Visual Navigation System for Autonomous Drone using Fiducial Marker Detection

Mohammad Soleimani Amiri
Department of Mechanical and Manufacturing Engineering,
Faculty of Engineering and Built Environment,
Universiti Kebangsaan Malaysia,
Bangi 43600, Selangor, Malaysia

Rizauddin Ramli
Department of Mechanical and Manufacturing Engineering,
Faculty of Engineering and Built Environment,
Universiti Kebangsaan Malaysia,
Bangi 43600, Selangor, Malaysia

Abstract—**Drones have been quickly developing for civilian applications in recent years. Because of the nonlinearity of the mathematical drone model, and the importance of precise navigation to avoid possible dangers, it is necessary to establish an algorithm to localize the drone simultaneously and maneuver it to the desired destination. This paper presents a visual-based multi-stage error tolerance navigation algorithm of an autonomous drone by a tag-based fiducial marker detection in finding its target. Dynamic and kinematic models of the drone were developed by Newton-Euler. The position and orientation of the drone, related to the tag, are determined by AprilTag, which is used as feedback in a closed-loop control system with an Adjustable Proportional-Integral-Derivative (APID) controller. Parameters of the controller are tuned based on steady-State error, which is defined as the distance of the drone from the desired point. The sequence of path trajectory, that drone follows to reach the desired point, is defined as a navigation algorithm. A model of the drone was simulated in a virtual outdoor to mimic hovering in complex obstacles environment. The results present satisfactory performance of the navigation system programmed by the APID controller in comparison with the conventional Proportional-Integral-Derivative (PID) controller. It can be ascertained that the proposed navigation system based on a tag marker in the closed-loop control system is applicable to maneuvering the drone autonomously and useful for various industrial tasks in indoor/outdoor environments.**

Keywords—Proportional-Integral-Derivative (PID) controller; AprilTag detection system; autonomous navigation; fiducial marker detection

I. Introduction

The drone has the advantage of reaching high locations and extreme environments that are difficult to be accessed by humans and other ground vehicles [1], [2], [3]. Since its deployment to industrial applications, drones have attracted a lot of attention in various industries [4], [5], [6]. For instance, Huang et al. [7] presented for scene detection by high-resolution imagery adopted through autonomous drone navigation using landmark detection and recognition

The drone can be navigated autonomously or by a human pilot. There are several works about developing the navigation algorithm for the drone[8], [9], [10], [11]. Hodge et al. [12] presented a generic navigation algorithm that utilizes onboard sensors' data of the drone to navigate the drone to the target. Miranda et al. [13] developed a navigation system for autonomous drones that generates a path between a start and a final point and controls the drone to follow this path. Tang et al. [14] studied an algorithm based on a multi-sensor system including multiple cameras and a 2-D laser scanner using the AprilTag target for drone application. Malyuta et al. [15] presented an autonomous drone for precision agriculture applications by employing the drone, flying through the farmland, for long-term monitoring missions without any human supervision by using AprilTag for localization. Lee et al. [16] established drone navigation by mapping the definitions of vehicular autonomy levels to specific drone tasks in order to create a clear definition of autonomy when applied to drones.

From literature [12], [13], [14], [15], [16], there are various types of strategies to increase the performance of navigation of the drone. Developing a simpler navigation strategy with more efficient performance has attracted much attention from researchers. The contributions of this paper are given as follows:

- The kinematics and dynamics of the drone are determined.
- The visual-based navigation and Adjustable Proportional-Integral-Derivative (APID), in which the actual position of the drone is captured by Fiducial Marker Detection, are investigated.

This paper is organized as follows: dynamic models of a drone are expressed in Section II. Section III addresses the development of APID and visual-based navigation. Section IV represents the performance and result of the proposed APID and navigation strategy in real-time navigation of a drone model in the virtual environment interacting with Robot Operating System (ROS). The conclusion is mentioned in Section V.

II. Dynamics of Drone

The mathematical model of the drone is presented as follows [17]:

$$\dot{\eta} = \eta\nu \tag{1}$$

where the vector η represents position and orientation in the earth or inertial frame shown in Fig. 1 as follows:

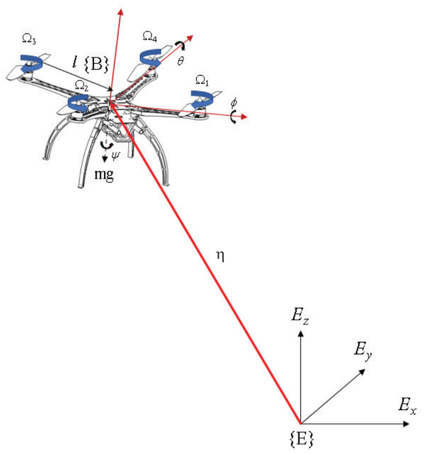


Fig. 1. Drone Model Coordinate Systems.

$$\eta = \left| x \quad y \quad z \quad \phi \quad \theta \quad \psi \right|^T \tag{2}$$

And the equation of motion for the drone can be represents as,

$$M_{RB}\dot{\nu} + C_{RB}(\nu)\nu = \tau \tag{3}$$

where M_{RB} is the rigid-body inertia matrix, $C_{RB}(\nu)$ represents the centripetal and Coriolis terms and τ is the generalized forces and torques, all of which are expressed in the body-fixed frame.

And vector ν represents the linear velocities and angular velocities of the drone in the body-fixed frame as follows:

$$\nu = \begin{vmatrix} V^B \\ \omega^B \end{vmatrix} = \left| u \quad v \quad w \quad p \quad q \quad r \right|^T \tag{4}$$

where V^B is the linear velocities in the body (B) frame and ω^B is the angular velocities in the body frame respectively. In 3 dimension rigid body dynamics, the two reference frames are linked by the linear and angular velocities as,

$$\nu = R(\eta)V^B \tag{5}$$

$$\omega = T(\eta)\omega^B \tag{6}$$

In general, $\jmath(\eta$ is the transformation matrix of the orientation and position of the body frame with respect to the inertial frame of the drone can be shown as [18],

$$\jmath(\eta) = \begin{vmatrix} R(\eta) & 0_{3X3} \\ 0_{3X3} & T(\eta) \end{vmatrix} \tag{7}$$

The transformation matrix that indicates the relation between angular velocities in the body frame and angular velocities in the inertial frame is represented as follows,

$$T(\eta) = \begin{vmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{vmatrix} \tag{8}$$

The transformation matrix $\jmath(\eta)$ with the rotation matrix $R(\eta)$ which describes the relationship between the linear velocities in the body-fixed frame and the linear velocities in the inertial frame is shown as,

$$R(\eta) = R_B^E(\psi)R_B^E(\theta)R_B^E(\phi) \tag{9}$$

$$R(\eta) = \begin{vmatrix} c_\psi c_\theta & -s_\psi c_\phi + c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi c_\phi s_\theta \\ s_\psi c_\theta & c_\psi c_\phi + s_\psi s_\theta s_\phi & -c_\psi s_\phi + s_\psi c_\phi s_\theta \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{vmatrix} \tag{10}$$

Since the vector of linear and angular velocities $\dot{\eta}$ in earth frame can be expressed as,

$$\dot{\eta} = \begin{vmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{vmatrix} \tag{11}$$

The kinematics relation between the linear and angular velocities of the body-fixed frame related to the inertial frame is described by

$$\dot{\eta} = \begin{vmatrix} u(s_\psi c_\theta) + v(s_\psi s_\theta s_\phi + c_\psi c_\theta) + w(c_\psi s_\theta c_\phi + s_\psi s_\phi) \\ u(c_\psi c_\theta) + v(c_\psi s_\theta s_\phi - s_\psi c_\theta) + w(s_\psi s_\theta c_\phi - c_\psi s_\phi) \\ u(-s_\theta) + v(c_\theta s_\phi) + w(c_\theta c_\phi) \\ p + q(s_\psi t_\theta) + r(c_\psi t_\theta) \\ q(c_\psi) - r(s_\psi) \\ q\left(\frac{s_\psi}{c_\theta}\right) + r\left(\frac{c_\psi}{c_\theta}\right) \end{vmatrix} \tag{12}$$

By assuming that the center of mass is fixed at the center of the origin of the body frame and the body has rotational symmetry around the center of mass, the resulting rigid-body inertia matrix M_{RB} is presented as,

$$\begin{aligned} M_{RB} &= \begin{vmatrix} mI_{3X3} & 0_{3X3} \\ 0_{3X3} & I \end{vmatrix} \\ &= \begin{vmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{vmatrix} \end{aligned} \tag{13}$$

where m is the total mass of the drone and I_{xx}, I_{yy} and I_{zz} are the moments of inertia [19]. By using Newton-Euler equation, the drone rigid body are affected by external forces F^B and torques τ^B as,

$$\begin{vmatrix} mI_{3X3} & 0_{3X3} \\ 0_{3X3} & I \end{vmatrix} \begin{vmatrix} \dot{V}^B \\ \dot{\omega}^B \end{vmatrix} + \begin{vmatrix} \omega^B \times mV^B \\ \omega^B \times I\omega^B \end{vmatrix} = \begin{vmatrix} F^B \\ \tau^B \end{vmatrix} \tag{14}$$

Expanding the equation above, we obtain,

$$\begin{vmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{vmatrix} \begin{vmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{vmatrix} + \begin{vmatrix} -rmv + qmw \\ rmu - pmw \\ -qmu + pmv \\ -rI_{yy}q + qI_{zz}r \\ -rI_{xx}p + pI_{zz}r \\ qI_{xx}p - pI_{yy}q \end{vmatrix} = \begin{vmatrix} F_x^B \\ F_y^B \\ F_z^B \\ \tau_x^B \\ \tau_y^B \\ \tau_z^B \end{vmatrix} \tag{15}$$

The external forces and torques for each component of x, y and z axis can be determined as,

$$F_x^B = m(\dot{u} - rv + qw) \tag{16}$$

$$F_y^B = m(\dot{v} + ru - pw) \tag{17}$$

$$F_z^B = m(\dot{w} - qu + pv) \tag{18}$$

$$\tau_x^B = I_{xx}\dot{p} - rq(I_{yy} - I_{zz}) \tag{19}$$

$$\tau_y^B = I_{yy}\dot{q} + rp(I_{xx} - I_{zz}) \tag{20}$$

$$\tau_z^B = I_{zz}\dot{r} - qp(I_{xx} - I_{yy}) \tag{21}$$

The centripetal and Coriolis terms are represented by the matrix

$$C_{RB}(\nu)\nu = \begin{vmatrix} 0 & 0 & 0 & 0 & mw & -mv \\ 0 & 0 & 0 & -mw & 0 & mu \\ 0 & 0 & 0 & mv & -mu & 0 \\ 0 & 0 & 0 & 0 & I_{zz}r & -I_{yy}q \\ 0 & 0 & 0 & -I_{zz} & 0 & I_{xx}p \\ 0 & 0 & 0 & I_{yy}q & -I_{xx}p & 0 \end{vmatrix} \begin{vmatrix} u \\ v \\ w \\ p \\ q \\ r \end{vmatrix} \tag{22}$$

The generalized forces τ can be divided into three components,

$$\tau = \tau_{gravitational} + \tau_{damping} + \tau_{actuators} = G(\eta) + D(\nu) + \tau_c(u) \tag{23}$$

where $G(\eta)$ is the gravitational component, $D(\nu)$is the damping component, $\tau_c(u)$ is the forces generated by the actuators and u is the input vector to the drone's motors.

The gravitational component points downwards along the z-axis with respect to the earth frame which in the body frame corresponds to:

$$G(\eta) = \begin{vmatrix} R_E^B & 0 \\ 0 & 1 \end{vmatrix} \begin{vmatrix} 0 \\ 0 \\ -mg \\ 0 \\ 0 \\ 0 \end{vmatrix} = \begin{vmatrix} -mgs_\theta \\ mgc_\theta sin_\phi \\ mgc_\theta c_\phi \\ 0 \\ 0 \\ 0 \end{vmatrix} \tag{24}$$

where R_E^B denotes the inverse of $R(\eta)$ as rotation matrix of the body-fixed frame relative to inertial frame.

$$R_E^B = \begin{vmatrix} c_\psi c_\theta & s_\psi c_\theta & -s_\theta \\ c_\psi s_\theta s_\phi - s_\psi c_\phi & s_\psi s_\theta s_\phi + c_\psi c_\theta & c_\theta s_\phi \\ c_\psi s_\theta c_\phi + s_\psi s_\phi & s_\psi s_\theta c_\phi - c_\psi c_\phi & c_\theta c_\phi \end{vmatrix} \tag{25}$$

The damping component is the linear matrix

$$D(\nu) = D_0\nu = \begin{vmatrix} D_{0,u}u \\ D_{0,v}v \\ D_{0,w}w \\ D_{0,p}p \\ D_{0,q}q \\ D_{0,r}r \end{vmatrix} \tag{26}$$

The forces and torques generated by the actuators are assumed to be linear in the input,

$$\tau_c(u) = u = \begin{vmatrix} 0 \\ 0 \\ u_{trottle} \\ u_{roll} \\ u_{pitch} \\ u_{yaw} \end{vmatrix} \tag{27}$$

$u_{trottle}$ is the control input of the trust force as the force affecting the velocity w in the z-direction. The force of trust gives a lifting power that makes the drone flies and depends on the sum of the speed of the four propellers. Since the rotors are fixed their total thrust will always pull upwards along the z-axis of the body frame.

$$u_{trottle} = k_f \sum_{i=1}^{4} \Omega_i^2 = k_f(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \tag{28}$$

where i is the number of motors (propeller), Ω_i is the speed of motor i and, k_f is thrust constant. Next, by increasing or decreasing the speed of the four rotors independently, it will create torques around the x-y-z axes and thus create roll-,pitch- and yaw-rotations. By always decreasing the speed of one rotor as much as increasing the speed of another the total thrust is retained. Therefore, the control input of roll u_{roll} and u_{pitch} are related to the torque and can be obtained by multiplying the force by the distance and the rotors will affect the total rotation about a certain axis differently depending on the distance from the center of gravity(CoG) of the drone.

$$u_{roll} = k_f l(-\Omega_2^2 + \Omega_4^2) \tag{29}$$

$$u_{pitch} = k_f l(\Omega_1^2 - \Omega_3^2) \tag{30}$$

$$u_{yaw} = k_M(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \tag{31}$$

where k_f is the coefficient of the force affecting the velocity w in the z-direction related to the thrust constant of the drone. Meanwhile, l is the distance between the axis of rotation of each rotor to the origin of the body reference frame which should coincide with the CoG of the drone and k_M is the moment constants, respectively.

III. Development of APID for Visual based Navigation

In order to maneuver the drone autonomously, based on the position and orientation of the tag marker to the drone body frame, a closed-loop control system with APID is occupied, in which its parameters are tuned by an Adjustment Mechanism (AM) [20], [21]. PID has been regarded as one of the most popular controllers in the industry because of its ease of implementation, and efficient performance [22], [23], [24]. Fig. 2 shows the control system of AprilTag navigation.

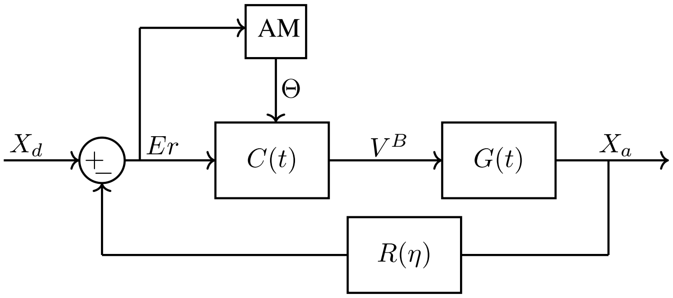


Fig. 2. Block Diagram of the Control System.

The output of AM, $\Theta = [K_p \quad K_i \quad K_d]$, is a matrix of APID parameters. The steady-State error is determined as the difference between the actual and desired position of the drone, given as follows:

$$E_r = X_d - X_a \tag{32}$$

where $Er = [er_x \quad er_y \quad er_z \quad er_\phi]^T$

is the error matrix; $X_d = [x_d \quad y_d \quad z_d \quad \phi_d]^T$

is desired matrix; $X_a = [x_a \quad y_a \quad y_a \quad \phi_a]^T$

is the actual path matrix. The output of the APID is the velocity of the body frame, which is influenced by the proportion, integral, and derivative of the steady-State error. The mathematical model of the controller is expressed as follows:

$$C(t) = K_p E_r + K_i \int E_r dt + K_d \frac{dE_r}{dt} \tag{33}$$

$G(t)$ is the equation of motion for the body frame of the drone regarding the earth frame as a rigid body. The actual data is captured by the camera, which is attached to the body of the drone. Because the coordinate of the AprilTag marker $\{t_i\}$ and the drone's body frame $\{B\}$ does not match, a rotation matrix is needed to align the axes of the tag marker with the axes of drone movement [25], as follows:

$$R_t^B = -R_t^B(\theta) R_t^B(\phi) \tag{34}$$

,

where $R_t^B(\theta) \in R^3$ and $R_t^B(\phi) \in R^3$ are pitch and roll rotation matrix of tag frame $\{t\}$ to drone body frame $\{B\}$. Fig. 3 represents schematic of drone navigation based on tag marker.

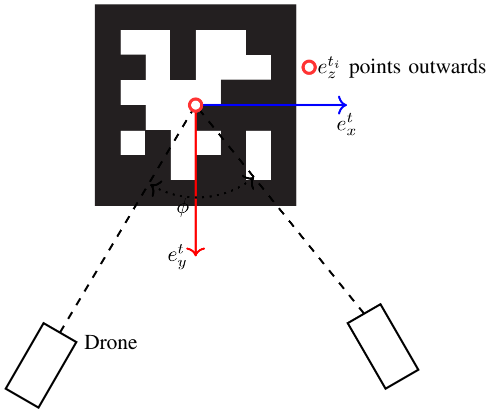


Fig. 3. Schematic of Drone Navigation based on Tag Marker.

The parameters of APID controller are tuned in such a way to prevent overshoot to avoid drone collision and achieve it slowly. Therefore AM is introduced for tuning of APID controller, in which three rules for distances and angle have been determined. Three sets of APID parameters have been established as follows:

$$\Theta_{s1} = [0.05 \quad 0.01 \quad 0.009] \tag{35}$$

$$\Theta_{s2} = [0.1 \quad 0.05 \quad 0.01] \tag{36}$$

$$\Theta_{s3} = [0.2 \quad 0.1 \quad 0.02] \tag{37}$$

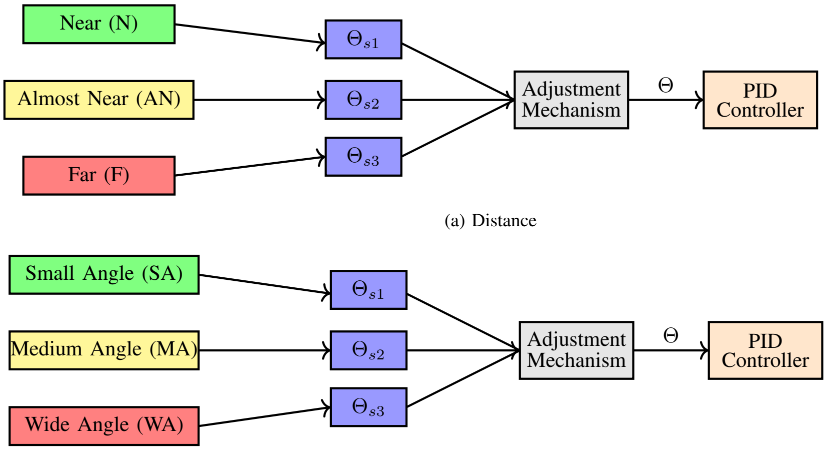
Table I represents the rules for AM to tune APID controllers and sets of APID parameters and related error range.

TABLE I. Rules of AM for APID Parameters and Error Ranges

Error range	Category	PID parameters
$0(m) \leq er_{x,y,z} < 0.2(m)$	N	Θ_{s1}
$0.2(m) \leq er_{x,y,z} < 1(m)$	AN	Θ_{s2}
$1(m) \leq er_{x,y,z} < D_{max}(m)$	F	Θ_{s3}
$\lvert er_\phi \rvert \leq 20^o$	SA	Θ_{s1}
$\lvert er_\phi \rvert \geq 20^o \quad \& \quad \lvert er_\phi \rvert \leq 45^o$	MA	Θ_{s2}
$\lvert er_\phi \rvert \geq 20^o \& \lvert er_\phi \rvert \geq 45^o \& \lvert er_\phi \rvert \leq A^o_{max}$	WA	Θ_{s3}

$D_{max}(m)$ and A^o_{max} are the maximum distance and angle that can be recognized by AprilTag system. Fig. 4 represents a flowchart of rules for AM to adjust the APID parameters.

The scenario of the navigation algorithm is divided into four stages. Firstly the control system is applied to uplift



(a) Distance

(b) Angle

Fig. 4. Flowchart of Rules for Adjustment Mechanism.

the drone upward, z-axis. After the drone reached a certain desired height, it rotated at a yaw angle until desired tolerance error range is obtained. Then it moved to its side direction of the y-axis. The last stage is the forward direction, the x-axis. Whenever the drone carries out the defined tolerance in all the stages, the drone will be in the desired position. Algorithm 1 exhibits the pseudo-code, which is the scenario of the navigation system. μ is the error tolerance.

Algorithm 1 Pseudo Code of Navigation Algorithm

```
Start
Take off
Navigate drone in z-axis
if $e_z < \mu_z$ :
    Navigate drone in yaw-axis
if $e_z < \mu_z$ & $e_{yaw} < \mu_{yaw}$ :
    Navigate drone in y-direction
if $e_z < \mu_z$ & $e_{yaw} < \mu_{yaw}$ & $e_y < \mu_y$ :
    Navigate drone in x-axis
if $e_z < \mu_z$ & $e_y < \mu_y$ & $e_{yaw} < \mu_{yaw}$ & $e_x < \mu_x$ :
    Land
End
```

In the scenario of the navigation system, the trajectory is presented to lead the drone to reach its target point. Therefore, a desired path trajectory to the target can be predicted to avoid potential crashes. In addition, the velocity of the drone is determined as the angular velocity of the propellers that are dependent on body frame velocity, V^B.

IV. Results and Discussion

In order to validate the navigation system, a model of the drone was created in a virtual environment called Gazebo integrated with Robot Operating System (ROS) [26], [27], [28].

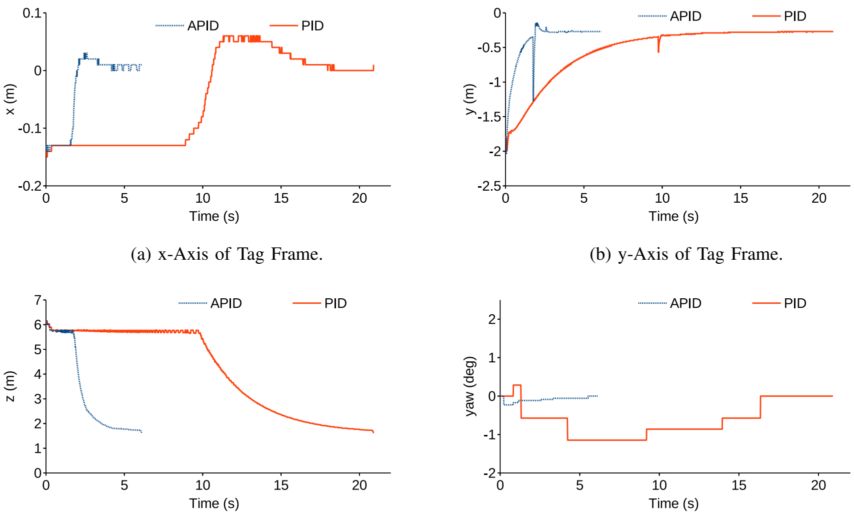
To validate the performance of the tag-based navigation algorithm, a multi-target navigation strategy is utilized to move the drone from the home point to the destination point. The multi-target navigation strategy is divided into three states. Each stage is marked by a tag with a different ID and the drone hovers in front of them before moving to the next stage. When the drone hovers in front of the first tag marker with the same procedure of hovering navigation, it rotates toward the next marker tag to face the next tag marker. This trend is followed for the next stages with different tag marker ID until the drone reach and land at the destination point. Therefore, the drone follows the defined trajectory to reach the target point without demanding the saved map in indoor/outdoor environment. Algorithm 2 represents the pseudo code of multi-target navigation.

Algorithm 2 Pseudo Code of Multi-Target Navigation Algorithm

```
Start
Take off
Go forward to tag marker #1
Hover in front of tag marker #1
If $e_x \leq \zeta_x$ and $e_y \leq \zeta_y$ and $e_z \leq \zeta_z$ and $e_\phi \leq \zeta_\phi$:
    Rotate 90°
If $e_\phi \leq \zeta_\phi$:
    Go forward to tag marker #2
If $e_x \leq \zeta_x$ and $e_y \leq \zeta_y$ and $e_z \leq \zeta_z$ and $e_\phi \leq \zeta_\phi$:
    Rotate $-90°$
If $e_\phi \leq \zeta_\phi$:
    Go forward to tag marker #3
If $e_x \leq \zeta_x$ and $e_y \leq \zeta_y$ and $e_z \leq \zeta_z$ and $e_\phi \leq \zeta_\phi$:
    Land
End
```

Fig. 5 represents the comparison of the proposed navigation systems powered by APID and convention PID controller with constant parameters.

The results show that the navigation system enriched with APID converged faster to the target point significantly. Because it uses multiple sets of parameters for controlling based on the error range. When the error is in the F range, the drone moves



(a) x-Axis of Tag Frame.

(b) y-Axis of Tag Frame.

(c) z-Axis of Tag Frame.

(d) ϕ of the Tag Frame.

Fig. 5. Comparison of APID and PID Performance.

faster to the target than it is in N and AN ranges.

Fig. 6 shows the trajectory of the drone in x and y direction from the home point to the target point. The drone took off in the home point moved forward to tag #1, hovered in front of it, then rotated to the second tag. Similarly, the drone moved to the destination point.

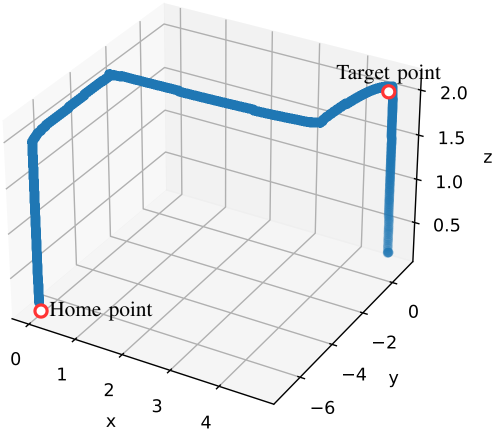


Fig. 6. Multi-Target Navigation.

The trajectory error and APID parameters changes are represented in Fig. 7 and 8, respectively.

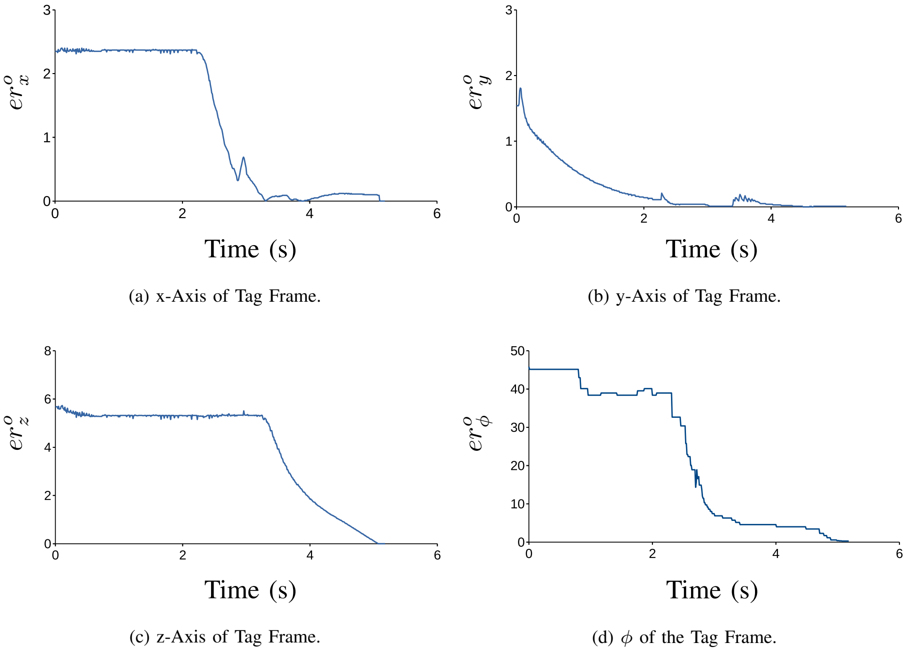
In Fig. 8, it is represented that if the error is in the range of each set, the adjustment mechanism sets the set of parameters for the APID controller. For instance, in Fig. 8a between starting and 2.6s the APID parameters are the first parameters set, Θ_{s3}, because the error is more than 1m. Subsequently, when the error is between 0.2m and 1m, from 2.6s to 3.16s, Θ_{s2} is set. Finally, after 3.16s until the drone reached the target, the APID parameters follow Θ_{s1}. In Fig. 8b, the switching time for APID parameter sets are 0.36 s and 1.57s. Furthermore, in Fig. 8c, at 4.46s the APID parameters turned from Θ_{s3} to Θ_{s2}, and it turned to Θ_{s1} at 4.94s, when the error reduced from more than 1m to less than 1m and then decreased to less than 0.2m, respectively. In Fig. 8d, the turning times for APID parameters are 0.8s and 2.7s based on the ranges of the error shown in Fig. 7d. By considering the turning times, it can be obvious that turning times are in ascending order. The first turning times are 0.39s, 0.8s, 2.64 s, and 4.46s, for y, yaw, x, and z direction, respectively. Similarly, the second turning times are 1.57s, 2.7s, 3.16s, and 4.46s, for the y, yaw, x, and z direction, respectively. This mimics the scenario of the navigation system as shown in Algorithm 1.

V. Conclusion

The paper presented a visual navigation system for an autonomous drone that hovers and navigates by using tag marker position and orientation autonomously. The kinematic and dynamic models of a drone have been determined by Newton- Euler. Moreover, AprilTag marker tag was introduced to obtain the position and orientation of the drone.

The results showed the navigation system using APID controller performed four times faster than the navigation system powered by the conventional PID controller. In addition, it demonstrated higher accuracy and reliable performance of the algorithms to accomplish various designed trajectories.

Besides its reliability and accuracy, one of the APID's advantages is less computational time rather than complex and memory consumable controllers. In addition, the tag navigation system and APID utilized an onboard camera available and usable for the light drones that cannot carry heavy sensors



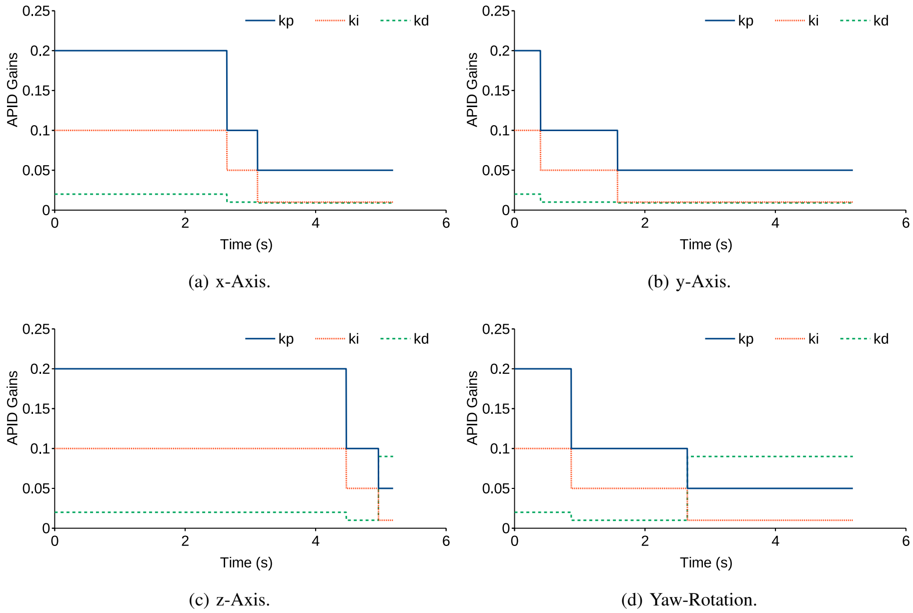
(a) x-Axis of Tag Frame.

(b) y-Axis of Tag Frame.

(c) z-Axis of Tag Frame.

(d) ϕ of the Tag Frame.

Fig. 7. Trajectory Error based on the Camera Frame.



(a) x-Axis.

(b) y-Axis.

(c) z-Axis.

(d) Yaw-Rotation.

Fig. 8. Changes of the APID Controller's Parameters.

such as lase sensor for navigation and localization. Although it can be ascertained that the proposed method is efficient, there are some limitations. For instance, in an outdoor environment, the shininess of the tag marker can affect the view of the camera and navigation system. Moreover, the tag marker should be in the camera view for the drone to follow the desired trajectory. In future work, fuzzy and adaptive control systems can be investigated for visual-based navigation. In addition, this proposed algorithm can be validated for drone navigation based on a laser sensor.

Acknowledgment

The authors would like to thank Universiti Kebangsaan Malaysia (UKM) and the Ministry of Higher Education Malaysia for the financial support received under research grant KK-2020-014 and TAP-K014062.

REFERENCES

[1] C. Alex and A. Vijaychandra, "Autonomous cloud based drone system for disaster response and mitigation," *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, pp. 1–4, 2016.

[2] A. Sagitov, K. Shabalina, L. Sabirova, and H. L. andEvgeni Magid, "Artag, apriltag and caltag fiducial marker systems: Comparison in a presence of partial marker occlusion and rotation," *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics*, vol. 2, pp. 192–191, 2017.

[3] P.-J. Bristeau, F. Callou, D. Vissière, and N. Petit, "The navigation and control technology inside the ar.drone micro uav," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 1477–1484, 2011.

[4] Y. Ham, K. K. Han, J. Lin, and M. Golparvar-Fard, "Visual monitoring of civil infrastructure systems via camera-equipped unmanned aerial vehicles (uavs): a review of related works," *Visualization in Engineering*, vol. 4, no. 1, pp. 1–8, 2016.

[5] R. Casado and A. Bermudez, "A simulation framework for developing autonomous drone navigation systems," *Electronics*, vol. 10, no. 1, p. 7, 2021.

[6] A. T. Azar, A. Koubaa, N. A. Mohamed, H. A. Ibrahim, Z. F. Ibrahim, M. Kazim, A. Ammar, B. Benjdira, A. M. Khamis, I. A. Hameed, and G. Casalino, "Drone deep reinforcement learning: A review," *Electronics*, vol. 10, no. 9, p. 999, 2021.

[7] Y.-P. Huang, L. Sithole, and T.-T. Lee, "Structure from motion technique for scene detection using autonomous drone navigation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 12, pp. 2559–2570, 2019.

[8] E. Cetin, C. Barrado, G. Munoz, M. Macias, and E. Pastor, "Drone navigation and avoidance of obstacles through deep reinforcement learning," *IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, pp. 1–7, 2019.

[9] X. Li and X. Yang, "Stability analysis for nonlinear systems with state-dependent state delay," *Automatica*, vol. 112, p. 108674, 2020.

[10] S. Zahran, A. Moussa, and N. El-Sheimy, "Enhanced drone navigation in gnss denied environment using vdm and hall effect sensor," *ISPRS International Journal of Geo-Information*, vol. 8, no. 4, p. 169, 2019.

[11] J. Upadhyay, A. Rawat, and D. Deb, "Multiple drone navigation and formation using selective target tracking-based computer vision," *Electronics*, vol. 10, no. 17, p. 2125, 2021.

[12] V. Hodge, R. Hawkins, and R. Alexander, "Deep reinforcement learning for drone navigation using sensor data," *Neural Comput & Applic*, vol. 33, pp. 2015–2033, 2021.

[13] V. R. F. Miranda, A. M. C. Rezende, T. L. Rocha, H. Azpurua, L. C. A. Pimenta, and G. M. Freitas, "Autonomous navigation system for a delivery drone," *J Control Autom Electr Syst*, vol. 33, pp. 141–155, 2022.

[14] D. Tang, T. Hu, L. Shen, Z. Ma, and C. Pan, "Apriltag array-aided extrinsic calibration of camera–laser multi-sensor system," *Robotics and Biomimetics*, vol. 3, no. 1, 2016.

[15] D. Malyuta, C. Brommer, D. Hentzen, T. Stastny, R. Siegwart, and B. Roland, "Long-duration fully autonomous operation of rotorcraft unmanned aerial systems for remote-sensing data acquisition," *Journal of Field Robotics*, vol. 37, no. 1, pp. 137–157, 2020.

[16] T. Lee, S. Mckeever, and J. Courtney, "Flying free: A research overview of deep learning in drone navigation autonomy," *Drones*, vol. 5, no. 2, p. 52, 2021.

[17] H. Yang, Y. Lee, S. Y. Jeon, and D. Lee, "Multi-rotor drone tutorial: systems, mechanics, control and state estimation," *Intelligent Service Robotics*, vol. 10, no. 2, pp. 79–93, 2017.

[18] A. Ajami, J.-P. Gauthier, and L. Sacchelli, "Dynamic output stabilization of control systems: An unobservable kinematic drone model," *Automatica*, vol. 125, p. 109383, 2021.

[19] K. V. Rao and A. T. Mathew, "Dynamic modeling and control of a hexacopter using pid and back stepping controllers," *4th International Conference on Power, Signals, Control and Computation*, pp. 1–7, 2018.

[20] M. S. Amiri, R. Ramli, and M. F. Ibrahim, "Initialized model reference adaptive control for lower limb exoskeleton," *IEEE Access*, vol. 7, pp. 167 210–167 220, 2019.

[21] C. Conker and M. K. Baltacioglu, "Fuzzy self-adaptive pid control technique for driving hho dry cell systems," *International Journal of Hydrogen Energy*, vol. 45, no. 49, pp. 26 059–26 069, 2020.

[22] J. J. Castillo-Zamora, K. A. Camarillo-Gomez, G. I. Perez-Soto, and J. Rodriguez-Resendiz, "Comparison of pd, PID and sliding-mode position controllers for v-tail quadcopter stability," *IEEE Access*, vol. 6, pp. 38 086–38 096, 2018.

[23] M. S. Amiri, R. Ramli, and M. F. Ibrahim, "Genetically optimized parameter estimation of mathematical model for multi-joints hip–knee exoskeleton," *Robotics and Autonomous Systems*, vol. 125, p. 103425, 2020.

[24] D. Lee, S. J. Lee, and S. C. Yim, "Reinforcement learning-based adaptive pid controller for dps," *Ocean Engineering*, vol. 216, p. 108053, 2020.

[25] S. M. Abbas, S. Aslam, K. Berns, and A. Muhammad, "Analysis and improvements in apriltag based state estimation," *Sensors*, vol. 19, no. 24, pp. 1–32, 2019.

[26] M. S. Amiri and R. Ramli, "Intelligent trajectory tracking behavior of a multi-joint robotic arm via genetic–swarm optimization for the inverse kinematic solution," *Sensors*, vol. 21, no. 9, p. 3171, 2021.

[27] M. S. Amiri, R. Ramli, M. F. Ibrahim, D. A. Wahab, and N. Aliman, "Adaptive Particle Swarm Optimization of PID Gain Tuning for Lower-Limb Human Exoskeleton in Virtual Environment," *Mathematics*, vol. 8, no. 11, p. 2040, 2020.

[28] K. A. Juhari, R. Ramli, S. M. Haris, Z. Ibrahim, and A. Z. Mohamed, "Development of Floor Mapping Mobile Robot Algorithm Using Enhanced Artificial Neuro-Based SLAM (ANBS)," *Jurnal Kejuruteraan*, vol. 3, no. 1, pp. 59–64, 2020.