Project 3 & 4: Stereo Vision-Based 3-D Reconstruction Using Feature Detection and Disparity Mapping

Matteo De Angelis
College of Engineering
University of Miami
Miami, USA
mld163@miami.edu

Carter Falkenberg
College of Arts and Sciences
University of Miami
Miami, USA
ckf25@miami.edu

Kelly Rojas
College of Engineering
University of Miami
Miami, USA
knr20@miami.edu

***Abstract* — This project focuses on processing calibrated and rectified stereo images to detect and match features, calculate disparities for matched features, and reconstruct a 3D representation of the picture's scene. The process consists of four main tasks: feature detection and collection using multiple techniques (Harris, SIFT, SURF) combined with non-maximum suppression to reduce redundancy, feature matching, disparity computations and disparity growing, and depth estimation for 3D reconstruction. The results show accuracy in depth estimation for well-detected features but also difficulties in generating high density of features while maintaining accuracy.**

Keywords — depth estimation, disparity map, feature matching, feature detection, stereo vision, 3-D reconstruction.

I. Introduction

Stereo vision is a technique in computer vision that utilizes two cameras on a fixed plane and consistent heights (which can be further aligned during rectification); this allows for depth estimation through the use of the camera's intrinsic parameters, the known distance between the cameras, and disparity calculations in the resulting images. This technology has applications in many fields, such as augmented reality and medical imaging [1, 2]. This project focuses on the process of converting two stereo images of the same scene into a 3D representation of the scene.

Feature detection and matching are crucial in stereo vision because they allow for correspondence estimation between left and right images, which is needed for estimations of depth. Techniques such as Harris corner detection, Scale-Invariant Feature Transform (SIFT), and Speeded-Up Robust Features (SURF) have been used extensively for robust feature extraction and matching [3, 4]. Non-maximum suppression is also used when detecting features and combining features, which reduces redundant features by choosing the strongest features in a set range. Different techniques exist for increasing number of features, with local disparity growing being a simple solution that iteratively expands in the neighborhood of feature matches to find new matches in that neighborhood.

Using matched features, disparity maps and 3-D reconstructions can be created, for dense representations of a given scene captured by stereo cameras.

Ten pair of images of a lab scene were used for this project. Ten taken from a left point of view and ten from a right point of view at different distances from the subjects in the picture. The pictures given were already rectified and of size 1252 pixels wide and 904 pixels height.

The original calibration parameters (before rectification for original stereo pairs) were the following:

$$C_{left} = \begin{pmatrix} 1361 & 0 & 606.4 \\ 0 & 1365.4 & 518.3 \\ 0 & 0 & 1 \end{pmatrix}$$

$$C_{right} = \begin{pmatrix} 1360 & 0 & 629 \\ 0 & 1361.9 & 462.7 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R = \begin{pmatrix} 0.9999 & 0.0112 & -0.0121 \\ -0.0110 & 0.9998 & 0.0158 \\ 0.0123 & -0.0157 & 0.9998 \end{pmatrix}$$

$$T = \begin{pmatrix} -259.24 \\ 3.95 \\ -1.41 \end{pmatrix} [mm]$$

The calibration parameters after rectification:

$$C_{left} = \begin{pmatrix} 1277 & 0 & 623.2 \\ 0 & 1277 & 490.4 \\ 0 & 0 & 1 \end{pmatrix}$$

$$C_{right} = \begin{pmatrix} 1277 & 0 & 616.8 \\ 0 & 1277 & 490.4 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$T = \begin{pmatrix} -259.3 \\ 0 \\ 0 \end{pmatrix} [mm]$$

II. Theory

A. Stereo Vision and Depth Perception

Stereo vision is a method that imitates the human visual system to perceive depth. It allows for analysis of disparities between two slightly different images taken from different points on a consistent plane and scan line. A given disparity is the difference in x (horizontal coordinate) between two matched features in the left and right images. Objects at different depths will have different disparities. The depth of points can be estimated using their disparities if other parameters are known (focal length, distance between cameras) [5].

B. Feature Detection and Matching

Feature detection is significant in stereo matching because it identifies key points in an image which are distinctive and can be matched easily to its corresponding location in another image. The feature detection methods used include the following:

1) *Harris corner detector*
   - Identifies areas with significant intensity variations.
2) *Scale Invariant Feature Transform*
   - Extracts key points based on the differences of Gaussian at multiple scales [6].
3) *Speeded Up Robust Features*
   - Faster alternative to SIFT, and which is based on integral images and the Haar wavelet responses [7].

C. Disparity Computation and Depth Estimation

Disparity refers to the pixel shift between the corresponding points in the left and right images, which is computed through the following equation:

$$d^* = x - x'$$

where x and x' are the x-coordinates of a matched feature point in the left and right images of a scene, respectively. Furthermore, the depth Z is determined using the disparity equation shown below:

Formula 1: Calculation of Z/depth

$$Z = f\frac{|B|}{d^*}$$

where f is the camera's focal length, B is the baseline distance between the two cameras, and d^* is the computed disparity [8]. Given a fixed focal point and baseline distance, larger disparities correspond to closer objects, whereas smaller disparities correspond to further objects.

D. Local Disparity Growing

Local disparity growing is performed in order to achieve a denser feature map between two images. This technique goes through each feature (in the left) and grows outwards, for each neighbor, a corresponding set of potential matches are checked in the image taken from the right, with the lowest sum of squared difference (SSD) being chosen (or no matches chosen if SSD exceeds a cutoff threshold).

E. 3D Reconstruction

After the disparity values are obtained for the matched features, a 3D point cloud representation of the scene is built. With Z already calculated, X and Y can be calculated as well (by converting pixels to mm) as shown below:

Formula 2: Calculation of X/Y

$$X = \frac{(x - c_x) \times Z}{f_x}$$

and similarly for Y. As a result, visualization of the 3D reconstructed environment can be plotted.

III. Methodology

A. Task 1 – Multimethod Feature Detection for Stereo Image Analysis

In this task, key point detection and feature extraction were performed on rectified stereo images using multiple feature detection methods. The workflow involved feature extraction from given images, merging features from different detectors, applying non-maximum suppression, and visualizing the selected key points. These images served as the basis for feature extraction and analysis. The pixel area parameter was set to control the suppression of weak features in the later steps.

To extract robust features from the images, three feature detection algorithms were used: Scale-Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF), and the Harris Corner Detector. These functions are available directly in MATLAB. The three algorithms were applied to each image independently, each having non-maximum suppression run on them after calculation. After the features from SIFT, SURF, and Harris detectors were obtained, they were merged into one feature matrix. Once again, non-maximum suppression was run on the combined features, reducing redundancy, keeping the most significant features only.

B. Task 2 – Feature Matching Across Stereo Images for Correspondence Estimation

In this task, feature matching was performed between the rectified stereo image pairs to establish correspondences for disparity calculations. Features from Task 1 were used as an input to Task 2.

The detected features from Task 1 were used to extract feature descriptors. These descriptors allow for matching between two sets of features. For the match threshold parameter, 10 was found to be best performing, controlling the maximum allowable matching distance as a percentage to the strongest feature detected. Furthermore, the max ratio was set to 0.6, controlling the rejection of ambiguous matches by requiring the ratio of the best match and the second-best match to exceed the parameter value. Moreover, a match was rejected if the difference in y values was greater than 10, which greatly reduced erroneous matches not on the same scan line. Finally, matches that included an x value smaller than 400 on a rectified image taken from the left or an x value greater than 852 on a rectified image taken from the right were discarded. This was done to prevent any erroneous matches between features detected outside of the scope of either one of the images respective to the other. For example, any pixel on the image taken from the right with a x-coordinate greater than 852 is not seen by the image taken on the left, therefore any feature detected in these regions should not be considered for matching. Afterward, the index pairs corresponding to the matched features were extracted.

C. Task 3 – Disparity Estimation and Disparity Growing

In Task 3, disparity of matched features was calculated. Additionally, local disparity growing was implemented to

create a denser disparity map between an image pair. Calculating disparity of features was simple, as it is just the difference in the x values for each match. For local disparity growing, each match contributed to the generation of new matches in an iterative process. For a given match, all neighbors were visited. For a given neighbor value, a set of possible matches was created in the image taken from the right (corresponding to a $3x3$ area in the image taken from the right defined by the disparity of the neighbor's corresponding feature). SSD (Sum of Square Difference) was then calculated for all potential matches, with the lowest SSD being chosen as a match, so long as the SSD was below the maximum threshold (which was found 250 to work well. This was roughly the average SSD of the features identified in Task 2). Then, the new feature was added to the feature set and growing could subsequently be done on it as well.

D. *Task 4 – Depth Estimation and 3-D Reconstruction from Stereo Disparity*

Task 4 required the calculation of depth and 3-D points for each feature. First, depth in mm was calculated (i.e. 'Z') using Formula 1 identified in the 'Theory' section. The focal point and distance between cameras were given, and the disparities were used from Task 3 calculations. From this, it was possible to calculate X and Y in mm, as defined by Formula 2. Two ways were used to plot the results, X vs Y with colors corresponding to depth, as well as a 3-D plot of (X, Y, Z).

IV. Results

A. *Task 1*

Results for Task 1 can be seen in Figures A-1 through A-10. The plots show the detected features when combining SIFT, SURF, and Harris feature detections, while also applying non maximum suppression. Many features appear in the monitor, laptop, water bottle, and checkerboard. This makes sense as these are distinct locations in the image, where plain objects may contain less information to generate a strong feature; for example the background cabinets.

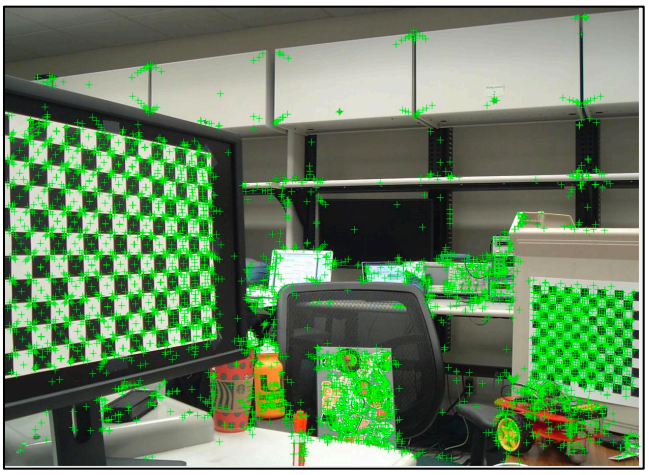

Figure A-1. Features detected from scene 1 with the image taken from the left (top 5000 strongest features).

B. *Task 2*

Results for Task 2 can be seen in Figures B-1 through B-10. The plots draw lines between matched features in the left and right images of a scene. There is a contrast between Task 1 in that many of the features on the monitor are not matched anymore, since they get cut off in the right image and also may be too similar to matched features in the right image. However, those features that are matched are very accurate, indicating strong feature matches.

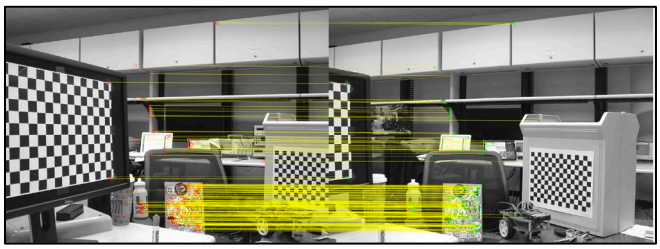

Figure B-1. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 1.

C. *Task 3*

Task 3 results are shown in Figures C-4 through C-13 in the Appendix. Similar to Task 2 results, yellow lines are drawn between matched features. The number of drawn features is vastly increased, which can be seen in the thickness of the yellow lines. It is difficult to see which points correspond to which, A check was performed by picking random features from a random image to display to ensure matched features were accurate in a visual manner. Some examples can be seen in Figure C-1, Figure C-2, and Figure C-3.

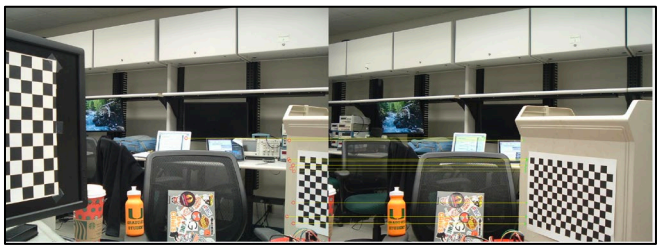

Figure C-1. Example taken from the first iteration of the check.

Figure C-2. Example taken from the second iteration of the check.

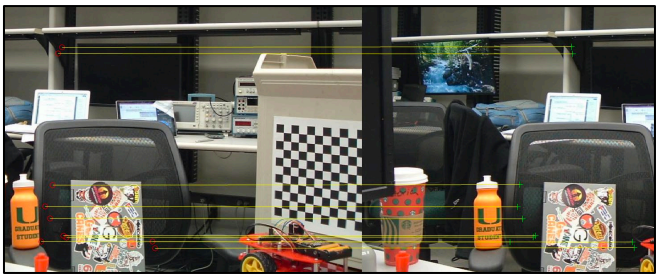

Figure C-3. Example taken from the third iteration of the check.

D. Task 4

Results for 3D mappings for scene 1 is shown in Figure D-1, and divided in.Figure D-1, Figure D-1B, and Figure D-1C.
.

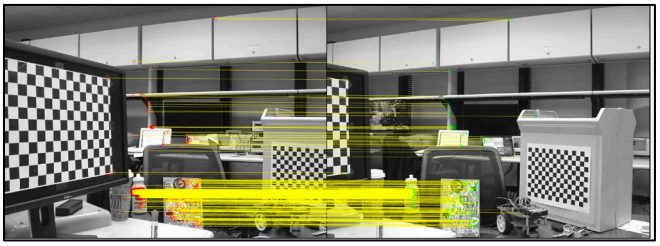

Figure D-1A. Reference to scene 1 matched features from image taken from the left and image taken from the right.

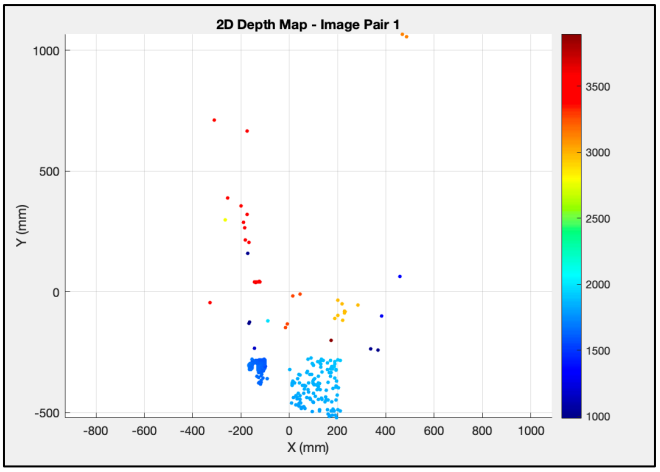

Figure D-1B. Color mapped 2D depth map for scene 1 matched points.

As seen in Figure D-1B, the water bottle cluster is marked as closest (dark blue), the laptop cluster next (light blue), and the laptop in the background cluster (yellow), and then the rest of the points are marked further. However, the feature matching was not dense enough to recognize certain shapes and objects such as the monitor, cabinets in the back, the chair, and more. It was decided to not sacrifice accuracy for more matches, since errors largely increased with more matches (through techniques such as increasing allowed SSD in disparity growing or finetuning feature detection parameters).

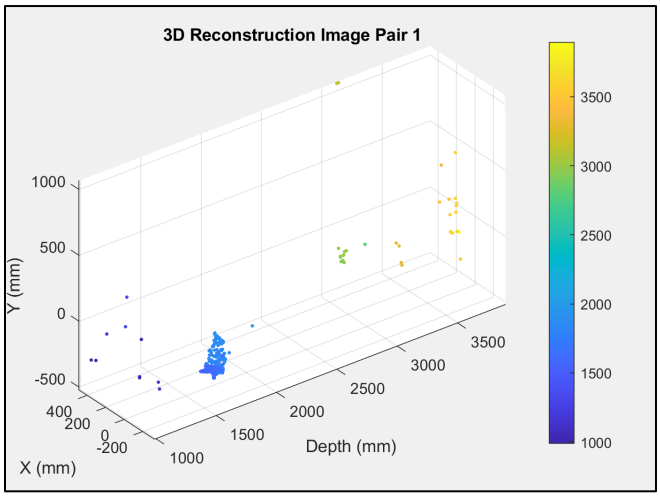

Figure D-1C. Color mapped 3D map for scene 1 matched points.

A 3D map was also plotted using real depth values to visualize better the depth. Noticeably, the 3D maps show a high accuracy in determining the distances between the objects but low precision. An example can be seen in Figure D-1C, where generally the objects are detected with a correct hierarchy of distances (closest object can be seen detected as closest in the image as well and so on until the furthest), however there are some sparse points that do are not assigned to the correct object.

V. Conclusion

Overall, the project was a success. The combination of different techniques for feature detection proved effective and enabled efficient extraction and matching of features across stereo images, allowing accurate disparity calculations. The 3D depth maps show however the unreliability of using exact values found from the disparity matches. Even though, the objects are generally correctly detected at their depth distances relative to each other, the true values of the depth of these objects are inaccurate.

One reason for this inaccuracy could be coming from the picture taken themselves. Albeit the calibration of the rectified pictures shows that the vertical coordinate of the principal point is identical ($c_y = 490.4$), meaning that the pair of images taken of the same scene are aligned horizontally and lie on the same scan line, when performing feature detection it was clear that this was not the case. There was a constant tendency that can be seen from the images of the feature matches of the picture taken from the right of the scene had a slightly lower y value. This alone demonstrates that the features detected were not perfectly aligned, which could introduce errors in the disparity calculations.

Another important mention that could be done for future improvements on the project is about the selection of the features. In fact, the features were detected first using MATLAB built-in functions [9][10], and discarded later according to the various checks to only keep accurate features. This is inefficient as giving 'guidelines' to the feature detection algorithms beforehand onto how, or better where, to look for features directly can lead to better results.

Other feature detection algorithms could have been implemented to better analyze the results but, despite the fact that the depth values are not very accurate, the results outcome is satisfying as it shows consistency onto understanding which objects are closer to the camera and which are further apart.

APPENDIX

A - Task 1

Figures A-2 through A-10 correspond to the identified features for images 2-10 (left images, top 5000 strongest features).

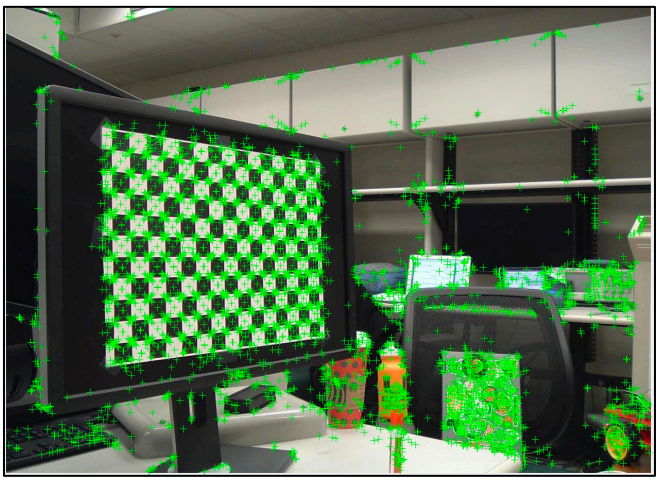

Figure A-2. Features detected from scene 2 with the image taken from the left (top 5000 strongest features).

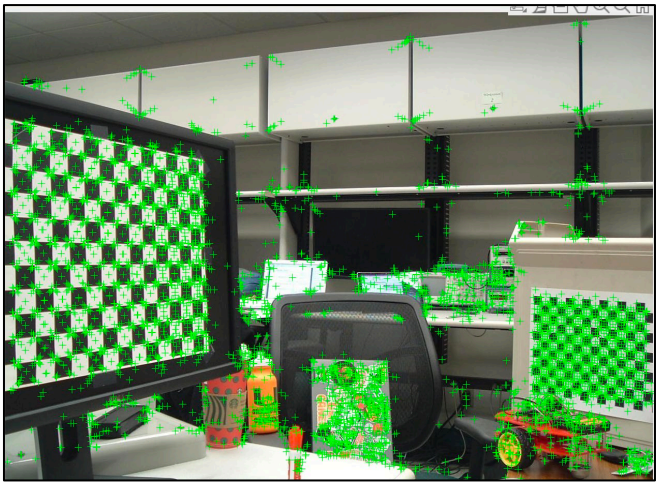

Figure A-3 Features detected from scene 3 with the image taken from the left (top 5000 strongest features).

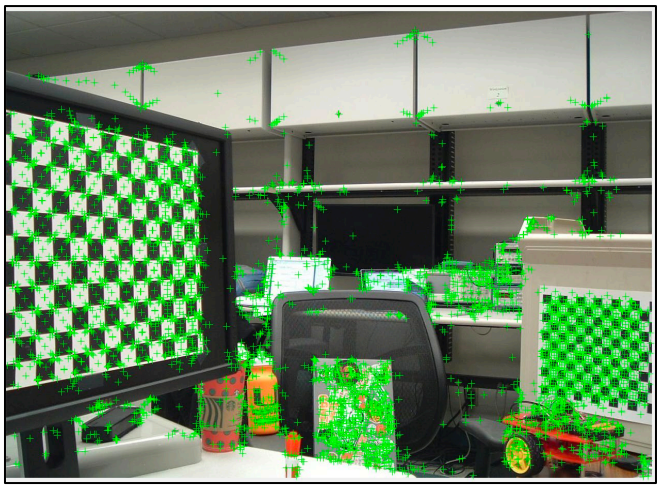

Figure A-4. Features detected from scene 4 with the image taken from the left (top 5000 strongest features).

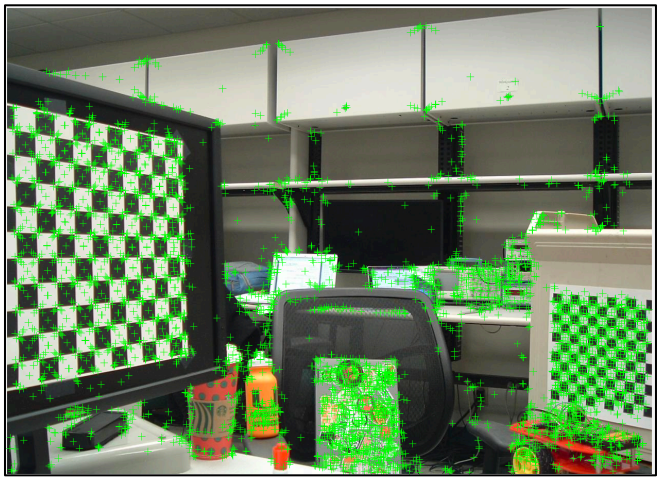

Figure A-5. Features detected from scene 5 with the image taken from the left (top 5000 strongest features).

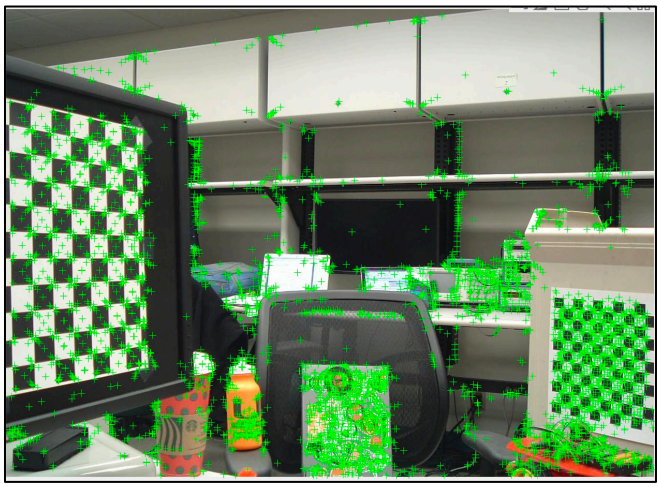

Figure A-6. Features detected from scene 6 with the image taken from the left (top 5000 strongest features).

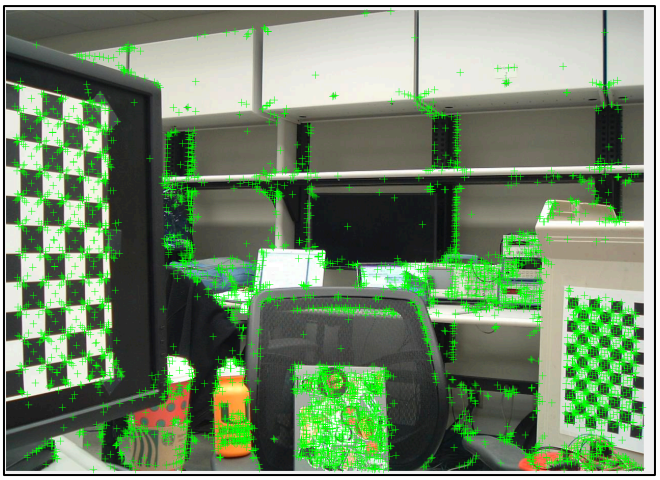

Figure A-7. Features detected from scene 7 with the image taken from the left (top 5000 strongest features).

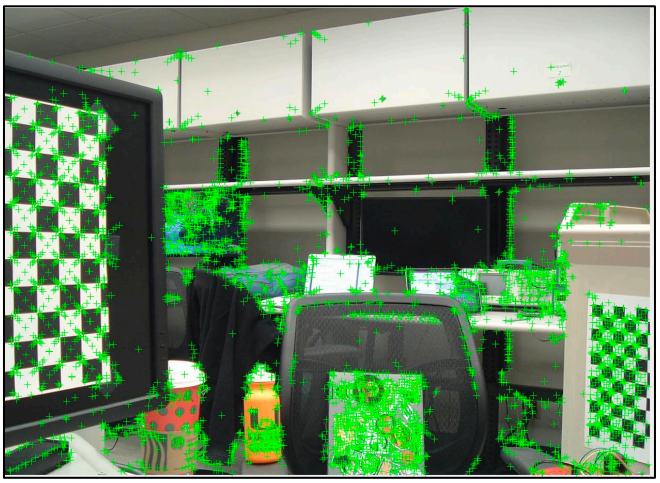

Figure A-8. Features detected from scene 8 with the image taken from the left (top 5000 strongest features).

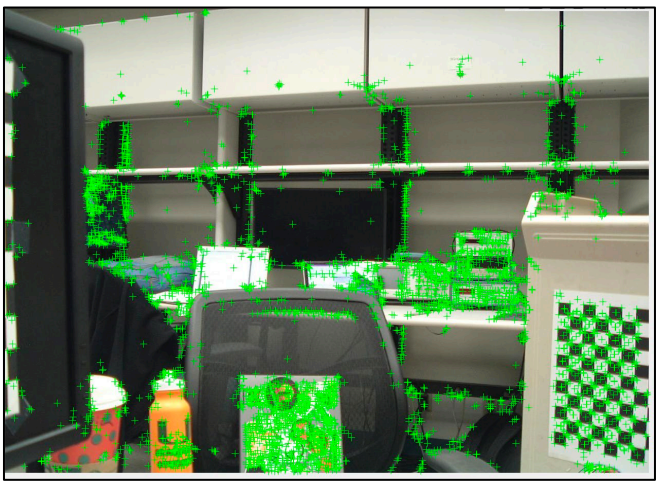

Figure A-9. Features detected from scene 9 with the image taken from the left (top 5000 strongest features).

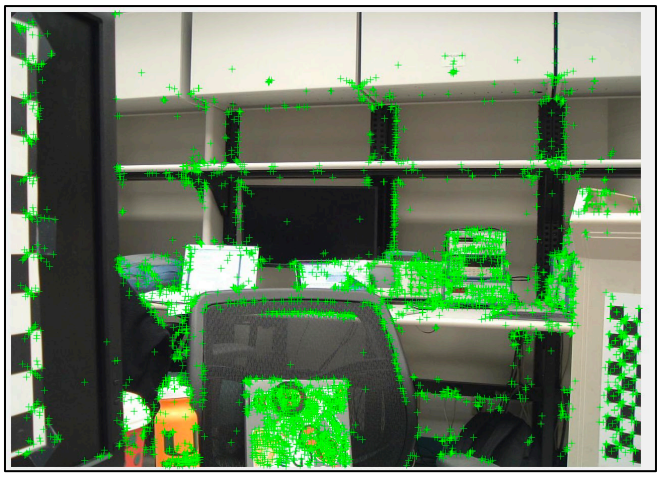

Figure A-10. Features detected from scene 10 with the image taken from the left (top 5000 strongest features).

B - Task 2

Figures B-2 through B-10 correspond to the matched features for images 2-10.

Figure B-2. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 2.

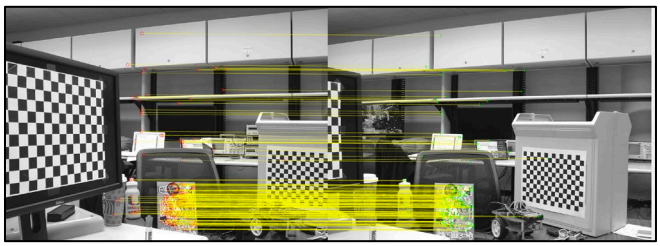

Figure B-3. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 3.

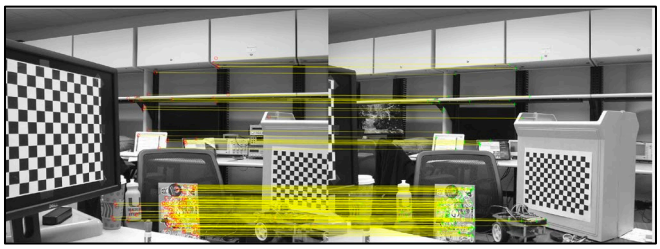

Figure B-4. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 4.

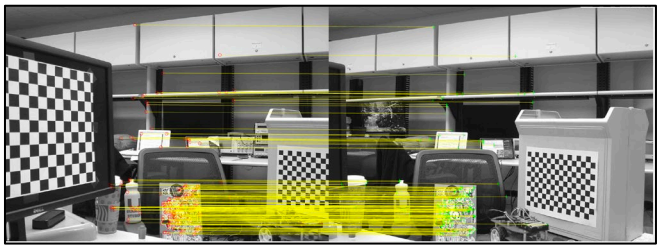

Figure B-5. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 5.

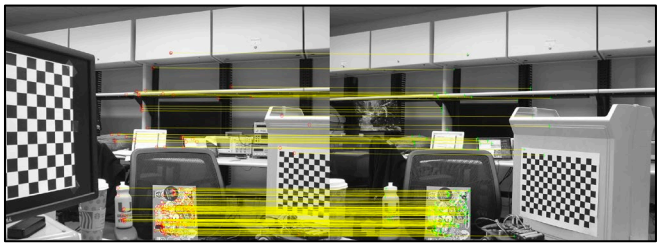

Figure B-6. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 6.

Figure B-7. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 7.

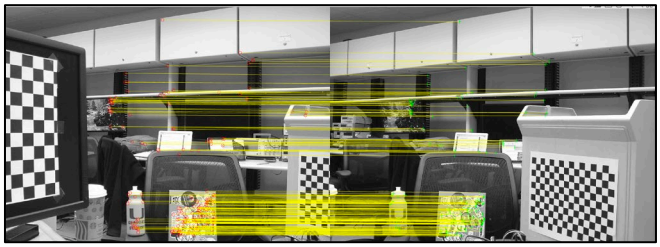

Figure B-8. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 8.

Figure B-9. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 9.

Figure B-10. Final feature matches retrieved between the first pair of images taken from the left and the right at the same distance in scene 10.

C - Task 3

Figures C-4 through C-13 correspond to the disparity growing matched features for images 1-10.

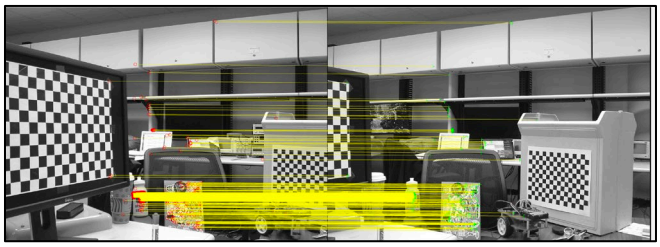

Figure C-4. Disparity growing of matched features in scene 1.

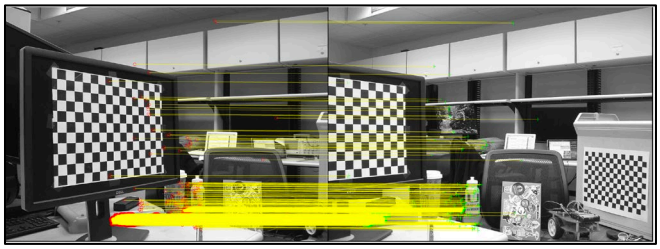

Figure C-5. Disparity growing of matched features in scene 2.

Figure C-6. Disparity growing of matched features in scene 3.

Figure C-7. Disparity growing of matched features in scene 4.

Figure C-8. Disparity growing of matched features in scene 5.

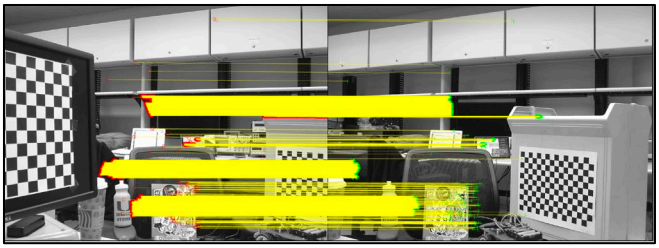

Figure C-9. Disparity growing of matched features in scene 6.

Figure C-10. Disparity growing of matched features in scene 7.

Figure C-11. Disparity growing of matched features in scene 8.

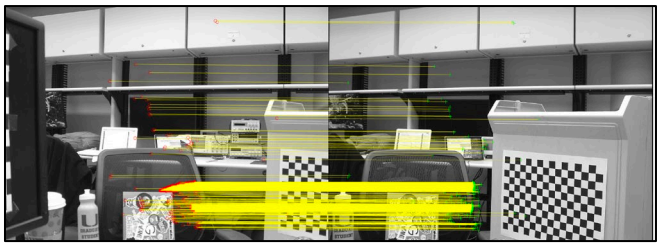

Figure C-12. Disparity growing of matched features in scene 9.

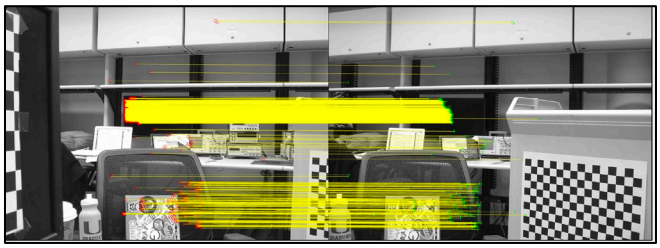

Figure C-13. Disparity growing of matched features in scene 10.

D - Task 4

Figures D-2 to Figure D-10 display the mapped depth of the matched features in images 2 through 10.

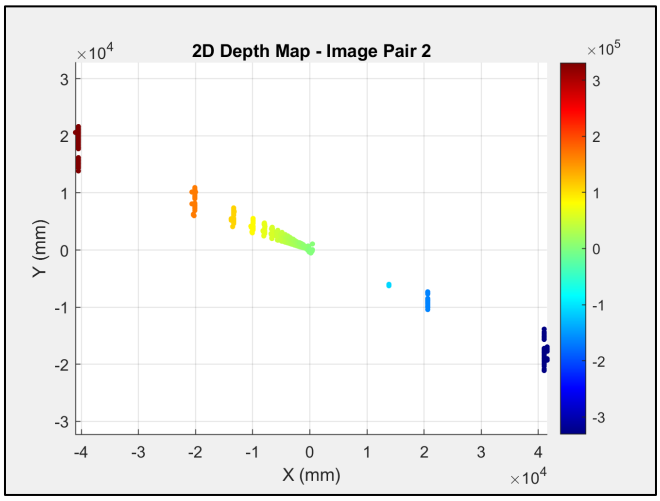

Figure D-2B. Color mapped 2D depth map for scene 2 matched points.

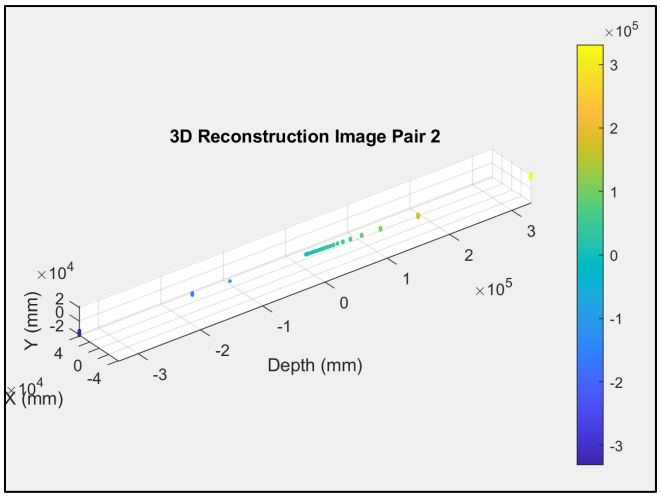

Figure D-2C. Color mapped 3D map for scene 2 matched points.

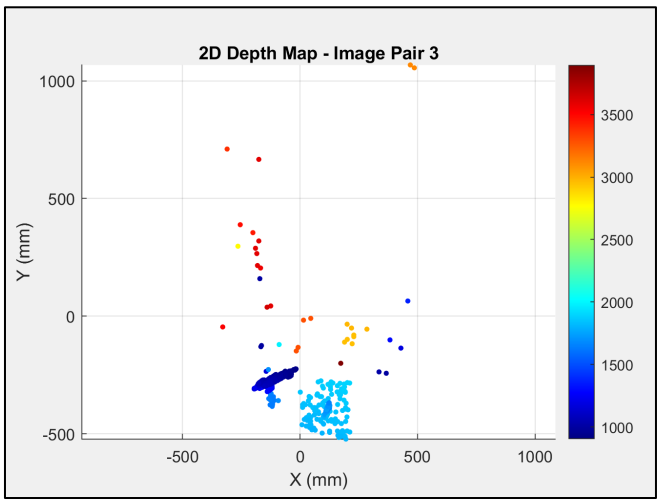

Figure D-3B. Color mapped 2D depth map for scene 3 matched points.

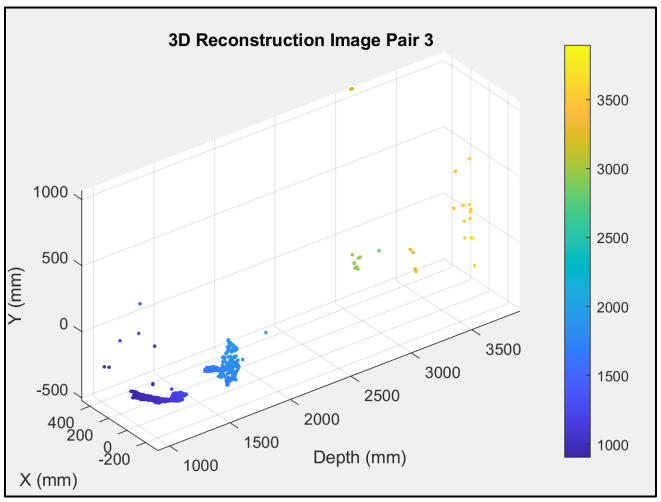

Figure D-3C. Color mapped 3D map for scene 3 matched points.

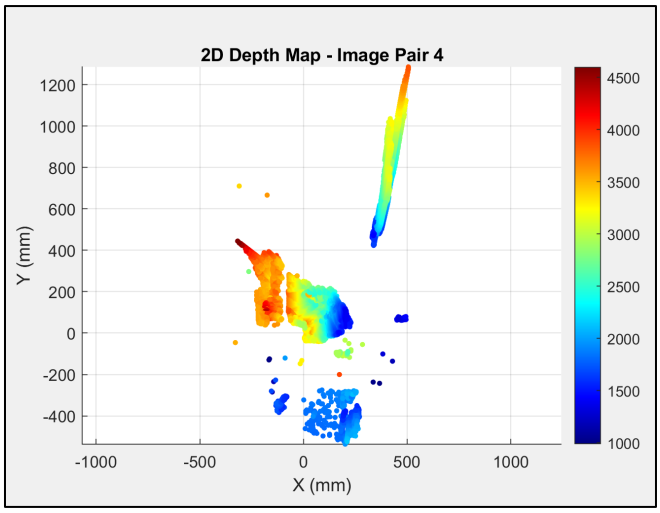

Figure D-4B. Color mapped 2D depth map for scene 4 matched points.

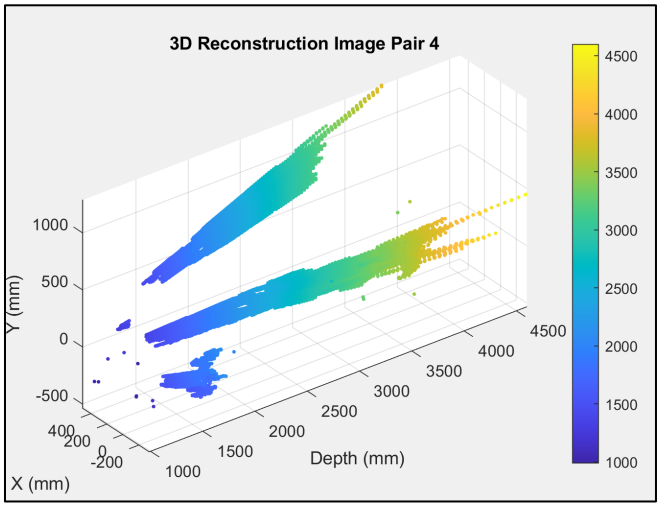

Figure D-4C. Color mapped 3D map for scene 4 matched points.

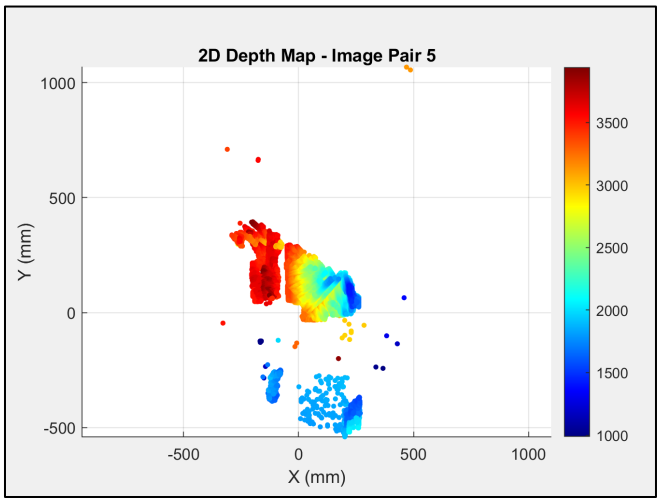

Figure D-5B. Color mapped 2D depth map for scene 5 matched points.

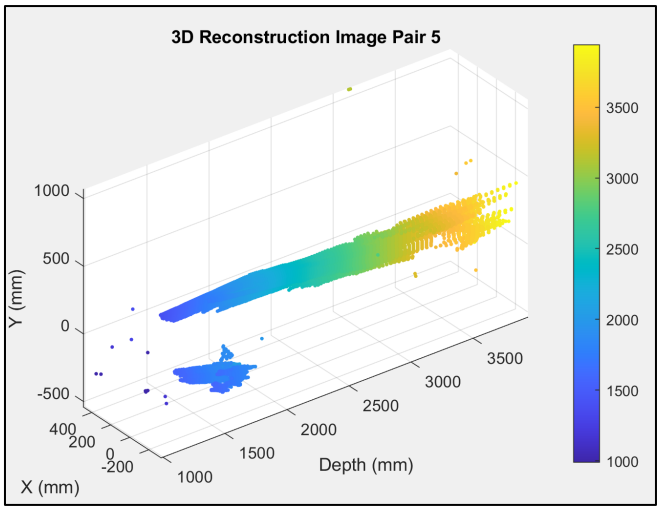

Figure D-5C. Color mapped 3D map for scene 5 matched points.

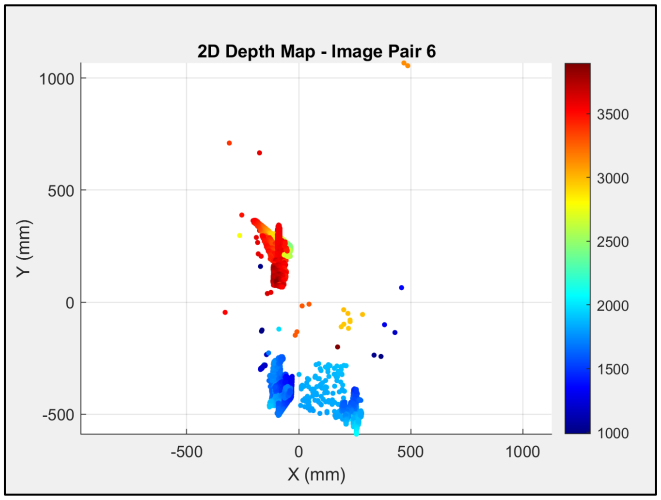

Figure D-6B. Color mapped 2D depth map for scene 6 matched points.

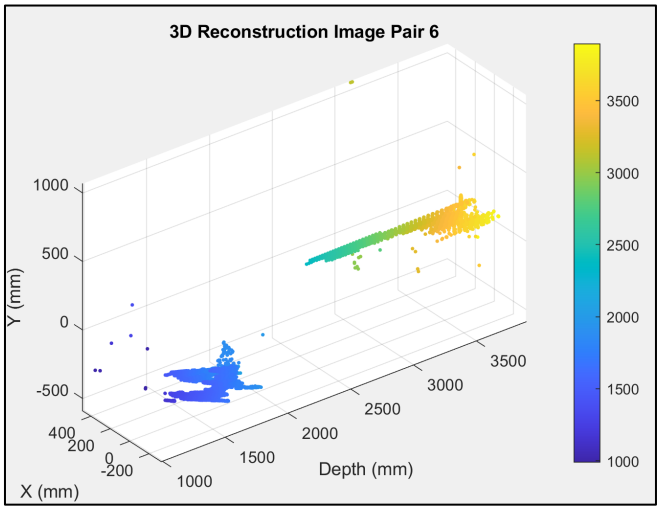

Figure D-6C. Color mapped 3D map for scene 6 matched points.

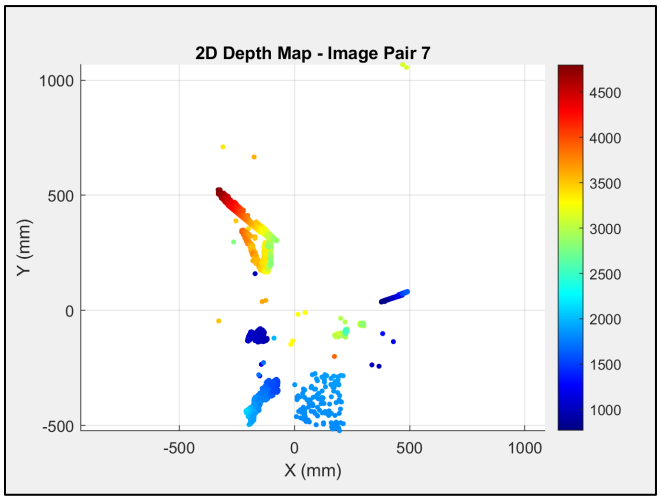

Figure D-7B. Color mapped 2D depth map for scene 7 matched.

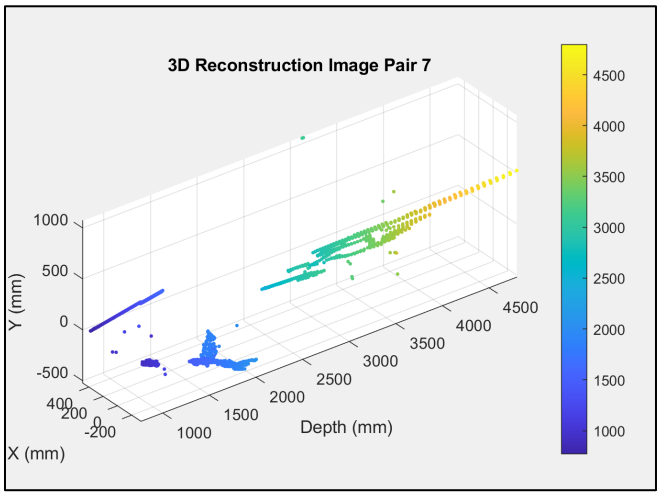

Figure D-7C. Color mapped 3D map for scene 7 matched points.

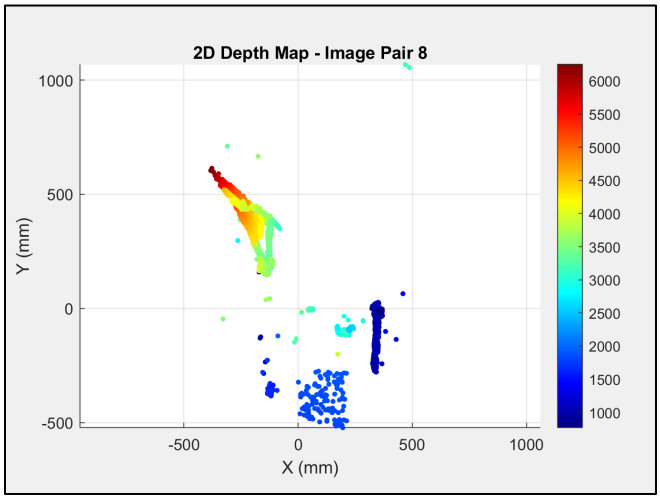

Figure D-8B. Color mapped 2D depth map for scene 8 matched points.

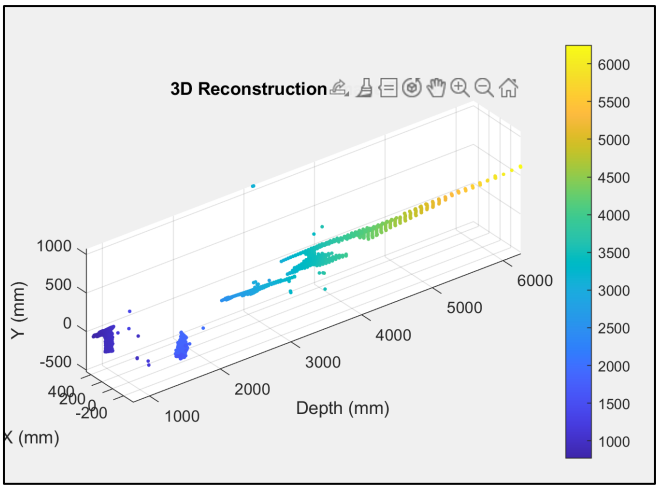

Figure D-8C. Color mapped 3D map for scene 8 matched points.

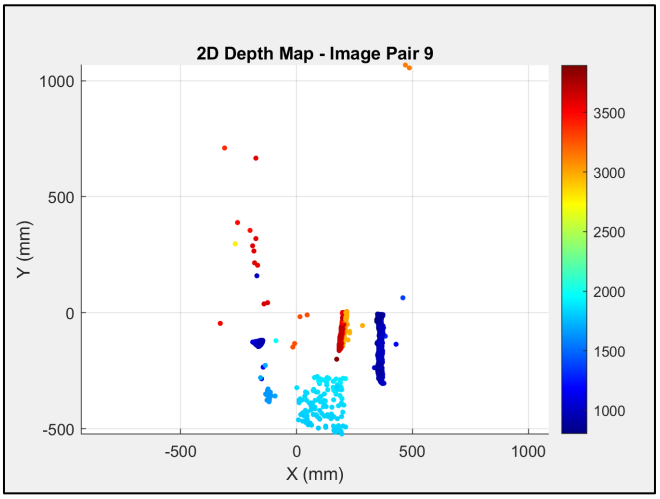

Figure D-9B. Color mapped 2D depth map for scene 9 matched.

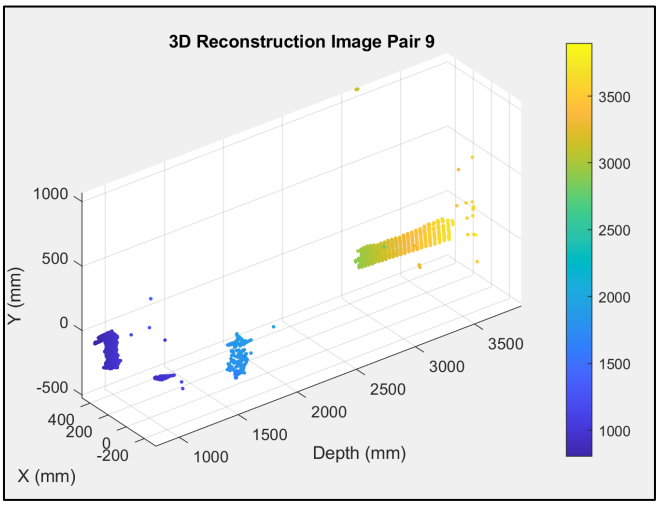

Figure D-9C. Color mapped 3D map for scene 9 matched points.

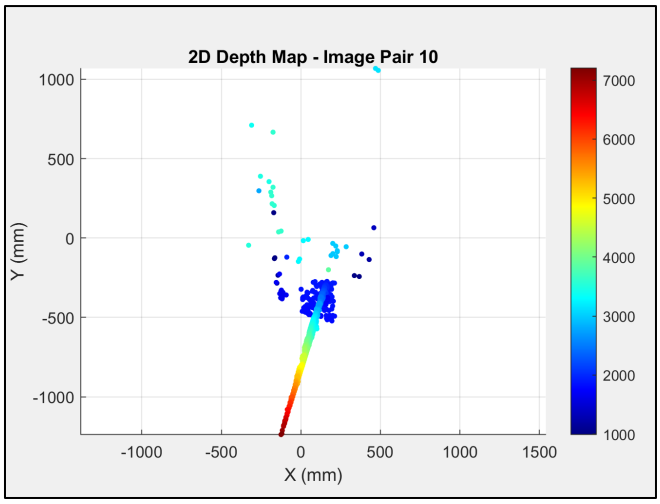

Figure D-10B. Color mapped 2D depth map for scene 10 matched.

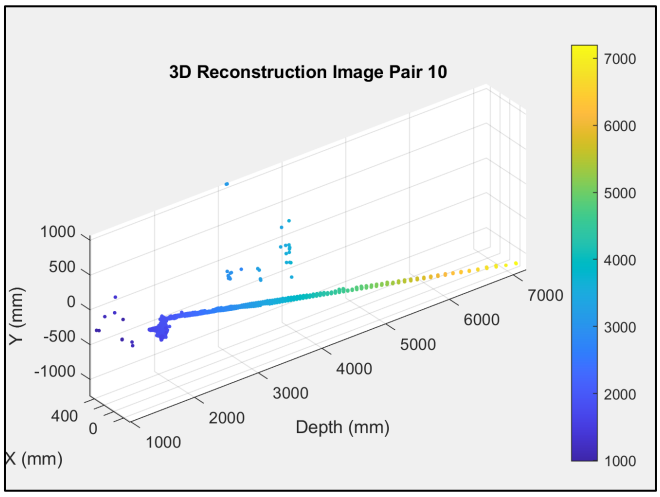

Figure D-10C. Color mapped 3D map for scene 10 matched points.

REFERENCES

[1] R. Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed. Cham, Switzerland: Springer, 2022. [Online]. Available: https://library.huree.edu.mn/data/202295/2024-06-03/Computer%20Vision%20-%20Algorithms%20and%20Applications%202nd%20Edition%2C%20Richard%20Szeliski.pdf

[2] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, Apr. 2002. [Online]. Available: https://vision.middlebury.edu/stereo/taxonomy-IJCV.pdf

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004. [Online]. Available: https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Graz, Austria, May 2006, pp. 404–417. [Online]. Available: https://people.ee.ethz.ch/~surf/eccv06.pdf

[5] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed. Cambridge, UK: Cambridge University Press, 2004. [Online]. Available: https://www.robots.ox.ac.uk/~vgg/hzbook/

[6] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004. [Online]. Available: https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf

[7] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 404-417. [Online]. Available: https://people.ee.ethz.ch/~surf/eccv06.pdf

[8] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, 1st ed. Sebastopol, CA: O'Reilly Media, 2008, pp. 377-380. [Online]. Available: https://docs.opencv.org/3.4/dd/d53/tutorial_py_depthmap.html

[9] Mat1 MathWorks, "extractFeatures," *MathWorks Documentation*, Accessed: Mar. 11, 2025. [Online]. Available: https://www.mathworks.com/help/vision/ref/extractfeatures.html

[10] MathWorks, "matchFeatures," *MathWorks Documentation*, Accessed: Mar. 11, 2025. [Online]. Available: https://www.mathworks.com/help/vision/ref/matchfeature.html