Geometric Stereo Increases Accuracy of Depth Estimations for an Unmanned Air Vehicle with a Small Baseline Stereo System

Eleanor Tursman (Grinnell College, Physics), *SUNFEST Fellow*

Camillo Jose Taylor, CIS

***Abstract*— A small stereo camera is a light and economical solution for obstacle detection and avoidance for unmanned air vehicles (UAVs). It is possible to create depth maps of a scene given a pair of stereo frames from such a camera. However, the smaller and lighter the stereo camera, the smaller its baseline, which in turn limits its ability to discriminate objects that are farther away. Maximizing the effective range of the stereo setup is essential for real-time applications, where quick decisions need to be made to avoid obstacles approaching the UAV. To overcome this difficulty, we use knowledge of the camera's position over time to mimic large baseline disparity calculations, a technique we have dubbed "geometric stereo." This paper outlines a simulation that shows that this technique is able to obtain better results for depth estimation with smaller confidence intervals than those obtained by simply averaging discrete depth maps over time.**

***Index Terms*— Geometric stereo, Obstacle detection, Robotics, Simulation, Small baseline stereo, Stereo vision UAV**

I. INTRODUCTION

Unmanned air vehicles (UAVs) are used in both military and civilian contexts, aiding in tasks from search and rescue missions to structural inspections of power lines and bridges [1][2]. In order to effectively accomplish these tasks, UAVs must be able to adapt their flight paths in 3D space in order to avoid detected obstacles in both static and dynamic environments. To meet real-time application goals, it is necessary for the UAV to meet size, weight, and power (SWAP) constraints. Both laser sensors (LiDAR) and cameras have been used by UAVs capable of working in real-time. However, while obstacle detection using LiDAR has proved successful for both ground and air based vehicles [4], LiDAR cannot meet SWAP constraints. One example of a small commercially available LiDAR which could be mounted on a UAV is the Hokuyo UTM-30LX. It weighs 370 grams, consumes 12V at 0.7amps, and is 60x60x87mm. Though the laser is precise, using LiDAR for real-time applications on a small UAV cannot be ideal due to these limitations. On the other hand, stereo cameras are smaller, lighter, and consume much less power, making them more attractive for this task.

Computer vision is a field that focuses on enabling machines to mimic the high-level perceptive power of humans. A stereo camera, paired with computer vision algorithms, can therefore provide a sensor that better meets SWAP constraints while accomplishing real-time tasks. Through the combination of stereo and visual odometry, two vision techniques, we hope to maximize how quickly a UAV can move while accurately avoiding obstacles. Since we wish to minimize SWAP constraints, our stereo camera must be as small as possible. This caveat means that the baseline, or the distance between the two lenses, limits the distance the camera can "see." There is an inherent trade-off between the baseline and accuracy of a stereo camera. To make depth measurements of a scene, a stereo algorithm must correctly match points between the left and right views from each lens. While a larger baseline stereo camera can detect objects farther away, it also makes it more difficult to match points correctly, which lowers the accuracy of depth results. We therefore use a stereo camera with a small baseline, for its greater accuracy, lighter weight, smaller size, and smaller power consumption.

We show through simulation that it is possible to compensate for the short range of a small baseline stereo camera by integrating knowledge of camera pose with our images to mimic the results of a larger-baseline stereo system. By then integrating this system onto an onboard graphics processing unit (GPU) for parallel processing of stereo feature matching, we can produce a UAV that better meets SWAP constraints, and is also able to more effectively build an accurate 3D representation of the

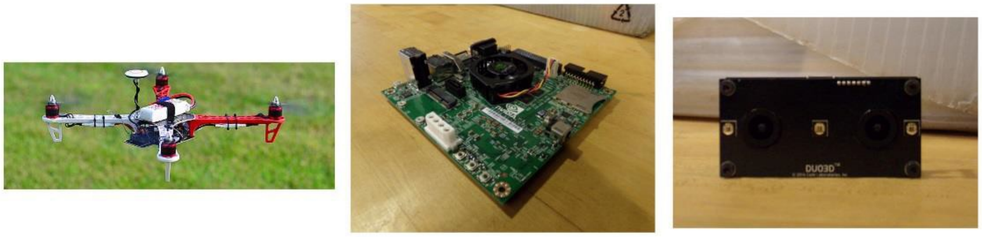

Figure 1– From left to right: a small quadcopter UAV (DJI F450), the Jetson TK1, and the DUO-MLX. The DUO-MLX has a small 30mm baseline, and a 2.0mm focal length.

world around it. This UAV can then be purposed towards accomplishing tasks such as search and rescue with improved speed and accuracy.

To accomplish this goal, we choose to use the DUO-MLX stereo camera, pictured in Figure 1. The simulation's parameters are based upon the specifications of this small stereo camera. This specific camera was chosen for being light at 12.5g, small at 52.02x25.40x13.30mm, for featuring a wide field of view (FOV) at 170 degrees, for only consuming 5V at 0.5amps, and for having a built-in inertial measurement unit (IMU). We will integrate the DUO-MLX with the small Jetson TK1 board, which has GPU processing capabilities, for speed.

II. BACKGROUND

2.1 Stereo Vision

Stereo vision is a tool used to gather depth information from a pair of left and right images, mimicking the binocular system of human vision. Once the camera is calibrated, the left and right images are rectified, a process that simplifies the geometry of the scene, and places matching features across the left and right images along the same horizontal lines. It is then possible to determine the physical depth of pixels in an image by calculating each pixel's disparity, the difference between the location of the pixel in the left image and the right image. To obtain disparity measurements, it is necessary to correctly match pixels between the left and right images. This hard problem has typically been solved using feature or featureless methods. For example, some researchers have employed the Small Vision System (SVS) stereo engine, which uses feature matching to generate depth maps [3]. Others have forgone features and instead use dense or semi-dense methods that make use of most or all of the pixels in each image [6][7][8]. In [8], researchers suggest comparing temporal and spatial gradients to quickly discern disparity measurements. These featureless methods typically rely upon exploiting camera motion to estimate where pixels have moved across image frames.

A sufficiently accurate depth map allows the UAV to determine which points are closest to the stereo camera in the scene, and therefore what obstacles must immediately be avoided. Stereo cameras have been used for obstacle detection in [1][2][3][4]. In section three, we will explore some of the basic mathematical details of stereo.

2.2 Visual Odometry

Visual odometry is the process of estimating the orientation and location of a camera based on the camera's video feed. With this tool, it is possible to determine the relative 3D locations of both points in images and the location of the camera, and therefore help an autonomous robot navigate through an unknown environment. Visual odometry typically locates and tracks features across image frames, and then uses their motion to calculate the essential or fundamental matrices, which through standard value decomposition (SVD) will produce the rotation matrix and translation vector that describe how the camera has shifted from one frame to the next. The essential matrix uses calibrated camera information to determine the rotation and translation information between frames. The fundamental matrix, which can be used with uncalibrated cameras, uses projective geometry to determine a line along which one point in the first image may be found in the second image [9]. However, more recent literature details a method for visual odometry that bypasses feature extraction entirely [6][7].

2.3 UAV Obstacle Detection

Several different methods have already been applied to UAV obstacle detection and avoidance, which use both monocular (single camera) and stereo (dual camera) systems. In [1], objects close to the stereo camera are isolated by segmenting depth maps. Another group combines frontal stereo vision with two side-facing fisheye cameras using optical flow to avoid both objects in front of and walls to the sides of the UAV [3].

Simpler monocular systems detect and track features in image frames from video feed, relying on optical flow to situate a sparse number of these features in 3D space. The features are then clustered based on their relative distance from one another, determining the areas for the UAV to avoid [5]. However, the current state-of-the-art in robotic obstacle avoidance arguably lies with SLAM (simultaneous localization and mapping), where a robot builds and navigates within a map of its unknown surroundings using monocular vision [6][7]. In [6], LSD-SLAM (large-scale direct monocular SLAM) sidesteps feature extraction, which can be a time-consuming process, and instead uses every pixel in every frame, performing optimization on actual images. The goal of this system is to construct a large-scale depth map over time by tracking the camera's position and estimating its current pose through a series of frames, then using these frames to imitate a stereo system and calculate depth maps. These depth maps are refined, and then integrated into the large-scale depth map, which is finally optimized. The camera's pose in 3D space is estimated by referencing the dense depth map. The system is capable of running in real-time, however, since it is ultimately based on monocular input, it will only be able to build a 3D model up to a scale factor. By using a stereo camera, we can mitigate this limitation, leading to accurately scaled representations of the 3D world.

Unlike the previous literature, we will implement geometric stereo in order to improve upon monocular SLAM. We will be integrating scaled stereo depth maps from a small baseline camera with visual odometry information in order to mimic large baseline stereo results while achieving real-time performance.

III. GEOMETRIC STEREO ALGORITHM

3.1 The Problem

We need to be able to compensate for the limited range of a small baseline camera, because its small size, weight, and power consumption is integral for optimizing our UAV. The equation to model a stereo setup can be determined by exploiting properties of similar triangles (see Fig. 2):

$$Z = \frac{Bf}{�} \quad (1)$$

where Z is the depth of the point in space, B is the baseline of the stereo system, d is the disparity, and f is the focal length of the cameras. We then add a pixel error � to this disparity calculation, resulting in

$$Z = \frac{f}{�\pm\delta} \quad (2)$$

Based on (1) and (2), there is an inherent trade-off between accuracy and baseline. As the baseline increases, the depth at which we are able to discriminate objects with a disparity of one pixel increases. Also, as the baseline increases, the error term � decreases relative to the constantly scaling signal term. However, there are limitations imposed by large-baseline stereo. The number of points shared by the left and right views decreases as the baseline increases, limiting the field of view of the system. In addition, the larger the baseline, the larger the range of disparities that must be searched through per pixel, which can drastically slow down the generation of depth maps.

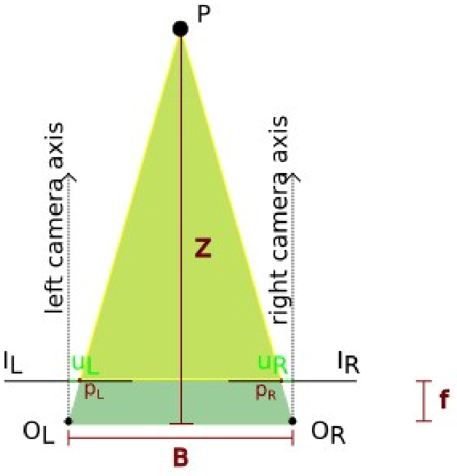

Figure 2– Similar triangle stereo setup. By using the properties of similar triangles, comparing the yellow and green triangles, we are able to derive the standard stereo equation. I_L and I_R are the left and right image planes, O_L and O_R are the left and right optical centers, B is the baseline, f is the focal length, P is the point in 3D space, Z is the depth, and u_L and u_R mark the pixel difference between where the ray to point P intersects each image plane and the camera axis. To set up the equation, simply let $\frac{}{Z} = \frac{B-�}{Z-f}$ and solve for Z, where $d = u_L - u_R$.

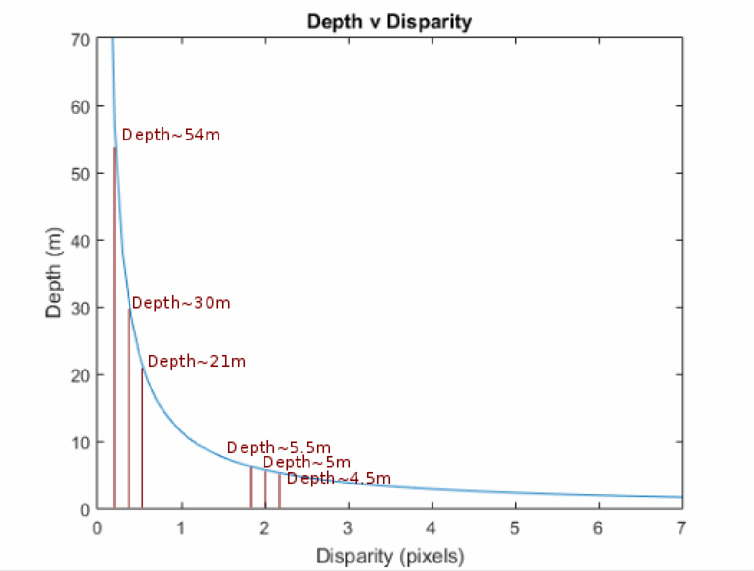

Figure 3– Depth v. disparity prediction for the DUO-MLX. The groups of three red lines represent a true depth measurement, and its possible disparity error to the left and right. Because of the small baseline, a sub-pixel disparity error could mean the difference between thinking a point is 30m away or 54m away.

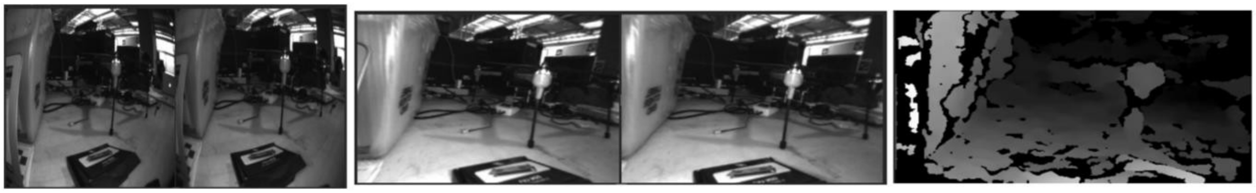

Figure 4– From left to right, the raw images of the DUO-MLX, the rectified images, and the disparity map. The images were captured with the SDK and dashboard provided with the DUO-MLX. The disparity map illustrates the good capabilities of the camera at ranges under one meter.

With the DUO-MLX, our small baseline stereo camera, measurements of points beyond approximately ten meters become very unreliable, since their accuracy heavily relies upon exact disparity calculations, which simply isn't realistic (see Fig. 3). In practice the stereo imagery we are able to obtain is displayed in Figure 4.

3.2 The Simulation

We propose circumventing these small baseline limitations by using our knowledge of camera translation over time in order to imitate large baseline stereo results using our small baseline images. The simulation we use to demonstrate this plan implements a simple stereo scenario, adds noise and bias to measurements to better imitate real-world data, then compares the accuracy of depth calculations and the size of confidence intervals, for both averaging and geometric stereo methods.

For the purposes of this paper, the term "averaging stereo" refers to taking the average of two small-baseline depth maps to crudely obtain a more accurate depth estimation of a given scene. Averaging stereo does not integrate camera position data into its depth estimations. In turn, the term "geometric stereo" refers to using knowledge of camera position to mimic large-baseline stereo with a small-baseline system by taking one image from one position and one image from the second position to calculate a more accurate depth map (see Fig. 5). We will show that geometric stereo provides superior depth estimation results.

3.3 Depth Error

We create a simple scenario where two stereo cameras are separated by a purely horizontal translation and both focus on one point in 2D x-z space. The points u_L and u where the point will be projected on the left and right image planes of each stereo system will be determined using

$$u_L = f\frac{x_p - x_c}{z}, u_R = f\frac{x_p - x_c - B}{z} \quad (3)$$

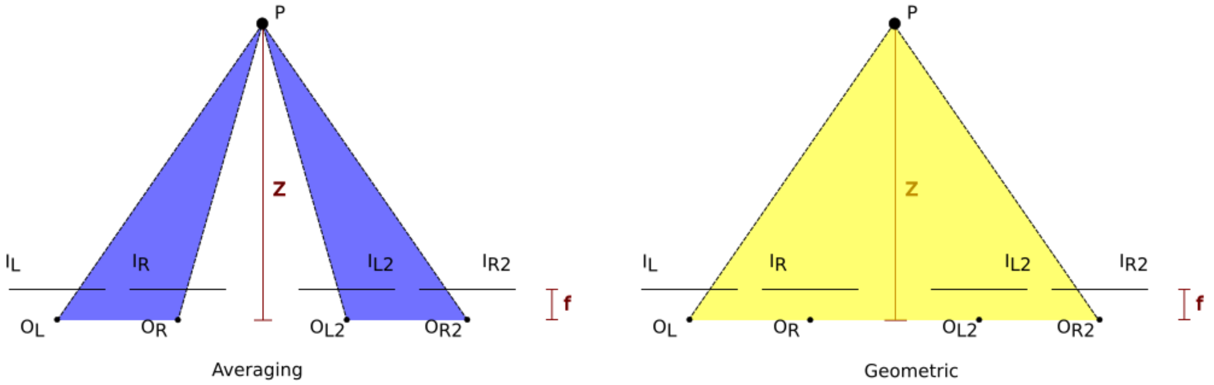

Figure 5– Averaging stereo takes two depth measurements from a stereo camera in two separate positions, while geometric stereo uses the outer image planes to mimic one large baseline stereo setup.

where (x_p, Z) is the location of the point in space, $(x_�, 0)$ is the location of the optical center of the left camera, $(x_� + B, 0)$ is the location of the optical center of the right camera, and f is the focal length of the cameras (see Fig. 6). We then add a noise term

and a bias term to both u_L and u_R, to get

$$\tilde{u}_L = u_L + � + �, \tilde{u}_R = u_R + � + � \qquad (4)$$

We add a random amount of noise � between

negative one and one pixel to each projection onto the image plane to better simulate pixel matching in a stereo matching algorithm. We also add a random bias � between zero and one pixel to each calculation

in order to avoid the unrealistic zero-means scenario, where we would theoretically be able to remove noise from our calculations by taking a large number of images of the same scene.

To simulate the averaging technique, we calculate depth value error by taking the depths calculated by (1) for the left and right lenses, subtracting the true depth from each calculated depth to get the error of each calculation, then averaging these error results together. To simulate the geometric technique, we use the far left and far right image planes to get our first depth estimate, then we subtract the true depth from our estimate to get our error. Because of the noise and bias, each resulting depth error will change for every iteration of the simulation, so we run the algorithm 1000 times, and then produce histograms of our results.

3.4 Confidence Intervals

Stereo depth confidence intervals stipulate a range of values that should contain the true depth for

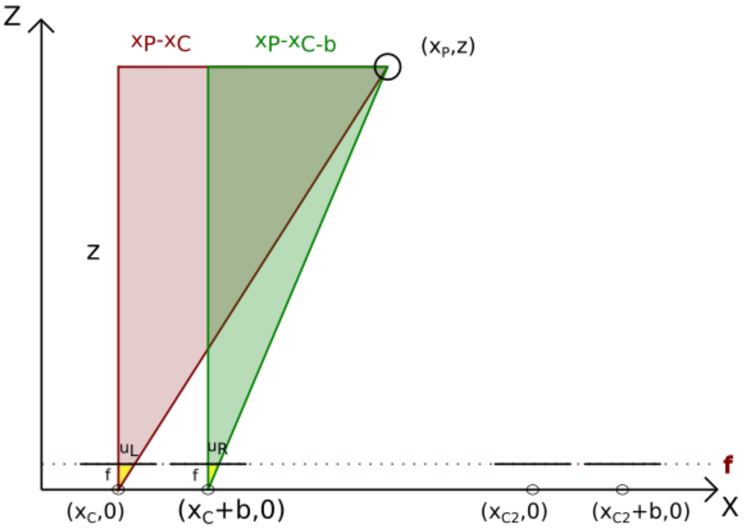

Figure 6– Simulation setup. We obtain (3) by once again comparing similar triangles. The red and yellow triangles can be used to solve for

u, and the green and yellow triangles can be used to solve for u_R.

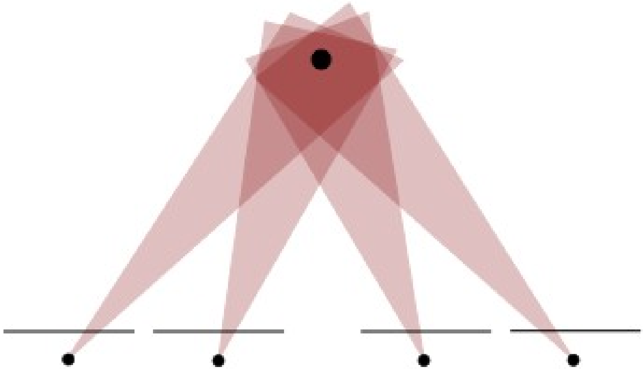

Figure 7– Confidence intervals with our setup. Given each cone coming from each optical center describes the confidence interval for that lens' depth calculation, the dark area where all of the cones intersect is the overall confidence interval.

a given point in space. The smaller the interval, the more certain we can be about the accuracy of the calculated depth. To determine confidence intervals in the simulation, we obtain intervals for each depth calculation, then overlap them to get the overall confidence interval. Each interval is determined by the � term from (2), or $\left[\frac{Bf}{\text{�}+\delta}, \frac{Bf}{\text{�}-\delta}\right]$. Because negative depth would put the point behind the camera, if part of the calculated interval is negative, it is set to zero. We take the intersection of the intervals for the averaging technique, and the solitary interval for the geometric technique. We then compare the average size of these intervals over 1000 trials.

IV. EXPERIMENTAL RESULTS

4.1 Depth Error Results

As pictured in Figure 8, we see that while for a small true depth of one meter, the averaging and geometric techniques return error on the same scale, as the true depth increases to five and ten meters, the geometric results maintain error of around a meter or less, and the averaging results quickly become sporadic and widespread. The geometric results maintain high counts around the zero error area, but the averaging results quickly drop to low counts that in the depth ten scenario ranged from -1884.0 meters to 586.2657 meters. As we predicted, the geometric results provide better depth estimates than the averaging results as the point in space moves away from the camera.

4.2 Confidence Interval Results

Our confidence interval results mirror our depth error results. In Table I, it is apparent that the average size of our confidence intervals increases much more rapidly for depths calculated using the averaging technique than for depths calculated using the geometric technique. The standard deviations of the intervals from the averaging technique also increase rapidly as true depth increases, making the depth calculations increasingly unreliable, especially at a true depth of ten meters. Our results indicate that our geometric results are much more accurate than our averaging results.

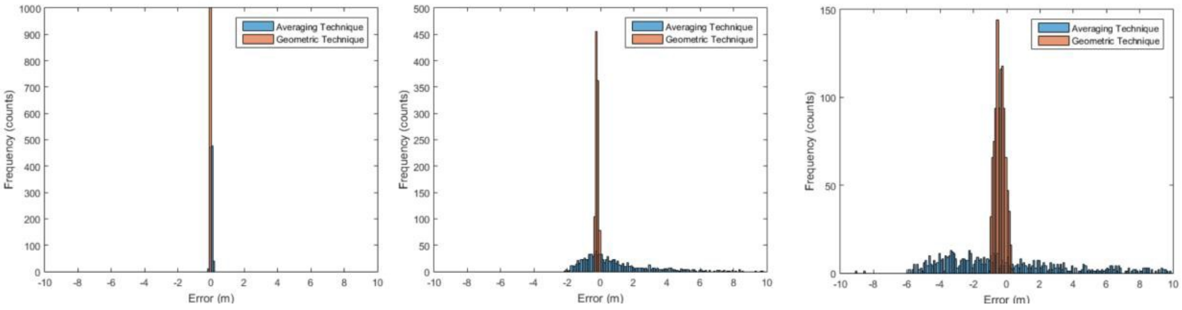

Figure 8– Error of averaging and geometric techniques for varying depths, where error is depth minus the true depth. Plots are set to a [-10,10] range for clearer visual comparison. For Z = 1m, the averaging technique has a range of [-0.1410m,0.1487m] and the geometric technique has a range of [-0.0477m,-0.0345m]. For Z = 5m, the averaging technique has a range of [-2.1288m,18.4689m] and the geometric technique has a range of [-0.3676m,-0.0307m]. For Z = 10m, the averaging technique has a range of [-1884.0m,586.2657m] and the geometric technique has a range of [-1.0136m,0.3156]. The pixel error is � ± 1 pixel, the focal length is 380 pixels, the baseline is 0.03m, and the bias is between zero and one pixels. Notice that the spread of error values greatly increases for the averaging technique as the true depth of the point increases, while the spread of error values remains comparatively small for the geometric technique.

Depth (m)	Mean Averaging Interval Size (m) ± one standard deviation	Mean Geometric Interval Size (m) ± one standard deviation
1	0.0961 ± 0.0617	0.0069 ± 4.0129e-5
2	0.4003 ± 0.2793	0.0277 ± 3.2261e-4
3	1.0031 ± 0.7848	0.0623 ± 0.0011
4	2.0353 ± 2.1785	0.1107 ± 0.0026
5	3.4811 ± 4.8566	0.1731 ± 0.0049
10	8.5002 ± 23.0214	0.6944 ± 0.0412

Table I– Average confidence interval size for varying depths. If the minimum of the interval predicted negative depth, it was replaced with a zero. Notice that the averaging technique intervals not only drastically increase as the true depth increases, but their standard deviations also drastically increase. The geometric technique intervals remain under a meter, with small standard deviations for every true depth value in this table.

V. DISCUSSION AND CONCLUSION

We are able to show through simulation that using knowledge of camera motion through time to mimic large baseline stereo using a small baseline camera provides more accurate depth calculations than averaging small baseline depth calculations over time. Not only does geometric stereo provide depth calculations for large true depth results with consistently small error, but its calculations also have steadily small confidence intervals. The results of this simulation motivate tracking camera motion over time to generate more accurate stereo depth data, with the ultimate goal of building a system that will give a confidence interval for each depth measurement, so it is easy to throw out or minimize the value of unreliable points. By implementing geometric stereo on the DUO-MLX camera, we will be able to mirror results from a camera with a larger baseline without its extra weight and power consumption, therefore meeting our SWAP constraints without sacrificing the accuracy and range of a larger system.

ACKNOWLEDGEMENT

The author would like to acknowledge the support of the National Science Foundation, through NSF REU grant no. 1062672, and the support of Professor Taylor and Professor Van der Spiegel throughout the summer.

REFERENCES

[1] J. Byrne, M. Cosgrove, and R. Mehra, “Stereo Based Obstacle Detection for an Unmanned Air Vehicle,” in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation,* May 2006.

[2] S. Hrabar, “3D Path Planning and Stereo-based Obstacle Avoidance for Rotorcraft UAVs,” in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems,* September 2008.

[3] S. Hrabar, G. Sukhatme, P. Corke, K. Usher, and J. Roberts, "Combined Optic-Flow and Stereo-Based Navigation of Urban Canyons for a UAV," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems,* August 2005.

[4] S. Hrabar, “An Evaluation of Stereo and Laser-Based Range Sensing for Rotorcraft Unmanned Aerial Vehicle Obstacle Avoidance,” *Journal of Field Robotics* 29 (2012): 215–239, doi: 10.1002/rob.21404.

[5] B. Call, R. Beard, C. Taylor, and B. Barber, “Obstacle Avoidance For Unmanned Air Vehicles Using Image Feature Tracking,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit,* August 2006.

[6] J. Engel, T. Schops, and D. Cremers, “LSD-SLAM: Large-Scale Direct Monocular SLAM,” in *European Conference on Computer Vision (ECCV),* 2014.

[7] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, “DTAM: Dense Tracking and Mapping in Real-Time,” in *International Conference on Computer Vision (ICCV),* November 2011.

[8] K. Skifstad and R. Jain, “Range Estimation from Intensity Gradient Analysis,” *Machine Vision and Applications* 2 (1989): 81-102, doi: 10.1007/BF01212370.

[9] R. Szeliski, *Computer Vision: Algorithms and Applications* (Springer, 2010), http://szeliski.org/Book/, 348-354.