Inter-vehicle Distance Estimation Using Displaced Stereo Vision

Alfa Budiman, Mathieu Falardeau
University of Ottawa, Faculty of Engineering

Abstract *– placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here placeholder abstract here*

Keywords – Stereoscopic vision, Robot Operating System, Unmanned Aerial Vehicle, Unmanned Ground Vehicle

I. INTRODUCTION

II. PROBLEM DESCRIPTION

III. SYSTEM DESCRIPTION

IV. METHODOLOGY

The goal is to estimate distances between a target and a UGV. This will be done by combining monocular vision on the UGV's camera with monocular vision on an overhead camera in a similar manner to [3]. In this case, a quadcopter will act as that overhead camera. This method will occur over 4 steps: object detection, direction calculation, position calculation, and distance calculation.

A. Object Detection

The overhead camera is able to detect the UGV and the target object. While many methods exist for object detection using vision, for this application, color was used to facilitate ease of detection of the target [8], [9]. The target was a burgerbot and was colored in solid red. The UGV, a wafflebot was colored in solid black.

The overhead camera sees the burgerbot as a group of red pixels. It also sees the wafflebot as a group of black pixels. The red pixel closest to the group of black pixels is the location of the burgerbot form the quadcopter's perspective. This pixel is annotated with a "+" on figure 1.

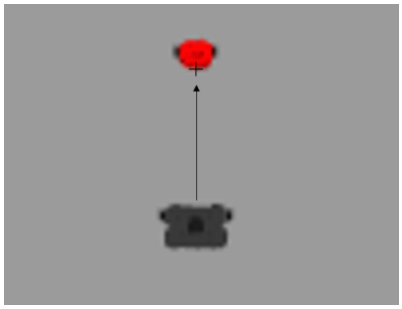

Fig. 1. Quadcopter Camera Image

The wafflebot camera sees the burgergot also as a group of red pixels. The center of these red pixels is the target's location from the perspective of the wafflebot. This is shown as the "+" on figure 2.

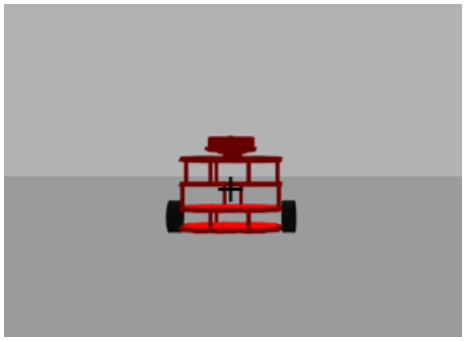

Fig. 2. Wafflebot Camera Image

These pixel coordinates, the locations of the "+" on figures 1 and 2 are extracted and then used to calculate the direction from the respective observer to the target, as unit vectors.

B. Direction Calculation

C. Distance Calculation - Closest Points

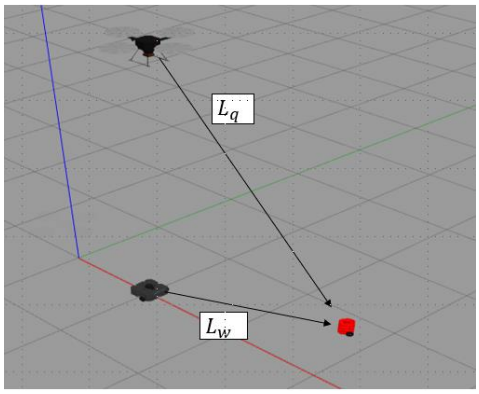

Fig. 3. Wafflebot and Quadcopter Observing Target

Using stereoscopic vision, an object's location can be calculated if given the direction from two cameras and the locations of these cameras [2]. In figure 1 and equations (1), (2), L_q and L_w are the lines in 3D space from the quadcopter to the target and the wafflebot to the target, respectively.

$$L_q = P_q + s\vec{v}_q, \quad L_w = P_w + k\vec{v}_w \qquad (1),(2)$$

P_q and P_w are the location of the quadcopter and wafflebot in reference to the world frame. These locations can be

obtained from the onboard sensors on these observers. $\vec{v}_q$ and $\vec{v}_w$ are unit vectors from the quadcopter to target and wafflebot to target, respectively. These were obtained from direction calculation:

$$P_q = \begin{bmatrix} x_q \\ y_q \\ z_q \end{bmatrix}, \quad P_w = \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} \tag{3),(4}$$

$$\vec{v}_q = \begin{bmatrix} \Delta x_q \\ \Delta y_q \\ \Delta z_q \end{bmatrix}, \quad \vec{v}_w = \begin{bmatrix} \Delta x_w \\ \Delta y_w \\ \Delta z_w \end{bmatrix} \tag{5),(6}$$

Ideally, lines L_q and L_w would intersect, but due to the imperfection of the feature matching proess in object detection and direction calculation, this will almost never occur. Therefore, the two closest points on non-intersecting lines must be found. The middle between these two points is the location of the target.

The parameters s and k that yield the closest points are found when the following conditions are satisfied:

$$(L_q - L_w) \cdot \vec{v}_q = 0, \quad (L_q - L_w) \cdot \vec{v}_w = 0 \tag{7),(8}$$

Expanding these dot products yields a system of linear equations that can be expressed in matrix form $AX = B$:

$$A = \begin{bmatrix} (\Delta x_q{}^2 + \Delta y_q{}^2 + \Delta z_q{}^2) & -(\Delta x_w \Delta x_q + \Delta y_w \Delta y_q + \Delta z_w \Delta z_q) \\ (\Delta x_w \Delta x_q + \Delta y_w \Delta y_q + \Delta z_w \Delta z_q) & (\Delta x_w{}^2 + \Delta y_w{}^2 + \Delta z_w{}^2) \end{bmatrix} \tag{9}$$

$$B = \begin{bmatrix} -(x_q - x_w)\Delta x_q - (y_q - y_w)\Delta y_q - (z_q - z_w)\Delta z_q \\ -(x_q - x_w)\Delta x_w - (y_q - y_w)\Delta y_w - (z_q - z_w)\Delta z_w \end{bmatrix} \tag{10}$$

Solving this system yields the following values of s and k:

$$X = \begin{bmatrix} s \\ k \end{bmatrix} = A^{-1}B = \begin{bmatrix} X_{11} \\ X_{12} \end{bmatrix} \tag{11}$$

Therefore the target's location, P_t can be calulcated by substituting the values of s and k from (11). The distance, d, between the wafflebot and the target is the magnitude of the difference between P_t and P_w.

$$P_t = \frac{1}{2}[L_q(s = X_{11}) + L_w(k = X_{12})] \tag{13}$$

$$d = ||P_t - P_w|| \tag{14}$$

D. Distance Calculation - Sine Law

IV. TESTING AND RESULTS

The distance calculation algorithms were tested by having the target drive through four waypoints as per figure 3. Two tests were conducted; test 1 with the wafflebot as a stationary observer and test 2 with the wafflebot following the burgerbot at a preset distance.

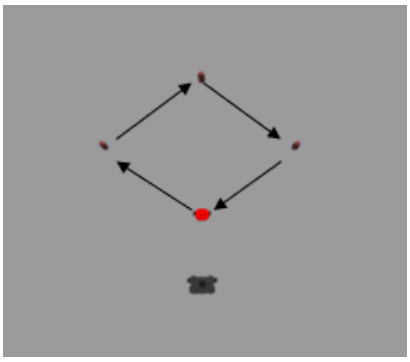

Fig. 4. Test Drive Waypoints

The results of test 1 are on figure 5. The results of test 2 are on figure 6. Both figures 5 and 6 plot the calculated distance from visual data, using methodology described in section IV - C and D, as well as the measured distance from the wafflebot's onboard lidar.

For test drive 2 a simple proportional controller was used to control the wafflebot's position and orientation. This constituted a simplified form of visual servoing [10]. The linear wafflebot's speed, $\dot{x}$ was proportional to the distance between it and the target. Where K_1 is a control gain and d was calculated from (14).

$$\dot{x} = K_1 d \tag{15}$$

The wafflebot's rotational speed $\dot{\theta}$ was proportional to the angle between it and the target. This was calculated as the angle between the vector of the wafflebot's direction $\vec{v}_{WB}$, and the unit vector in from the wafflebot to the target $\vec{v}_w$:

$$\dot{\theta} = K_2 * \frac{\vec{v}_{WB} \cdot \vec{v}_w}{||\vec{v}_{WB}|| * ||\vec{v}_w||} \tag{16}$$

$\vec{v}_w$ is obtained from direction calculation and K_2 is a control gain. $\vec{v}_{WB}$ is known from the sensors on the wafflebot.

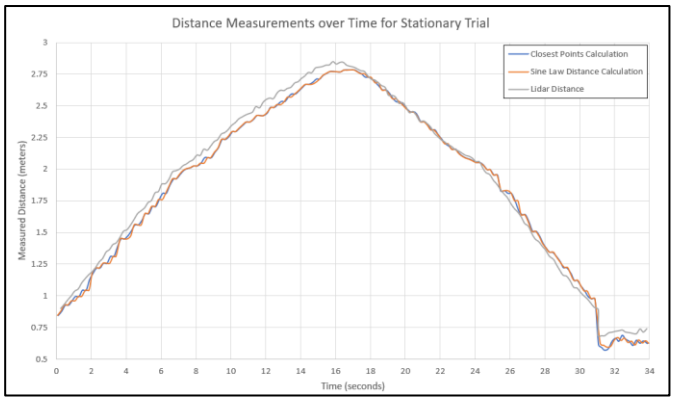

Fig. 5. Distance Measurements for Stationary Trial

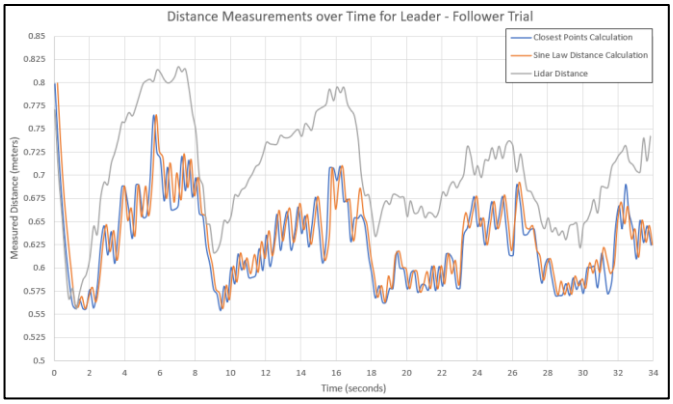

Fig. 6. Distance Measurements for Leader - Follower Trial

V. DISCUSSION AND CONCLUSION

From figures 5 and 6 testing, it can be observed that the calculated distances (obtained from section IV - methodology) closely tracked the lidar distance in the stationary trial. However for the leader follower trial, the calculated distances were not entirely consistent with the measured lidar distance.

During the leader-follower trial

REFERENCES

[1] G. Kim and J. -S. Cho, "Vision-based vehicle detection and inter-vehicle distance estimation," *2012 12th International Conference on Control, Automation and Systems*, 2012, pp. 625-629.

[2] Abdelmoghit Zaarane, Ibtissam Slimani, et all, “Distance measurement system for autonomous vehicles using stereo camera,” *Array Volume 5, 2020, 100016, ISSN 2590-0056*

[3] Z. Ziaei, R. Oftadeh and J. Mattila, "Vision-based path coordination for multiple mobile robots with four steering wheels using an overhead camera," *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2015, pp. 261-268, doi: 10.1109/AIM.2015.7222542.

[4] Y. Shima, "Inter-vehicle distance detection based on keypoint matching for stereo images," *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2017, pp. 1-6, doi: 10.1109/CISP-BMEI.2017.8302064.

[5] M. T. Bui, R. Doskocil and V. Krivanek, "Distance and angle measurement using monocular vision," *2018 18th International Conference on Mechatronics - Mechatronika (ME)*, 2018, pp. 1-6.

[6] P. Avanzini, B. Thuilot and P. Martinet, "Accurate platoon control of urban vehicles, based solely on monocular vision," *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 6077-6082, doi: 10.1109/IROS.2010.5650018.

[7] C. J. R. McCook and J. M. Esposito, "Flocking for Heterogeneous Robot Swarms: A Military Convoy Scenario," *2007 Thirty-Ninth Southeastern Symposium on System Theory*, 2007, pp. 26-31, doi: 10.1109/SSST.2007.352311.

[8] J. Li, J. Wang and J. Mao, "Color moving object detection method based on automatic color clustering," Proceedings of the 33rd Chinese Control Conference, 2014, pp. 7232-7235, doi: 10.1109/ChiCC.2014.6896196.

[9] A. N. Fitriana, K. Mutijarsa and W. Adiprawita, "Color-based segmentation and feature detection for ball and goal post on mobile soccer robot game field," 2016 International Conference on Information Technology Systems and Innovation (ICITSI), 2016, pp. 1-4, doi: 10.1109/ICITSI.2016.7858232.

[10] H. Mekki and M. Letaief, "Path planning for 3D visual servoing: For a wheeled mobile robot," 2013 International Conference on Individual and Collective Behaviors in Robotics (ICBR), 2013, pp. 86-91, doi: 10.1109/ICBR.2013.6729262..