoints for dense 3D reconstruction. Acknowledging the lack of calibration tools for relating multiple camera views, Seitz et. al. examine and advocate for multiple viewpoint reconstruction algorithms. [8]. These previous contributions guide our proposed reconstruction algorithm, which allows us to go from optical flow disparity calculation to noise reduction in 3D depth reconstruction. The following section describes the our setup and reconstruction algorithm.

III. Methodology

Camera Capture Setup

We utilize a 5-viewpoint camera capture system, with three cameras equally spaced on a common baseline, and two additional cameras on a vertically shifted baseline, angled towards the scene at around 30 degrees. Each camera location contains a pair of vertically stacked spherical cameras, each of which can capture 185 degree scene that can be stitched into one 360 degree spherical view. This setup allows us to utilize two to five viewpoints in depth map rectification.

Disparity Map Calculation and Depth Estimation

Motion estimation from image sources allows us to obtain a large amount of information to support many computer vision algorithms, from object recognition to scene understanding. Optical flow estimation is a common structure-from-motion principle that allows us to estimate the x and y components of motion of objects in a two dimensional scene. [2]. Our camera capture setup dictates that the only motion in our scene should be vertical, and so we are able to utilize the y component of the optical flow as our disparity estimation. Typically, disparity calculation in spherical images is is distorted as a result of the fish-eye lenses, so we were surprised to discover that this estimation results in a less distorted disparity estimation, and so our depth estimation was approximated well using the flat image disparity-to-depth model,

$$Depth = f * b/d$$

where f is the focal length, b is the baseline length, and d is the calculated disparity. Figure 3 shows the optical flow estimation results for one of our test scenes, while Figure 4 shows the original, noisy depth map.

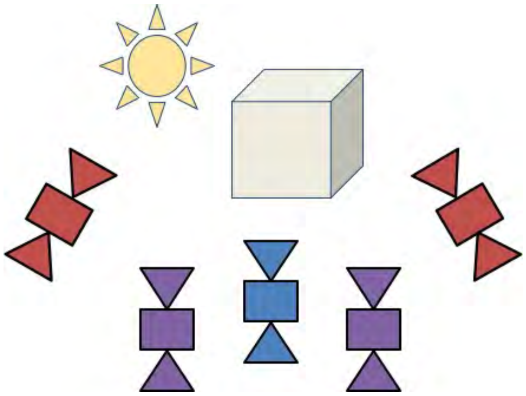


Fig. 2. Camera Capture setup.

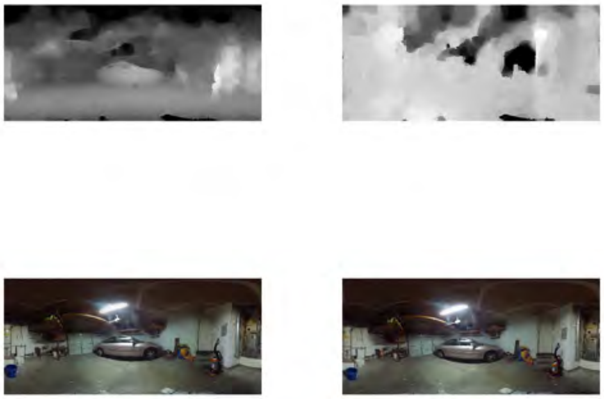


Fig. 3. Optical flow estimations for vertically displaced frames. The y-component (on the left) can be directly used for disparity estimation.

Fig. 4. Initial depth map for reference frame.

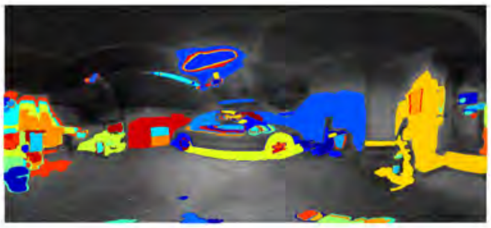


Fig. 5. MSERs for car scene.

The spherical model, however, depends on the spherical angle at the current pixel location [1],

$$Depth = \frac{\arcsin(b\sin(\theta)}{\arcsin(d)}$$

where b is the baseline length, θ is the corresponding spherical angle, and d is the calculated disparity. For our purposes, it is useful to be able to utilize the flat model, as it can be calculated more efficiently than the spherical model.

Depth Map Rectification and 3D reconstruction

From the generated depth maps, we choose one camera location as a reference for rectification and final reconstruction. The rectification process goes as follows: first, maximally stable extremal regions (MSERS) are averaged in the reference image to accentuate depth before denoising with other viewpoints. This is useful in preventing depth match rectification from flattening out noisy areas that likely correspond to foreground. Then, after matching SIFT keypoints in from the reference image, we perform box filtering at locations of SIFT keypoints and average corresponding patches in other viewpoints with the reference image. Finally, areas that remained untouched are slightly suppressed, as most of the foreground regions are found near MSERs and SIFT keypoints.

Challenges

We experienced various unexpected challenges during the development of this project, mostly related to image quality of our capture system. First, as spherical data is typically very high resolution, we are required to subsample images before processing. Not only does the rough subsampling result in artifacts in the images, but SIFT matching is less accurate as you continue to downsample the images. As a result, we had to balance the trade-off between accurate detection and algorithmic speed. An additional problem we encountered in the beginning of this project had to do with spherical distortion. Although our final algorithm used optical flow detection for disparity estimation, our original disparity map algorithm utilized a similarity accumulator technique [6] and required post-processing before depth estimation. The optical flow algorithm, though it does not solve all distortion issues, resulted in better disparity estimation than the previous technique, and so we opted to utilize it instead. Additionally, some scenes proved more difficult to reconstruct with the spherical camera. Optical flow when applied to large, flat areas of uniform color are typically difficult estimate. This fact combined with

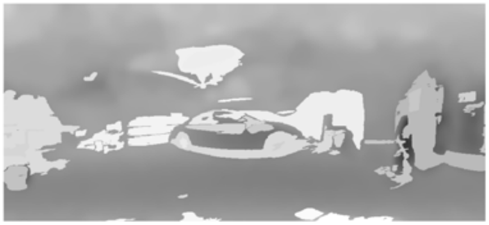


Fig. 6. Depth map rectified with MSERs, SIFT and additional frames.

the fish-eye camera's tendency to round straight lines leads to vertical camera displacement causing radial image displacement. Those areas were largely ignored in our reconstruction algorithm, but reasoning about them could lead to improved reconstruction algorithms. Lastly, camera capture conditions were a challenge. Under fluorescent, pulse-width modulated illuminants, Moire patterns can sometimes result, further skewing optical flow. These challenges can be visualized in some of the experimental results in the following section.

IV. Experimental Results

We tested our algorithm on four scenes: A car garage, a living room, a back yard, and a hallway. The car garage was the most successful example, with a good 3D representation of the car and various appliances in the garage. This can be attributed to The living room was not as successful, due to flat surfaces, like walls and patterned objects, like the carpet and curtain. The yard dataset was a special case, providing many cubic shapes and uniform lighting that provided good disparity readings with optical flow estimation. The hallway provided the least accurate results due to barrel distortion of the straight, long walls in the scene. The results of our algorithm on these datasets are shown at the end of the report.

V. Future Work

While this project did not fully succeed in believable 3D reconstruction, we believe there is area for adaptation and refinement of his algorithm. We did not explore any geometry based filtering algorithms, such as convex and visual hulls around objects. We noticed that this algorithm is more reliable a close distances and largely staggered viewpoints, so rather than utilizing 360 degree spherical cameras, it may be more logical to employ a 360 degree capture system for individual objects, and render scenes as worlds composed of 3D models, as most Virtual Reality development kits are modeled today.

Additionally, better methods could be employed in the actual 3D reconstruction of the scene. We opted to overlay the image onto the inverse of the depth map on 3D axes, however, a point cloud reconstruction would have resulted in a much more believable scene. Additionally, geometric estimation of image volumes based on shells around point cloud reconstruction can result in high-fidelity creation of realistic 3D models without the use of CAD software.

VI. Contributions

Contributions by Ifueko: Contributed to algorithmic development, poster, image capture, coding/debugging and final report.

Contributions by Harvey: Contributed to algorithmic development, coding/debugging, poster and final report.

References

[1] Arican, Zafer, and Pascal Frossard. Dense Depth Estimation from Omnidirectional Images. No. EPFL-REPORT-138767. 2009.

[2] Baraldi, Patrizia, Enrico De Micheli, and Sergio Uras. "Motion and Depth from Optical Flow." Alvey Vision Conference. 1989.

[3] C. Fehn, *Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV*, Proc. SPIE 5291, Stereoscopic Displays and Virtual Reality Systems XI, 93 (May 21, 2004).

[4] Kim, Hansung, and Adrian Hilton. "3D scene reconstruction from multiple spherical stereo pairs." International journal of computer vision 104.1 (2013): 94-116.

[5] L. Zhang; W. J. Tam, "Stereoscopic image generation based on depth images for 3D TV", *Broadcasting, IEEE Transactions*, vol.51, no.2, pp.191-199, June 2005.

[6] J. Schmidt, H. Niemann, and S. Vogt, Dense disparity maps in real-time with an application to augmented reality, in Applications of Computer Vision, 2002. (WACV 2002). Proceedings. Sixth IEEE Workshop on, 2002, pp. 225230.

[7] Schonbein, Miriam, and Andreas Geiger. "Omnidirectional 3d reconstruction in augmented manhattan worlds." *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on.* IEEE, 2014.

[8] Seitz, Steven M., et al. "A comparison and evaluation of multi-view stereo reconstruction algorithms." *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on. Vol. 1.* IEEE, 2006.

[9] Torii, Akihiko, Atsushi Imiya, and Naoya Ohnishi. "Two-and three-view geometry for spherical cameras." *Proceedings of the sixth workshop on omnidirectional vision, camera networks and non-classical cameras. Citeseer (cf. p. 81).* 2005.

Car Dataset

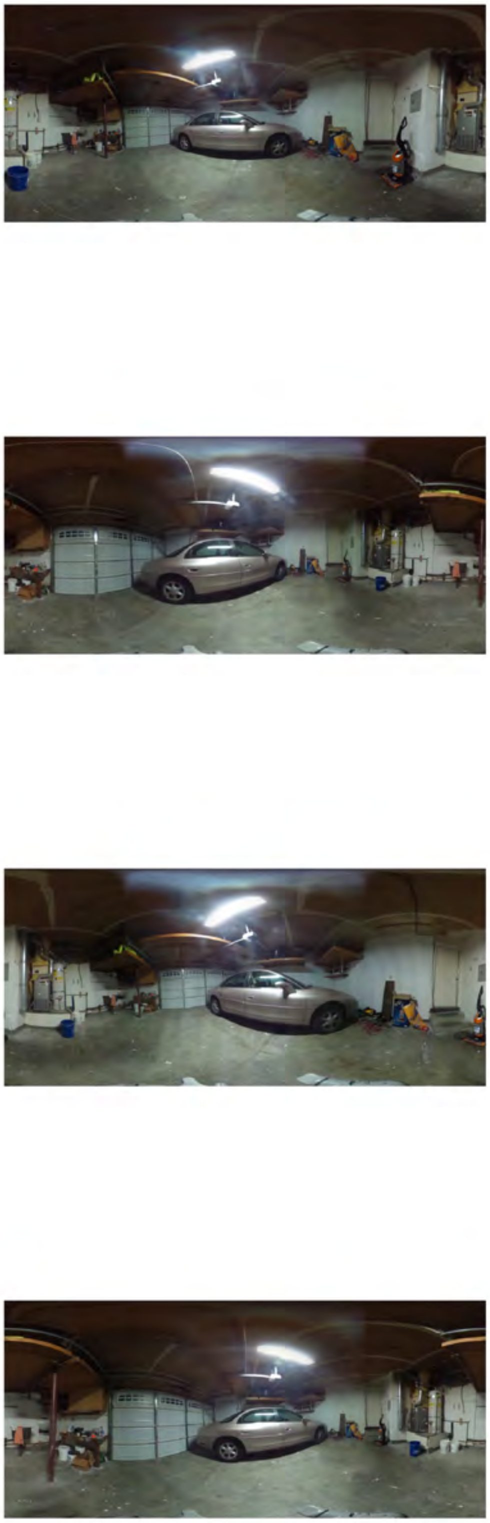


Fig. 7. Initial upper car frames.

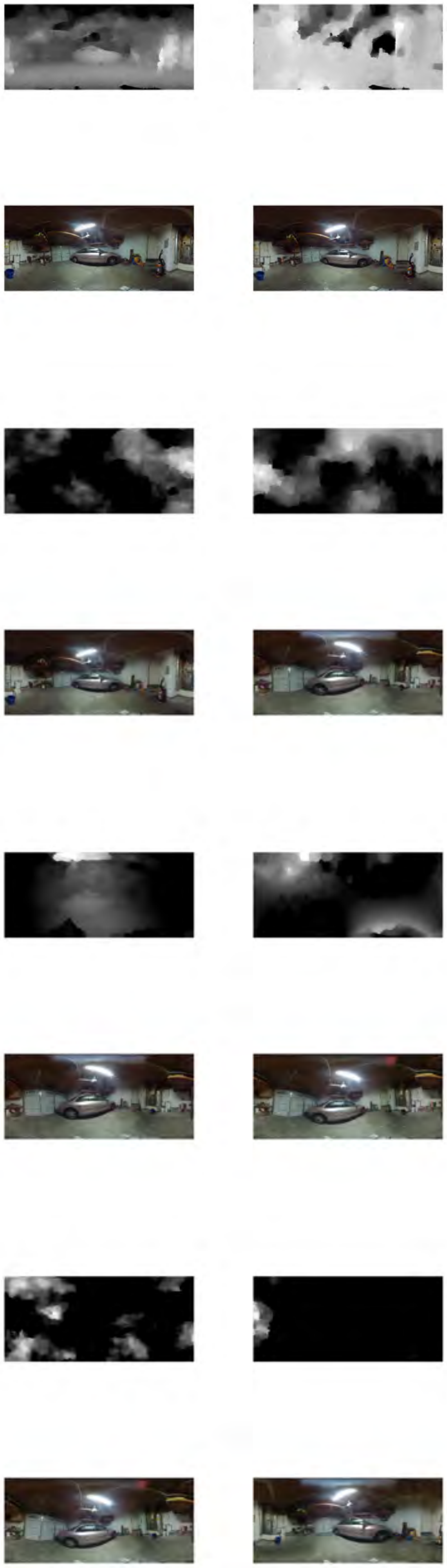


Fig. 8. Resulting optical flow estimation. The upper left corner is the y-component, used for disparity map calculation.

Fig. 9. Individual depth views for each frame.

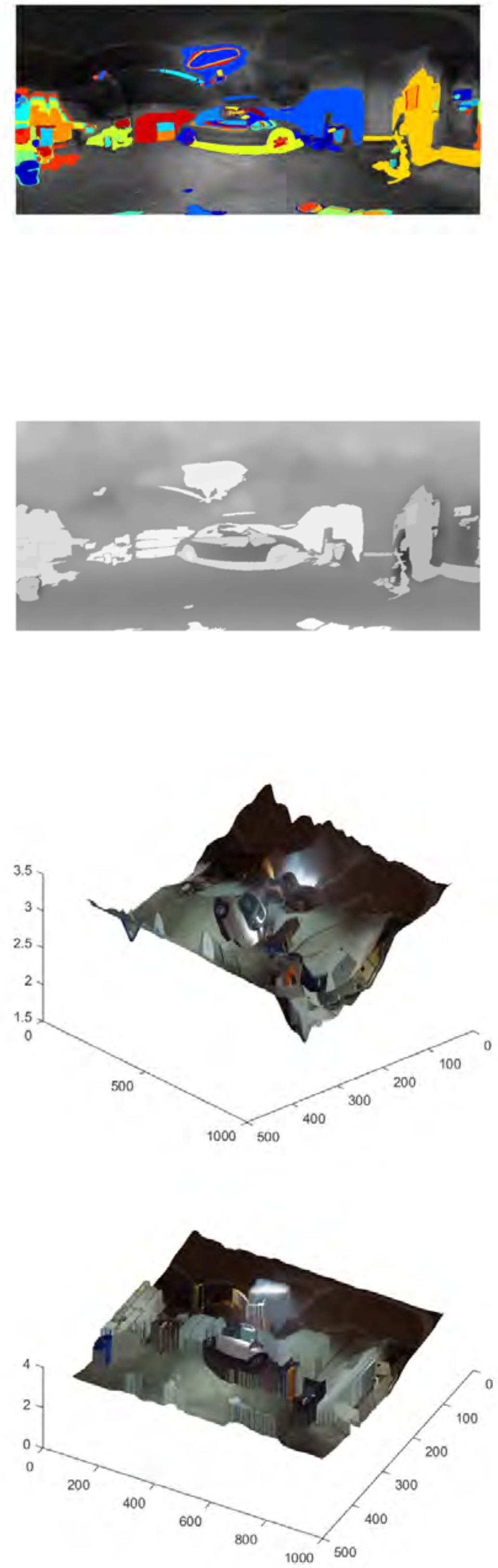
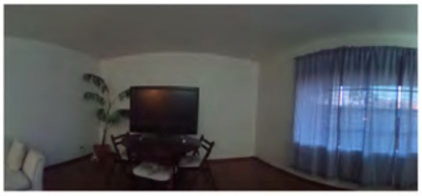
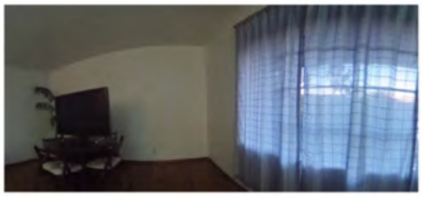
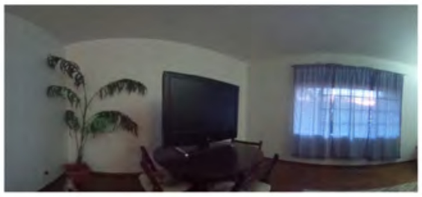


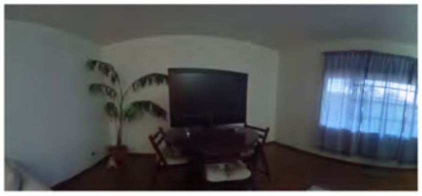
Fig. 10. MSER regions, final depth segmentation, and before and after reconstruction.

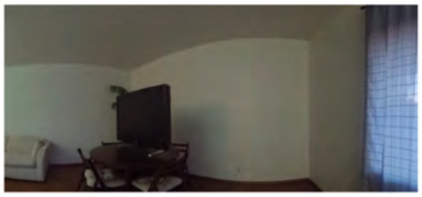
Room Dataset

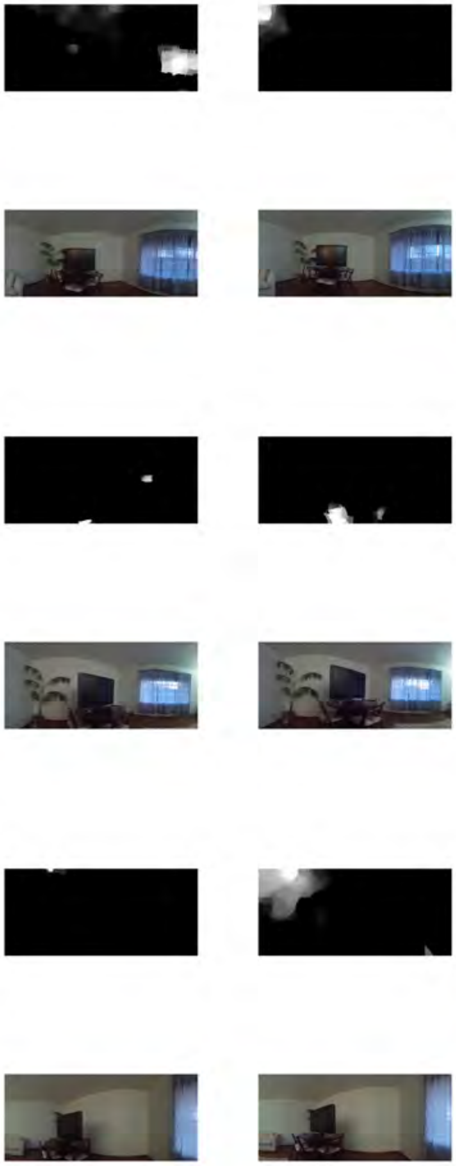


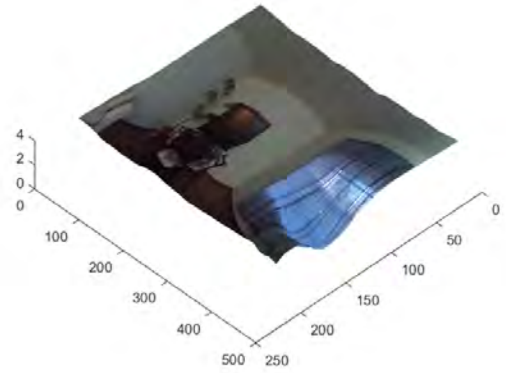


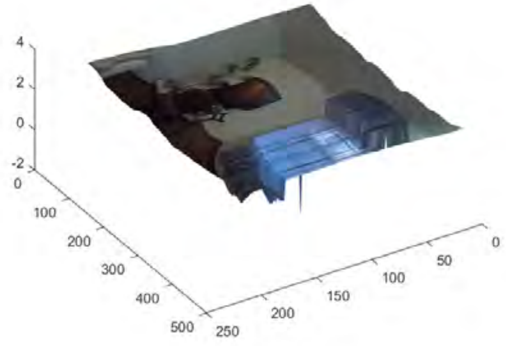




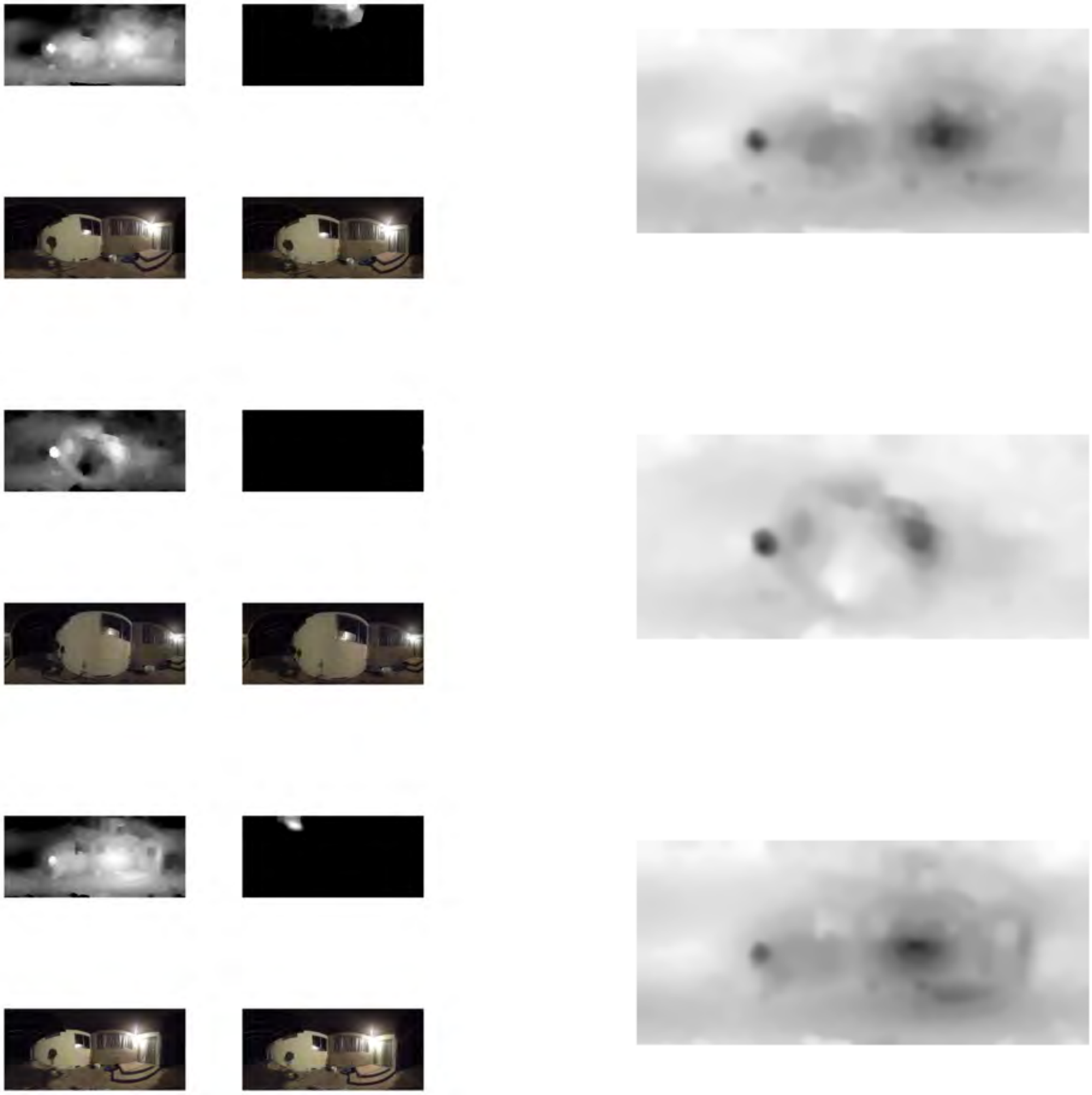




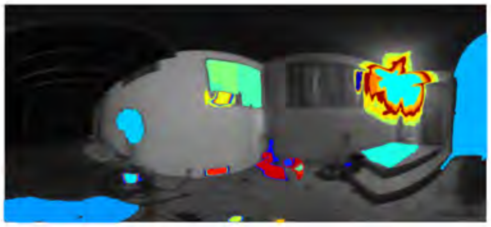


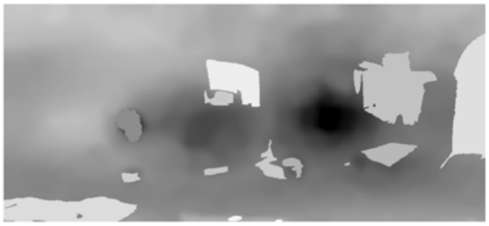


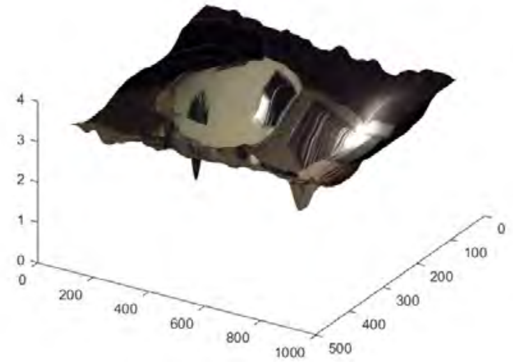
Yard Dataset

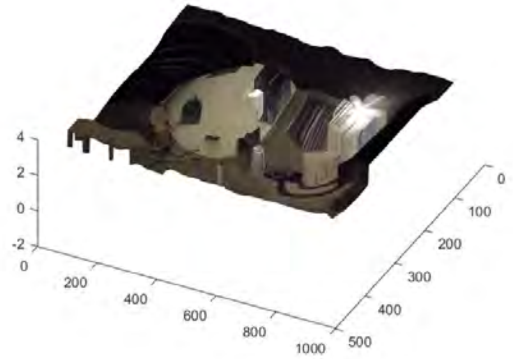


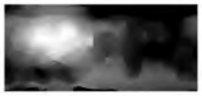
Hall Dataset

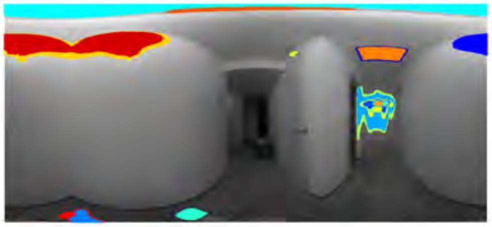


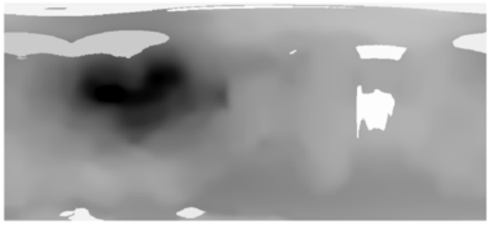


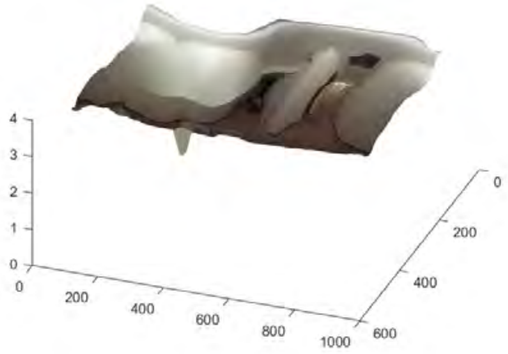


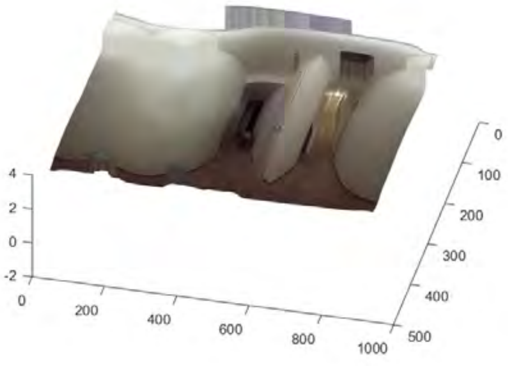









4
3
2
1
0
0
200
400
600
800
1000
0
200
400
600


4
2
0
-2
0
200
400
600
800
1000
0
100
200
300
400
500