

Camera Image Based Moving Platform Rotation Estimation for Quadrotor Landing

Joonyoung Jang

Mechanical and Aerospace Engineering Department
University of California, San Diego
San Diego, USA
j7jang@eng.ucsd.edu

Miguel Angel Martinez Macias

Mechanical and Aerospace Engineering Department
University of California, San Diego
San Diego, USA
mam013@eng.ucsd.edu

Thomas Bewley

Mechanical and Aerospace Engineering Department
University of California, San Diego
San Diego, USA
bewley@eng.ucsd.edu

Abstract—This paper considers the 3D rotation estimation of a moving platform from 2D images captured by a camera. Assume that a circular pattern marker is on the flight deck of a ship and quadrotor hovers on the center of a platform. The quadrotor has a camera that captures 2D images of markers and measures the distance between markers. The circular pattern of markers changes to an ellipse as the platform rotates. The ellipse equation can be derived from marker positions on an ellipse, and a platform's rotation sequence can be determined by leveraging geometry and trigonometric functions. As the platform's rotation can be estimated, the quadrotor is able to decide the timing that platform rotation is within the quadrotor's capacity to land safely. A simulation and a hardware test were performed to verify the estimation method, and the estimation error was discussed.

Index Terms—Rotation estimation; Quadrotor landing; Ellipse; Trigonometric function

I. INTRODUCTION

The 3D rotation estimation of a moving platform based on a 2D camera image is a valuable method for safely recovering aerial vehicles to the platform. Quadrotors and other Vertical Take Off and Landing Unmanned Aerial Vehicles (VTOL UAVs) have limited operation time because of their battery capacity. Thus, these assets are required to land on moving platforms after missions. There is a study about extending a quadrotor's operation time by a tethering power cable[1]. However, quadrotors must return to the base platform for maintenance or to prepare further operations. Platforms on the sea, such as navy ships, move in 6-DOF motion during operations because of winds and waves. Motivated by the navy helo landing on the navy ship flight deck[2], we suggest the rotation estimation in the quadrotor landing process. In the video, a navy helo pilot tries to land on a navy ship in a rough sea. A pilot waits for the platform motion to be stable enough and safely lands on the flight deck.

Researches about vision-guided autonomous UAV landing and Autonomous Landing of an Aerial Vehicle on Ground Vehicles consider precise landing on a target[3][4]. However, in these studies, the target marker is stationary or moves

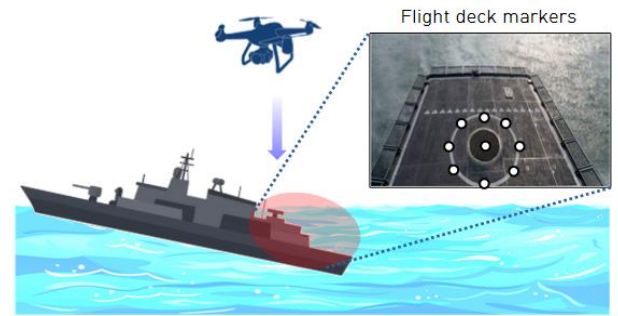


Fig. 1: Conceptual image of a quadrotor landing on the navy ship after its mission. The circular pattern markers on the flight deck appear as an ellipse as the ship rotates.

translationally but did not consider platform rotations. Although the landing problem on a ship was discussed in another research, this study focused on the estimation algorithm of the ship's position. Still, it did not consider the ship's rotational motion[5]. Since the ship's roll is approximately maximum $\pm 30^\circ$ at sea state 6, it could be severe to damage UAVs[6]. Hence, a UAV needs to stand by until the rotational motion is within its capacity for a safe landing.

ArUco markers were broadly used for estimating platform pose and aiding the autonomous landing of UAVs[7][8][9]. Although these markers help estimate rotations, setting up ArUco markers on the navy ship flight deck is problematic. ArUco markers are conspicuous so that enemy reconnaissance aircraft can easily spot the navy ship. Also, the algorithm that distinguishes the ArUco markers could be ineffective if markers are damaged in combat situations. Hence, we decided to study using simple circle markers in a circular pattern on the flight deck for estimating rotations.

We assume a quadrotor has a camera that can obtain 2D images and recognize markers on the flight deck, as seen in Figure 1. The camera can acknowledge circle images and

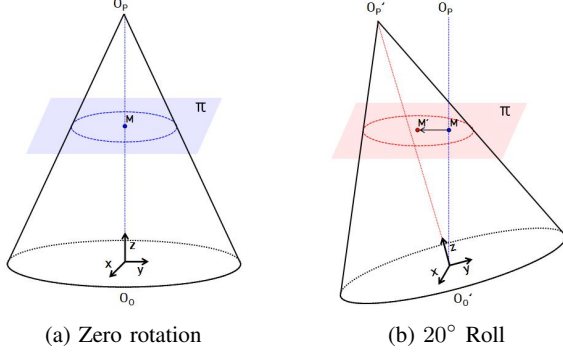


Fig. 2: The conic section camera images as an object rotate. O_O is the circular shape object center, π is regarded as the camera image sensor, M is the center of the image on π , and O_P is the projection center.

measure the pixel position of the circles[10]. Initially, the centers of the camera image sensor and the circular pattern are aligned on the circular corn's revolution axis[11]. As the object is rotated, the contour of the image becomes an ellipse, and the eccentricity is larger. The ellipse equation can be determined by marker positions, and by analyzing the geometry of the ellipse, the rotation estimation can be achieved[12].

The simulation model platform replicating ship motion and circular pattern markers were developed to imitate conic section camera images. As the platform rotated, circular pattern markers formed an ellipse, and the rotation estimation algorithm evaluated the rotation angles of the deck. The hardware test was also conducted to verify the performance of the method.

Overall, this paper mainly discusses the followings: First, the characteristics of camera images and ellipse properties. Second, the rotation estimation process by utilizing the geometry of the ellipse. Lastly, analyzing simulation and hardware test results.

The organization of this paper is as follows. Section II is preliminary that describes the camera image conic section and ellipse equation. The rotation sequence chosen for rotation estimation is also discussed. Section III provides details of the rotation estimation process. By using marker positions, the ellipse equation is determined, and the rotation sequence can be calculated by the ellipse geometry. Section IV presents the simulation and hardware test for validating the performance of the rotation estimation method. Lastly, the limitation of this study and future works are described in Section V.

II. PRELIMINARY

A. Camera Image

Camera images are reflections of three-dimensional(3D) objects on a two-dimensional(2D) plane. In 2D camera images, the object gets smaller as it goes far from the camera, and if the object is continuous, it finally converges at a single point[13]. Thus, camera images are distorted by this effect, also known as the foreshortening effect. Figure 2 shows camera images on

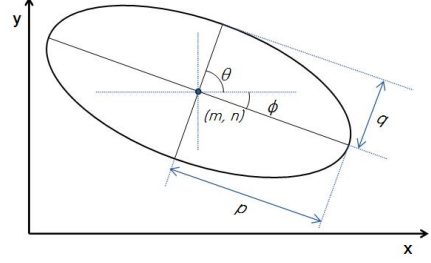


Fig. 3: Ellipse with the center (m, n) , semi-major axis p , and semi-minor axis q . θ , and ϕ are the tilt angles of an ellipse.

the image plane π depending on the rotation of the object plane O_O . The camera image is the conic section of corn formed by the revolution axis, a straight line connecting O_P and O_O . The shape of the projected images on π , blue, and red dashed lines in Figure 2 change from a circle to an ellipse. Also, since the corn's axis tilts, the circle's center M shifts to the circle of the ellipse M' [14][15]. Based on these properties, a camera model was built and used to simulate the 2D camera image from 3D points. The camera model and conic section can be defined by camera characteristics, such as pixel resolution, focal angle, and focal length[16].

B. Ellipse Equation

Section II-A describes shape changes on the camera image plane as a circular object rotates. Since the camera image appears in an ellipse depending on the object rotation, rotation estimation can be achieved by analyzing the ellipse. One can write the ellipse equation that defines Figure 3,

$$\frac{(x-m)^2}{p^2} + \frac{(y-n)^2}{q^2} = 1, \quad (1)$$

where (m, n) is the center of the ellipse position in the body frame. (p, q) are the length of the semi-major and semi-minor axis, relatively. The ellipse equation can be rewritten in the polynomial form[17],

$$ax^2 + bxy + cy^2 + dx + ey + f = 0. \quad (2)$$

As (2) has 6 unknown constants (a, b, c, d, e, f) , minimum of 6 positions (x, y) are required to determine the ellipse. Rearranging (2) by moving the term ax^2 to the right-hand side of the equation, the matrix form of the ellipse equation becomes,

$$\begin{bmatrix} x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ x_2 y_2 & y_2^2 & x_2 & y_2 & 1 \\ x_3 y_3 & y_3^2 & x_3 & y_3 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i y_i & y_i^2 & x_i & y_i & 1 \end{bmatrix} \begin{bmatrix} b' \\ c' \\ d' \\ e' \\ f' \end{bmatrix} = \begin{bmatrix} -x_1^2 \\ -x_2^2 \\ -x_3^2 \\ \vdots \\ -x_i^2 \end{bmatrix}, \quad (3)$$

with

$$b' = \frac{b}{a}, \quad c' = \frac{c}{a}, \quad d' = \frac{d}{a}, \quad e' = \frac{e}{a}, \quad f' = \frac{f}{a},$$

where $a = 1$, and (x_i, y_i) is the position of i -th point on the ellipse. Since a is defined, minimum of 5 points are

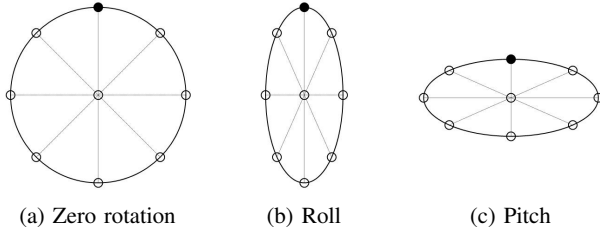


Fig. 4: Top view figures of markers on a platform. Small circles are markers, and the black circle is the bow marker. Markers form a circle with zero rotation but an ellipse when the platform rotates. In the cases of only roll or pitch, the direction of rotation is ambiguous.

required to define the ellipse equation. 5 unknown constants can be determined by solving (3), and the ellipse parameters (m, n, p, q) in (1) can be written as[18][19],

$$m = \frac{cd - bf}{b^2 - ac}, \quad n = \frac{ae - bd}{b^2 - ac},$$

$$p = \sqrt{\frac{2(ae^2 + cd^2 + fb^2 - 2bde - acf)}{(b^2 - ac)(\sqrt{(a - c)^2 + 4b^2} - (a + c))}},$$

$$q = \sqrt{\frac{2(ae^2 + cd^2 + fb^2 - 2bde - acf)}{(b^2 - ac)(-\sqrt{(a - c)^2 + 4b^2} - (a + c))}}.$$

The tilt angle of an ellipse represents the angle between the minor axis and the horizontal axis of the image plane can be formulated as follows[12][20],

$$\theta = \begin{cases} \frac{1}{2} \cot^{-1}\left(\frac{c-a}{2b}\right) + \frac{\pi}{2}, & \text{if } p < q \\ \frac{1}{2} \cot^{-1}\left(\frac{c-a}{2b}\right) - \pi, & \text{if } b < 0, \quad \frac{1}{2} \cot^{-1}\left(\frac{c-a}{2b}\right) > \frac{\pi}{2} \\ \frac{1}{2} \cot^{-1}\left(\frac{c-a}{2b}\right), & \text{otherwise} \end{cases} \quad (4)$$

and the range is $(-\pi/2 \leq \theta \leq \pi/2)$. Also, the angle between the major axis and horizontal axis of the image plane can be written in terms of θ ,

$$\phi = \text{sign}(\theta)\left(\frac{\pi}{2} - \theta\right)$$

C. Rotation Sequence

The 3D rotation of the solid body can be described by a set of three rotation angles (α, β, γ) . Tait-Bryan and Euler rotation sequences are commonly used to explain the rotation of an object.

1) *Sign Ambiguity*: The rotation estimation from a 2D image can cause the rotation sign ambiguity problem. If the distance between the object and the camera is far or the rotation is very small, the camera image distortion is trivial. Thus, when a platform rotates only roll or pitch, the direction of rotation cannot be determined since the images are equivalent, as seen in Figure 4.

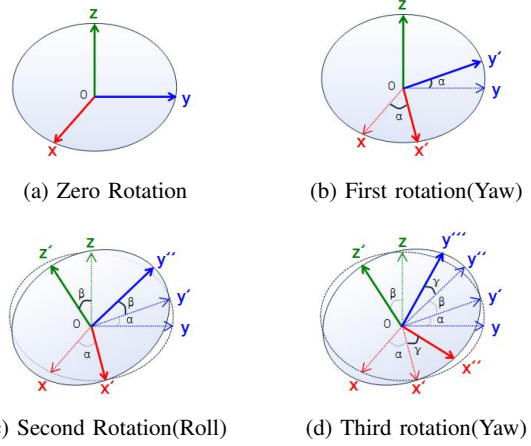


Fig. 5: 3-1-3 Euler rotation sequence

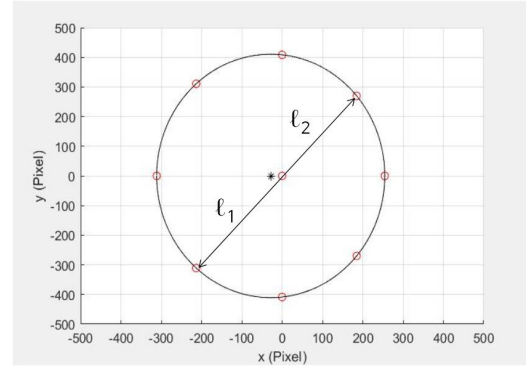


Fig. 6: Simulated distorted camera image determined by 20° roll platform rotation. Red circles are circular pattern markers. The black star is the ellipse's center that is shifted by rotation.

2) *3-1-3 Euler Rotation Sequence*: The 3-2-1 Tait-Bryan rotation sequence comprises yaw, pitch, and roll pitch angles. Potentially, there are two sign ambiguities, which are roll and pitch. However, the 3-1-3 Euler rotation sequence has yaw, roll, and another yaw angle. Thus, utilizing the 3-1-3 Euler angle rotation sequence is beneficial to reduce the possibility of sign ambiguity. As seen in Figure 5, the 3D rotation of an object can be described with 3-1-3 Euler rotation sequence[21]: Rotate the object by an angle α about z -axis of the body frame, then β about x -axis, finally γ about z -axis.

III. ROTATION ESTIMATION

A. Determine Camera Image Conic Section

Assume that there are 9 markers on a ship flight deck. 8 markers are positioned in a circular pattern, and 1 marker is in the circle's center. Figure 2 illustrates conic sections shown on the camera image plane as a ship rotates. As explained in Section II-A, a rotated circle image appears as an ellipse, and the ellipse's center is shifted. Each marker's pixel coordinates position (X_c, Y_c, Z_c) , representing position

relative to the camera coordinate with the origin at the center, can be computed from the pinhole camera model equation[16],

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = K * T * \begin{bmatrix} P_x \\ P_y \\ P_z + h \end{bmatrix}, \quad (5)$$

where (P_x, P_y, P_z) is 3D marker position in platform coordinate and h is the altitude of quadrotor. Matrices K , T are defined as,

$$K = \begin{bmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{bmatrix}, \quad T = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix},$$

where fx and fy are focal lengths, cx and cy indicate the camera center coordinate. The camera characteristics determine K , and T represents the transformation from the object coordinate to the camera image coordinate. The image coordinate system's origin is located at the top left corner so that 2D pixel marker positions (X_p, Y_p) can be written as,

$$X_p = \frac{X_c}{Z_c} - cx, \quad Y_p = cy - \frac{Y_c}{Z_c}. \quad (6)$$

Finally, by (3), the polynomial form ellipse equation can be determined since 2D pixel marker positions are given by (6).

B. Obtain Corrected Image

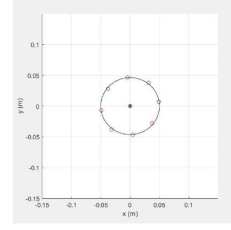
As described in II-A, camera images are distorted because of the foreshortening effect, and this effect can solve the sign ambiguity problem cause the ellipse center is shifted. However, distorted camera images must be corrected since the rotation sequence illustrated in Figure 5 is based on geometrically undistorted images. The 2D marker position vectors $\vec{l}_i = (X_{p_i}, Y_{p_i})$ can be defined as,

$$\vec{l}_i = d_i \vec{c}_i$$

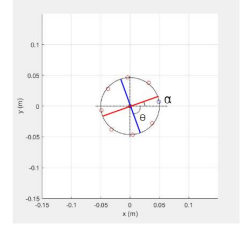
where $\vec{c}_i$ is the unit marker position vector, d_i is the distance between the center marker and the i -th marker. In the distorted image shown in Figure 6, $\vec{l}_1$ and $\vec{l}_2$ are marker position vectors in the opposite direction. Assuming that a quadrotor has an altimeter and maintains hovering at the center of the platform, the distorted camera image can be converted to the undistorted marker positions in the platform coordinate utilizing (5), (6). By plugging $\vec{l}_i$ into the pinhole camera model equation and rearranging equations, then the formulations are derived as below[12],

$$\begin{aligned} P_{x_i} &= \frac{-d_1 \vec{c}_{x_0} - d_2 \vec{c}_{y_0} - 2d_1 d_2 \cos(\Psi)}{fy(d_1 - d_2)} h \\ P_{y_i} &= \frac{d_1 \vec{c}_{x_0} + d_2 \vec{c}_{y_0} + 2d_1 d_2 \sin(\Psi)}{fx(d_1 - d_2)} h \\ P_{z_i} &= \frac{-d_1 - d_2}{d_1 - d_2} h, \end{aligned} \quad (7)$$

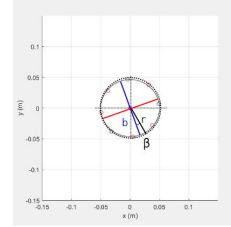
where $(\vec{c}_{x_0}, \vec{c}_{y_0})$ represents the unit center marker position vector, and Ψ is the angle of the marker position with respect to the center marker.



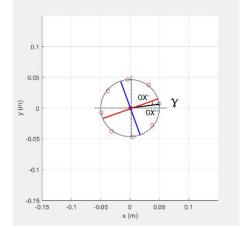
(a) Ellipse after rotation



(b) First rotation(Yaw)



(c) Second Rotation(Roll)



(d) Third rotation(Yaw)

Fig. 7: 3-1-3 Euler rotation sequence angles from undistorted camera images for 20° roll, 10° pitch, and 5° yaw in the platform frame. The red line is the major axis, and the blue line is the minor axis.

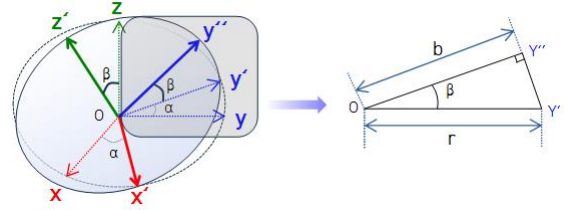


Fig. 8: The side section view of the right triangle formed by the second rotation. b is the semi-minor axis length and r is the marker circle radius.

C. Rotation Estimation

Figure 7 shows the simulated undistorted images illustrated in III-B and the process of determining the 3-1-3 Euler rotation sequence from the images. The detailed process is as follows[12].

1) *The first rotation(Yaw)*: The first rotation angle is the semi-major axis tilt angle for the horizontal axis. Figure 7b is graphically describing that α is the tilt angle of the ellipse and can be written as,

$$\alpha = \phi = \text{sign}(\theta) \left(\frac{\pi}{2} - |\theta| \right).$$

2) *The second rotation(Roll)*: As seen in Figure 7c and 8, β is the angle between the semi-minor axis and the radius vector of the marker circle. The side section of the semi-minor axis, and the radius vector is the right triangle with rotation angle β . Since markers form a circle before the second rotation, the longer side of the triangle is circle radius r . The sign of β is

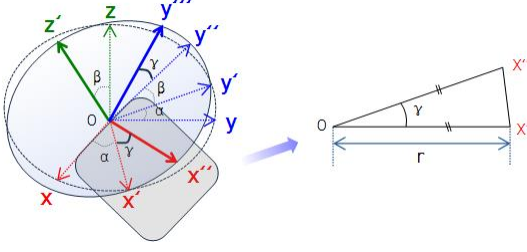


Fig. 9: The angled side section view of the isosceles triangle formed by the third rotation.

determined by the z component of the semi-minor axis OY'' . Finally, the equation of β can be formulated as,

$$\beta = \text{sign}(Y_z'') \cos^{-1}\left(\frac{b}{r}\right),$$

where b represents the semi-minor axis length.

3) *The third rotation(Yaw):* The third rotation can be estimated by calculating the rotation angle between major axes OX' , and the bow marker vector OX'' . In Figure 7d, the length of the major axis is the same as circular pattern radius r , equivalent to the length of OX' and OX'' . Figure 9 shows the triangle formed by the third rotation angle. Since the section is an isosceles triangle, γ can be calculated leveraging the law of cosines, and the sign of the angle is determined by the y component of the bow light vector OX'' , which can be determined by (7). Then, the formulation of γ becomes,

$$\gamma = \text{sign}(X_y'')(\cos^{-1}\left(\frac{2r^2 - |OX' - OX''|^2}{2r^2}\right)).$$

4) *Rotation conversion:* Since the estimated rotation angles are 3-1-3 Euler rotation sequences, rotations need to be converted to determine roll, pitch, and yaw angles. First, 3-1-3 Euler rotation angles (α, β, γ) calculated by the estimation process can be converted to quaternion rotations $q = [q_0, q_1, q_2, q_3]$ by,

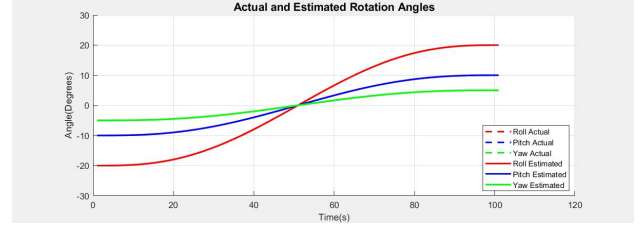
$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} c_{\gamma/2}c_{\beta/2}c_{\alpha/2} - s_{\gamma/2}c_{\beta/2}s_{\alpha/2} \\ c_{\gamma/2}s_{\beta/2}c_{\alpha/2} + s_{\gamma/2}s_{\beta/2}s_{\alpha/2} \\ s_{\gamma/2}s_{\beta/2}c_{\alpha/2} - c_{\gamma/2}s_{\beta/2}s_{\alpha/2} \\ c_{\gamma/2}c_{\beta/2}s_{\alpha/2} + s_{\gamma/2}c_{\beta/2}c_{\alpha/2} \end{bmatrix}$$

with $c_{\alpha/2} = \cos(\alpha/2)$, and $s_{\alpha/2} = \sin(\alpha/2)$. Once the quaternion rotation is obtained, the roll, pitch, and yaw angles can be computed by the quaternion-Euler angle conversion formulation as follows[22],

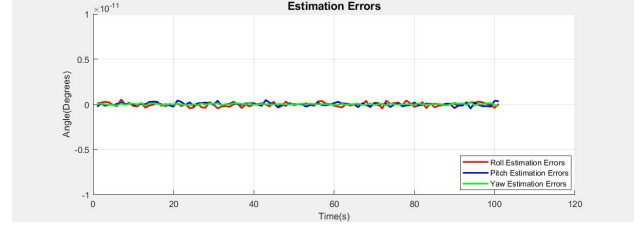
$$\begin{aligned} \text{Roll} &= \text{atan2}(2(q_0q_1 + q_2q_3), 1 - 2(q_1^2 + q_2^2)), \\ \text{Pitch} &= \text{asin}(2(q_1q_3 - q_0q_2)), \\ \text{Yaw} &= \text{atan2}(2(q_0q_3 + q_2q_1), 1 - 2(q_2^2 + q_3^2)). \end{aligned}$$

IV. EXPERIMENTAL RESULTS

To verify the theory, the simulation model was developed and tested. Also, the scaled-down hardware experiment was performed to test the performance of the rotation estimation, and the Open source Computer Vision(OpenCV)

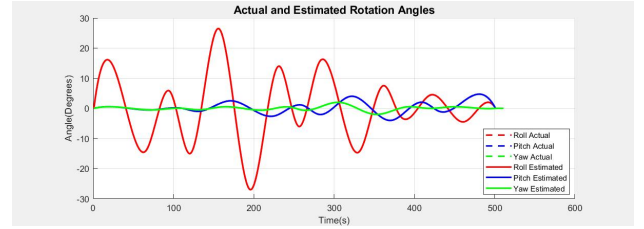


(a) Actual and Estimated Rotation Angles

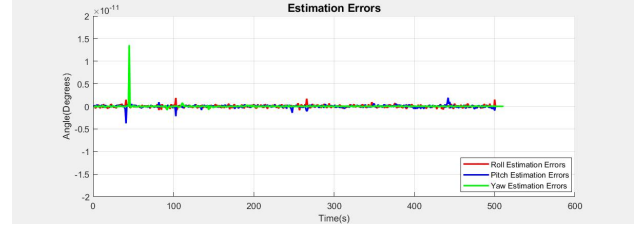


(b) Estimation Error

Fig. 10: Case 1. Angles change in one direction. Actual angles are dashed lines, and estimated angles are solid lines.



(a) Actual and Estimated Rotation Angles



(b) Estimation Error

Fig. 11: Case 2. The ship motion model

algorithm[23] was used to recognize markers and measure the marker positions.

A. Simulation Model Test

1) *Platform Model:* The simulation model platform is the navy ship flight deck with 9 markers on it: 8 markers form a circle, while 1 marker is located in the center of the platform. We assumed the quadrotor hovers at 5 meters altitude, tracking the platform center marker and maintaining positions while the ship moves. In the simulation, the marker circle diameter is 1 meter, and the flight deck is 10x10 meters square. 2 cases of motions were tested: Case 1. Rotation in one direction(Roll: $-20^\circ \sim 20^\circ$, Pitch: $-10^\circ \sim 10^\circ$, Yaw: $-5^\circ \sim 5^\circ$), and Case 2.

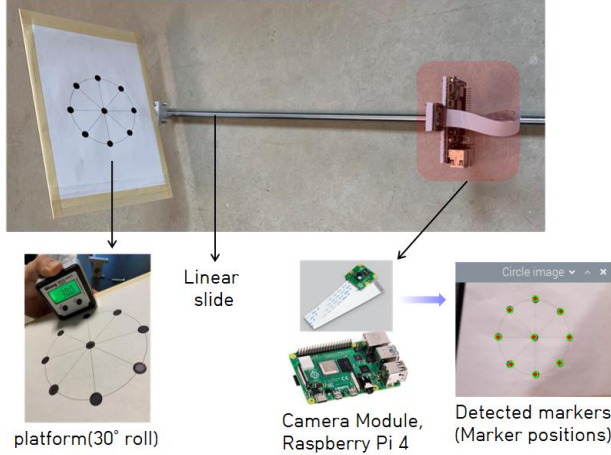


Fig. 12: Hardware model image. The camera module, Raspberry Pi, and a plate with circular pattern markers are connected to a linear slide. The camera and the center mark are aligned. The camera detects markers, and the marker positions are obtained by the OpenCV algorithm.

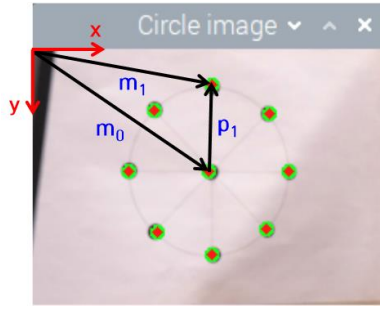


Fig. 13: Detected circle image. m_0 is the center marker, m_1 is the bow marker position in the camera image coordinate.

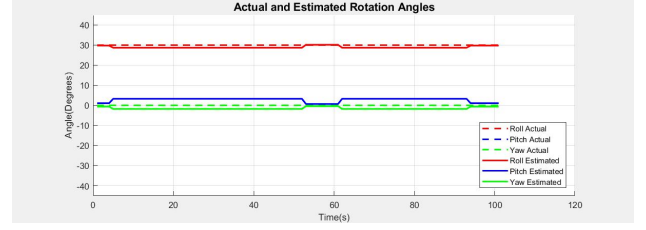
Mathematical Ship Motion Model(USS Oliver Hazard Perry class destroyer in Sea State 6)[24].

2) *Quadrotor Camera Model*: We assume the quadrotor has “Raspberry Pi Camera Module V2-8,” which is equivalent to the hardware model. The pixel resolution is 3280 x 2464, and the focal angle is horizontal 45.4° and vertical 31.1°.

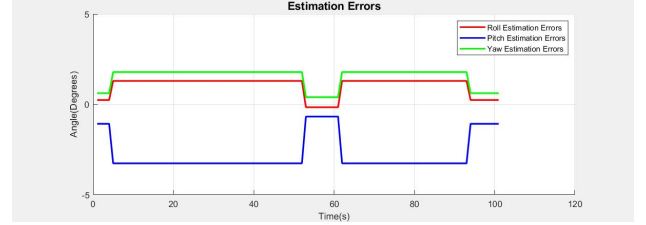
TABLE I: Estimation RMSE (Degrees)

Motion	RMSE: Case 1	RMSE: Case 2
Roll	0.2235×10^{-12}	0.3010×10^{-12}
Pitch	0.1814×10^{-12}	0.3126×10^{-12}
Yaw	0.0771×10^{-12}	0.6083×10^{-12}

3) *Result*: The rotation estimation of each case is shown in Figures 10 and 11. Table I shows each case and rotation’s Root Mean Square Errors(RMSEs). The RMSE of Case 1 was from $0.0771 \times 10^{-12} \sim 0.2235 \times 10^{-12}$, and Case 2 was $0.3010 \times 10^{-12} \sim 0.6083 \times 10^{-12}$. The results indicate that the estimation algorithm performs with minimal errors.



(a) Actual and Estimated Rotation Angles



(b) Estimation Error

Fig. 14: Hardware model experiment result

B. Hardware Model Test

Figure 12 shows the hardware model for testing the rotation estimation algorithm. Hardware model size is determined by scaling down the simulation model with a 1/10 ratio. Thus, the distance between the camera center and the platform center is 50 centimeters, and the diameter of circular pattern markers is 10 centimeters. The platform mount is the fixed mount rolled 30 degrees. In Figure 13, i -th marker 2D position from the center marker, $p_i = (X_{p_i}, Y_{p_i})$, can be calculated by,

$$p_i = m_i - m_0.$$

The position of markers is measured 10 seconds with a 10Hz frequency. By utilizing marker positions p_i , the ellipse equation is determined by (3). Figure 14 shows the hardware test result. The estimation errors tend to be consistent, and RMSEs were roll 1.1577°, pitch 2.9360°, and yaw 1.6080°.

V. CONCLUSION AND FUTURE WORKS

This study presents the moving platform rotation estimation for quadrotor landing. Assuming that the circular image with 9 markers is on the moving platform, and a quadrotor tracks the platform center, a quadrotor camera obtains a circular image on its image plane. Camera images are distorted because of the foreshortening effect. Thus, if the platform is rotated, circular marker images become ellipses on the image plane. By analyzing the ellipse determined by marker positions, the rotation estimation of a platform can be achieved. Simulation and hardware models were tested, and the result showed that the rotation estimation algorithm was performed with less than 0.7×10^{-12} RMSE. Future works focus on improving the algorithm and hardware model. Since estimation errors were consistent, a hardware model could be calibrated by performing further tests to minimize errors. Also, a quadrotor is assumed to be hovering on the platform center in this study. In reality, there is a tracking error causing the quadrotor

is not aligned to the platform center. Thus, the estimation algorithm needs to be enhanced to estimate 6-DOF platform motions including surge, sway, and heave. For improving the hardware model, a cable-driven motion simulator will be developed in the Coordinated Robotics Lab at the University of California, San Diego. This simulator moves in 6-DOF motions replicating a boat motion. Circular pattern markers will be set up on the simulator's platform, and a camera will be located on the simulator to perform the rotation estimation.

REFERENCES

- [1] K. A. Talke, "Hanging tether management for unmanned air - surface vehicle teams," (Order No. 28718622), UC San Diego; ProQuest Dissertations & Theses A&I, 2021, (2584293491). Retrieved from <https://www.proquest.com/dissertations-theses/hanging-tether-management-unmanned-air-surface/docview/2584293491/se-2>
- [2] R. Barbieri, "Lynx helicopter landing on ship in rough sea [Video]," YouTube, 2013 January, <https://www.youtube.com/watch?v=NJIZTL2ZyEw>
- [3] R. Liu et al., "Vision-guided autonomous landing of multirotor UAV on fixed landing marker," 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 2020, pp. 455-458, doi: 10.1109/ICAICA50127.2020.9182512.
- [4] J. Kim, S. Woo and J. Kim, "Lidar-guided autonomous landing of an aerial vehicle on a ground vehicle," 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Jeju, Korea (South), 2017, pp. 228-231, doi: 10.1109/URAI.2017.7992719.
- [5] P. Wu, Z. Shi and P. Yan, "Improved EKF-SLAM Algorithm of Unmanned Helicopter Autonomous Landing on Ship," 2018 37th Chinese Control Conference (CCC), Wuhan, China, 2018, pp. 5287-5292, doi: 10.23919/ChiCC.2018.8483338.
- [6] J. L. Sanchez-Lopez, J. Pestana, S. Saripalli, and P. Campoy, "An Approach Toward Visual Autonomous Ship Board Landing of a VTOL UAV," 2014, J. Intell. Robotics Syst., 74, 1-2 (April 2014), pp. 113-127, <https://doi.org/10.1007/s10846-013-9926-3>.
- [7] M. F. Sani and G. Karimian, "Automatic navigation and landing of an indoor AR. drone quadrotor using ArUco marker and inertial sensors," 2017 International Conference on Computer and Drone Applications (IConDA), Kuching, Malaysia, 2017, pp. 102-107, doi: 10.1109/ICONDA.2017.8270408.
- [8] A. S. Nair, P. A. Jeyanthi, L. Ramesh, G. M. Kurian and S. R. Mohamed, "Autonomous Precision Landing with UAV and Auto charging," 2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, 2021, pp. 463-466, doi: 10.1109/RTEICT52294.2021.9573947.
- [9] A. Marut, K. Wojtowicz and K. Falkowski, "ArUco markers pose estimation in UAV landing aid system," 2019 IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace), Turin, Italy, 2019, pp. 261-266, doi: 10.1109/MetroAeroSpace.2019.8869572.
- [10] W. Chinsatitf and T. Saitoh, "Improvement of eye detection performance for inside-out camera," 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, 2016, pp. 1-6, doi: 10.1109/ICIS.2016.7550794.
- [11] X. Ying and H. Zha, "Geometric Interpretations of the Relation between the Image of the Absolute Conic and Sphere Images," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2031-2036, Dec. 2006, doi: 10.1109/TPAMI.2006.245.
- [12] M. A. Martinez, "USV Attitude Position Estimation by a Hovering UAV Using Monocular Images of Deck-Mounted Lights," UC San Diego, 2022, ProQuest ID: Martinez_ucsd_0033M_21729. Merritt ID: ark:/13030/m5tr39jf, Retrieved from <https://escholarship.org/uc/item/5p02s4st>
- [13] J. Kennedy, I. Juricevic, "Foreshortening gives way to forelengthening. Perception, 31, pp. 893-894, 2002.
- [14] Y. Liu, Y. Dong, Z. Song and M. Wang, "2D-3D Point Set Registration Based on Global Rotation Search," in IEEE Transactions on Image Processing, vol. 28, no. 5, pp. 2599-2613, May 2019, doi: 10.1109/TIP.2018.2887207.
- [15] S. Su et al., "A Novel Camera Calibration Method Based on Multilevel-Edge-Fitting Ellipse-Shaped Analytical Model," in IEEE Sensors Journal, vol. 20, no. 11, pp. 5818-5826, 1 June 1, 2020, doi: 10.1109/JSEN.2020.2972615.
- [16] P. Sturm, "Pinhole Camera Model," In: Ikeuchi, K. (eds) Computer Vision. Springer, Boston, MA., 2014, https://doi.org/10.1007/978-0-387-31439-6_472
- [17] E. Weisstein, "Ellipse," From MathWorld—A Wolfram Web Resource. <https://mathworld.wolfram.com/Ellipse.html>
- [18] B. Peng, S. Dang and M. K. Dutta, "Feature detection and parameters calculation of arbitrary ellipse based on image processing technology," 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2015, pp. 418-420, doi: 10.1109/SPIN.2015.7095169.
- [19] Y. Soh, J. Bae, D. Kim and H. Kim, "A New Method for Ellipse Fitting in Satellite Images," 2009 Second International Conference on Intelligent Computation Technology and Automation, Changsha, China, 2009, pp. 502-506, doi: 10.1109/ICICTA.2009.128.
- [20] X. Morin, E. Pottier, J. Saillard, C. Pasdeloup and C. Delhote, "Polarimetric Detection of Slowly Moving Targets Embedded in Ground Clutter," Progress in Electromagnetics Research-pier - PROG ELECTROMAGN RES, 16, pp. 1-33, 1997, doi: 10.2528/PIER95103100.
- [21] T. Bewley, "Numerical renaissance," Renaissance Press, pp. 488-490, 2017.
- [22] J. Diebel, Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors. Matrix. 58, 2006
- [23] G. Bradski and A. Kaehler, "Learning OpenCV: Computer vision with the OpenCV library," O'Reilly Media, Inc, 2008.
- [24] J. L. Sanchez-Lopez, J. Pestana, S. Saripalli, and P. Campoy, "An Approach Toward Visual Autonomous Ship Board Landing of a VTOL UAV," J. Intell. Robotics Syst. 74, 1-2, April 2014, pp. 113-127, doi: <https://doi.org/10.1007/s10846-013-9926-3>.