

Pengembangan *Low-cost Quadrotor* dengan Misi *Waypoint Tracking* Berbasis Pengendali PID

ADNAN RAFI AL TAHTAWI¹, ERICK ANDIKA², MAULANA YUSUF³,
WILDAN NURFAUZAN HARJANTO⁴

¹Jurusan Teknik Elektro, Politeknik Negeri Bandung, Indonesia
^{2,3,4}Program Studi Teknik Komputer, Politeknik Sukabumi, Indonesia
Email: adnan.raf@polban.ac.id

Received 14 November 2019 | *Revised* 30 November 2019 | *Accepted* 9 Desember 2019

ABSTRAK

Quadrotor merupakan salah satu jenis pesawat tanpa awak yang dapat bekerja secara autonomous dalam melaksanakan tugasnya. Beberapa quadrotor komersial yang dapat bekerja secara autonomous memiliki harga yang relatif mahal. Penelitian ini bertujuan untuk merancang dan mengembangkan sebuah quadrotor berbiaya rendah dengan misi waypoint tracking berbasis pengendali PID. Quadrotor dirancang menggunakan modul Ardupilot Mega 2.6 sebagai pengendali terbang. Modul ini cocok digunakan untuk pengembangan sebuah quadrotor berbiaya rendah karena telah terintegrasi dengan sebuah antarmuka dan panel pengendali PID. Misi waypoint tracking dengan pengendali PID digunakan untuk menguji kinerja dari quadrotor yang dirancang. Hasil pengujian menunjukkan bahwa quadrotor dapat terbang secara autonomous pada dua buah skenario waypoint. Pengendali PID yang digunakan mampu menstabilkan sikap terbang quadrotor dengan maksimum overshoot kurang dari 4°.

Kata kunci: *Pengendali PID, sikap terbang, autonomous, waypoint, quadrotor*

ABSTRACT

Quadrotor is one type of unmanned aircraft that can work autonomously in carrying out its duties. Some commercial quadrotor that can work autonomously have a relatively expensive price. This study aims to design and develop a low-cost quadrotor with a waypoint tracking mission based on PID controller. Quadrotor is designed using the Ardupilot Mega 2.6 module as flight controller. This module is suitable for the development of a low-cost quadrotor because it has been integrated with an interface and PID controller panel. The waypoint tracking mission with PID controller is used to test the performance of the designed quadrotor. The test results show that quadrotor can fly autonomously in two waypoint scenarios. The PID controller used is able to stabilize the quadrotor attitude with a maximum overshoot less than 4°.

Keywords: *PID controller, flight attitude, autonomous, waypoint, quadrotor*

1. PENDAHULUAN

Quadrotor merupakan salah satu jenis pesawat tanpa awak yang memiliki empat buah rotor sebagai penggerak utamanya. Quadrotor memiliki kelebihan jika dibandingkan pesawat tanpa awak sayap tetap karena dapat melakukan lepas landas secara vertikal. Kemampuan ini biasa dikenal dengan istilah *Vertical Take-off and Landing* (VTOL) (**Cabecinhas, Naldi, Marconi, Silvestre, & Cunha, 2012**). Teknologi quadrotor sampai saat ini telah berkembang sangat pesat. Beberapa quadrotor komersial bermunculan dengan fitur-fitur yang sangat canggih. Bahkan terdapat beberapa yang memiliki fitur kecerdasan seperti *gesture control* (**Zhang, Li, & Zhang, 2017**). Pengembangan fitur ini biasanya disesuaikan dengan kebutuhan dan fungsi quadrotor itu sendiri seperti fotografi, pemetaan, dan pengawasan. Quadrotor komersial dengan fitur canggih tersebut biasanya memiliki harga yang relatif mahal. Oleh karena itu, merancang sebuah quadrotor berbiaya rendah dapat menjadi alternatif solusi dalam pengembangan teknologi quadrotor.

Dari sisi pengendalian, quadrotor dapat dikendalikan jarak jauh oleh seorang pilot atau mampu mengendalikan dirinya sendiri (*autonomous*). Pengendalian secara *autonomous* memerlukan metode sistem kendali agar sikap terbang quadrotor stabil saat melaksanakan sebuah misi. Beberapa metode pengendalian quadrotor dapat diterapkan seperti PID (Proporsional-Integral-Derivatif) (**Li & Li, 2011**), fuzzy logic (**Indrawati, Prayitno, & Kusuma, 2015**), LQR (**Valeria, Caldera, Lara, & Guichard, 2013**), *neural network* (**Boudjedir, 2012**), dan *robust control* (**Bai, Liu, Shi, & Zhong, 2012**). Setiap metode memiliki kelebihan dan kekurangan masing-masing dalam menghasilkan kestabilan sikap terbang quadrotor. Namun demikian, pengendalian dengan metode yang sederhana dan mudah diimplementasikan sangat diperlukan ketika sebuah quadrotor akan diterbangkan secara *autonomous*.

Metode terbang secara *autonomous* pada quadrotor dapat dikembangkan dengan menentukan titik-titik tujuan yang akan dituju oleh quadrotor. Teknik ini dikenal dengan istilah *waypoint tracking*. Metode ini biasanya digunakan untuk tugas-tugas tertentu seperti pemantauan kualitas udara (**Villa, Gonzalez, Miljevic, Ristovski, & Morawska, 2016**). Quadrotor harus mampu menuju ke beberapa titik koordinat yang telah ditetapkan secara *autonomous*. Pengembangan metodologi pengendalian sikap terbang quadrotor untuk misi *waypoint tracking* juga telah dikembangkan. Penelitian (**Hussein, Al-Kaff, De La Escalera, & Armingol, 2015**) merancang pengendali PID melalui simulasi MATLAB/Simulink untuk mengendalikan quadrotor bergerak ke beberapa titik secara *autonomous* di lingkungan *indoor*. Skenario yang digunakan adalah dengan merancang sebuah *path* yang nantinya akan dijejaki oleh quadrotor. Metode PID ini juga dapat dikombinasikan dengan metode LQ (**Reizenstein, 2017**) dan fuzzy logic (**Qian, Pebrianti, Chun, Hao, & Bayuaji, 2017**) untuk pengendalian quadrotor berbasis *waypoint tracking*. Pendekatan lain yang dapat digunakan adalah metode algoritma genetika seperti yang telah dikembangkan (**Tran & Nguyen, 2018**). Metode ini dapat digunakan untuk mengoptimasi parameter pengendali seperti PID agar diperoleh kinerja pengendalian yang optimal. Lebih lanjut lagi, penelitian (**Pham, La, Feil-Seifer, & Nguyen, 2018**) menggunakan metode *Reinforcement Learning* (RL) untuk pengendalian quadrotor secara *autonomous* agar mampu beradaptasi dengan kondisi lingkungan.

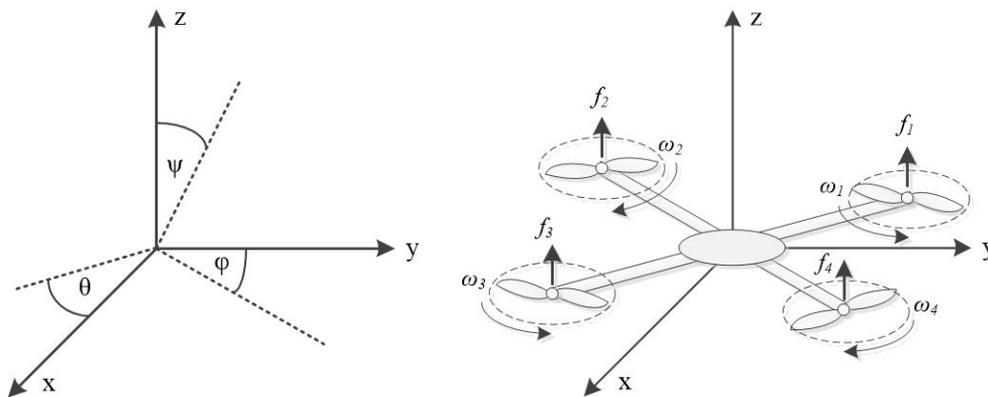
Tujuan dari penelitian ini adalah untuk merancang dan mengembangkan sebuah quadrotor berbiaya rendah dengan misi *waypoint tracking* berbasis pengendali PID. Penelitian ini

menggunakan *flight controller* Ardupilot Mega (APM) 2.6 dimana pengendali PID diimplementasikan. Modul ini sangat cocok digunakan untuk pengembangan sebuah quadrotor karena telah dilengkapi dengan sebuah antarmuka dan fitur kendali. Misi *waypoint tracking* diberikan kepada quadrotor yang dirancang dengan pengendali PID untuk menstabilkan sikap terbangnya. Beberapa titik koordinat yang diperoleh dari GPS digunakan sebagai referensi trajektori perjalanan quadrotor. Quadrotor yang dirancang harus mampu secara *autonomous* menuju ke titik koordinat yang telah ditentukan. Berbeda dengan penelitian yang telah dilakukan sebelumnya, penelitian ini merancang dan mengembangkan sebuah quadrotor dengan kendali PID secara eksperimen untuk kebutuhan *waypoint tracking*. Penelitian-penelitian sebelumnya lebih menekankan pada pengembangan metode kendali dan diuji pada lingkungan simulasi, sedangkan pada penelitian ini lebih ditekankan kepada implementasi perangkat keras. Penelitian ini merupakan pengembangan dari penelitian sebelumnya dimana pengendali PID dirancang dan ditanamkan dalam sebuah *flight controller* untuk mengendalikan sikap terbang quadrotor (Al Tahtawi & Yusuf, 2017) (Al Tahtawi & Yusuf, 2019). Pengendali PID digunakan karena telah terintegrasi pada perangkat antarmuka APM 2.6. Selain itu, pengendali PID lebih mudah dirancang dan diimplementasikan jika dibandingkan dengan jenis pengendali lainnya.

2. METODE PENELITIAN

2.1 Model Dinamika Quadrotor

Quadrotor dalam konfigurasi "x" memiliki model dinamika dan diagram gaya seperti ditunjukkan pada Gambar 1 (Al Tahtawi & Yusuf, 2019).



Gambar 1. Model Dinamika Quadrotor

Sisi kiri pada model tersebut merepresentasikan kerangka koordinat bumi termasuk sudut sikap pergerakan quadrotor terhadap kerangka tersebut. Sisi kanan menunjukkan gaya-gaya yang terdapat pada tubuh quadrotor yang terdiri dari kecepatan, torsi, dan gaya vertikal. Berdasarkan model tersebut, maka diperoleh *state* (q) pada quadrotor sebagai berikut (Luukkonen, 2011):

$$q = [\xi \quad \eta]^T = [x \quad y \quad z \quad \phi \quad \theta \quad \psi]^T \quad (1)$$

dengan ξ adalah vektor orientasi quadrotor terhadap kerangka bumi x , y , dan z , sedangkan η adalah vektor sikap terbang quadrotor yang dikenal dengan istilah *roll* (ϕ), *pitch* (θ), dan

yaw (ψ). Berdasarkan referensi (**Luukkonen, 2011**), model dinamika quadrotor secara umum dapat ditulis sebagai berikut:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{U_z}{m} \begin{bmatrix} C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi S_\theta C_\phi - C_\psi S_\phi \\ C_\theta C_\phi \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & \dot{\phi} C_\phi T_\theta + \dot{\theta} S_\phi / C_\theta^2 & -\dot{\phi} S_\phi C_\theta + \dot{\theta} C_\phi / C_\theta^2 \\ 0 & -\dot{\phi} S_\phi & -\dot{\phi} C_\phi \\ 0 & \dot{\phi} C_\phi / C_\theta + \dot{\phi} S_\phi T_\theta / C_\theta & -\dot{\phi} S_\phi / C_\theta + \dot{\theta} C_\phi T_\theta / C_\theta \end{bmatrix} v + W_\eta^{-1} \dot{v} \quad (3)$$

dengan g adalah percepatan gravitasi, m adalah massa quadrotor, C_x adalah $\cos(x)$, S_x adalah $\sin(x)$, dan T_x adalah $\tan(x)$. U_z adalah gaya angkat quadrotor yang memenuhi persamaan berikut:

$$U_z = \sum_{i=1}^4 f_i = k \sum_{i=1}^4 \omega_i^2 \quad (4)$$

dengan f_i adalah gaya angkat untuk motor ke- i , k adalah konstanta angkat, dan ω_i adalah kecepatan sudut motor ke- i . v adalah kecepatan sudut pada kerangka tubuh quadrotor, \dot{v} adalah percepatan sudutnya, dan W_η^{-1} adalah matriks transformasi yang memenuhi persamaan berikut:

$$v = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = W_\eta \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (5)$$

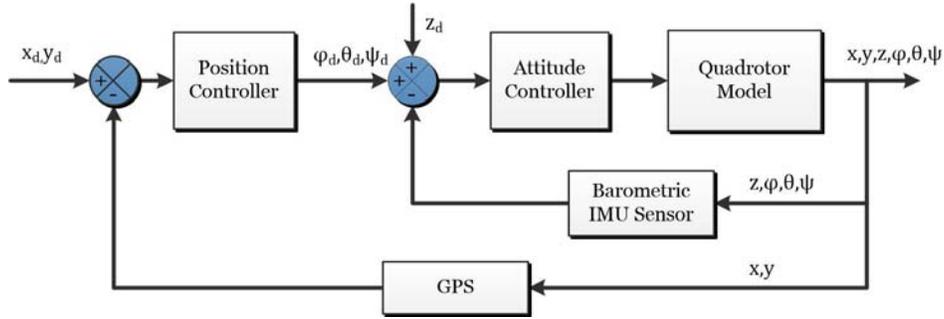
$$\dot{v} = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})qr / I_{xx} \\ (I_{zz} - I_{xx})pr / I_{yy} \\ (I_{xx} - I_{yy})pq / I_{zz} \end{bmatrix} - I_r \begin{bmatrix} q / I_{xx} \\ -p / I_{yy} \\ 0 \end{bmatrix} \omega_\Gamma + \begin{bmatrix} U_\phi / I_{xx} \\ U_\theta / I_{yy} \\ U_\psi / I_{zz} \end{bmatrix} \quad (6)$$

$$W_\eta^{-1} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi / C_\theta & C_\phi / C_\theta \end{bmatrix} \quad (7)$$

dengan I_{xx} , I_{yy} , dan I_{zz} adalah momen inersia pada masing-masing sumbu x , y , dan z , I_r adalah momen inersia total, $\omega_\Gamma = \omega_1 - \omega_2 + \omega_3 - \omega_4$ adalah kecepatan motor untuk pergerakan vertikal, U_ϕ adalah gaya untuk sikap *roll*, U_θ adalah gaya untuk sikap *pitch*, dan U_ψ adalah gaya untuk sikap *yaw*.

2.2 Sistem Kendali Quadrotor

Secara umum sistem kendali pada sebuah quadrotor yang diadopsi dari (Qian et al., 2017) dapat dilihat pada Gambar 2. Seperti yang telah disebutkan di bagian pemodelan, quadrotor memiliki enam buah *state* yang dikelompokkan ke dalam dua bagian yaitu posisi (x, y, z) dan sikap (ϕ, θ, ψ). Dengan demikian pengendalian quadrotor pada dasarnya dilakukan pada posisi dan sikap.



Gambar 2. Diagram Blok Sistem Kendali Quadrotor

Posisi quadrotor dalam koordinat (x, y) merepresentasikan data longitude dan latitude yang dapat dideteksi oleh sensor GPS. Ketinggian quadrotor (z) dapat dideteksi menggunakan sensor Barometric yang biasanya telah terintegrasi dalam sebuah *flight controller*. Untuk pengendalian sikap quadrotor digunakan sensor IMU yang mampu mendeteksi sudut dan kecepatan sudut sikap *roll* (ϕ), *pitch* (θ), dan *yaw* (ψ). Dari sisi pengendali, quadrotor memiliki dua level pengendalian yaitu pengendali posisi dan pengendali sikap. Pengendali posisi dengan *setpoint* posisi longitude dan latitude dan akan menghasilkan *setpoint* sudut sikap quadrotor untuk mencapai posisi tersebut. Pada level pengendalian sikap, *setpoint* sudut sikap yang dihasilkan akan dikendalikan dan akan menghasilkan sinyal kendali untuk diberikan ke aktuator. Sinyal kendali ini dihasilkan dalam bentuk tegangan untuk menggerakkan keempat rotor penggerak quadrotor yang memiliki persamaan sebagai berikut:

$$U_{\phi} = lk(-\omega_2^2 + \omega_4^2) \quad (8)$$

$$U_{\theta} = lk(-\omega_1^2 + \omega_3^2) \quad (9)$$

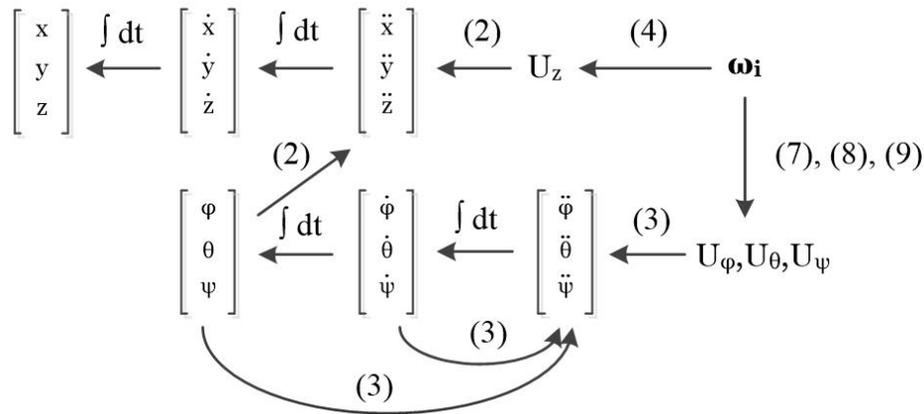
$$U_{\psi} = lk(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \quad (10)$$

dengan U_{ϕ} adalah sinyal kendali untuk sikap *roll*, U_{θ} adalah sinyal kendali untuk sikap *pitch*, U_{ψ} adalah sinyal kendali untuk sikap *yaw*, dan l adalah panjang lengan quadrotor.

2.3 Kendali Waypoint Tracking

Pengendalian *waypoint tracking* bertujuan untuk menggerakkan quadrotor dari satu titik ke beberapa titik lainnya secara *autonomous*. Kendali ini diperoleh dengan cara mengatur kecepatan masing-masing rotor sehingga quadrotor menghasilkan sikap *roll*, *pitch*, dan *yaw* yang akan membuat trajektori quadrotor menuju titik-titik yang telah ditentukan. Untuk menghasilkan pengendalian yang optimal sangat sulit dilakukan mengingat sistem dinamik quadrotor yang relatif kompleks. Dengan demikian, metode kendali yang sederhana cukup untuk dapat digunakan pada quadrotor untuk melakukan misi ini.

Secara umum, pengendalian *waypoint tracking* dapat bekerja melalui interaksi antara *state*, turunan *state*, dan sinyal kendali. Konsep kendali *waypoint tracking* dalam bentuk *state flow* yang telah dimodifikasi dari (Luukkonen, 2011) dapat dilihat pada Gambar 3 berikut.



Gambar 3. Diagram Hubungan Antara *State* pada Quadrotor

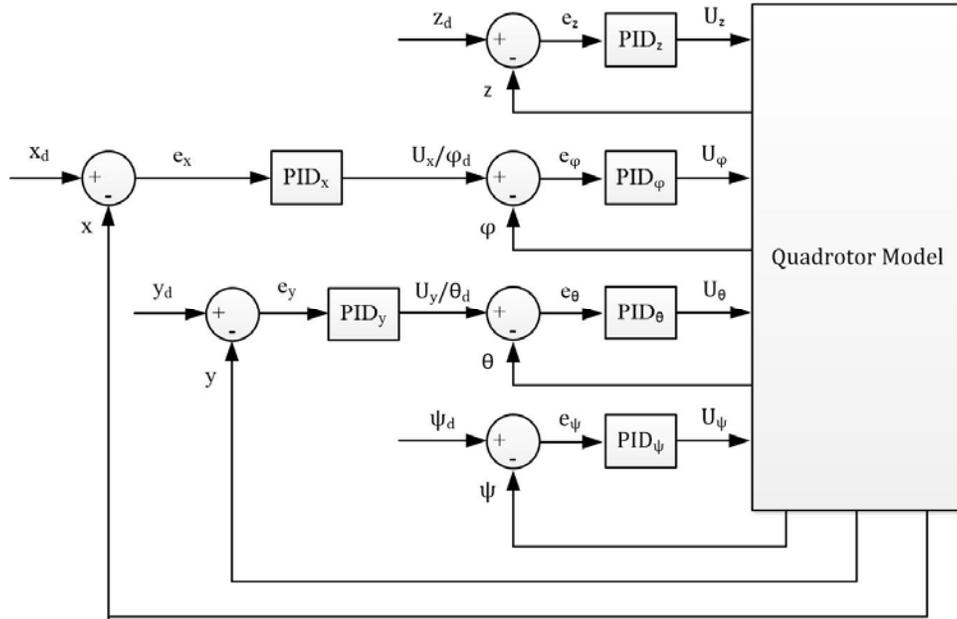
Kecepatan putaran rotor ω_i pada quadrotor akan menghasilkan sinyal kendali berupa gaya angkat U_z seperti pada Persamaan (4) dan torsi pada ketiga sikap quadrotor (U_ϕ, U_θ, U_ψ) seperti pada Persamaan (8), Persamaan (9), dan Persamaan (10). Gaya angkat U_z akan mempengaruhi percepatan translasi dari quadrotor $[\ddot{x} \ \ddot{y} \ \ddot{z}]^T$ yang juga akan mempengaruhi kecepatan translasi $[\dot{x} \ \dot{y} \ \dot{z}]^T$ dan posisi quadrotor $[x \ y \ z]^T$ sesuai dengan Persamaan (2). Torsi pada ketiga sikap quadrotor (U_ϕ, U_θ, U_ψ) akan mempengaruhi percepatan sudut sikap quadrotor $[\ddot{\phi} \ \ddot{\theta} \ \ddot{\psi}]^T$ yang juga akan mempengaruhi kecepatan sudut sikap $[\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ dan sudut sikap itu sendiri $[\phi \ \theta \ \psi]^T$ seperti pada Persamaan (3). Berdasarkan hubungan setiap *state* tersebut, pengendalian posisi quadrotor bergantung pada pengendalian sikap terbang quadrotor. Hal ini terjadi karena terdapat hubungan antara sudut sikap terbang dengan percepatan translasi quadrotor seperti tertulis pada Persamaan (2).

2.4 Pengendali PID

Penelitian ini menggunakan pengendali PID untuk menstabilkan quadrotor saat melakukan misi *waypoint tracking*. Pengendali ini digunakan karena kesederhanaan dalam hal perancangan dan implementasi. Dalam domain diskrit (kT), pengendali PID memiliki persamaan sebagai berikut:

$$u(kT) = K_p e(kT) + K_i T \sum_{k=1}^n e(kT) + K_d \frac{e(kT) - e(kT-T)}{T} \quad (11)$$

dengan u adalah sinyal kendali, K_p , K_i , K_d adalah konstanta PID, e adalah sinyal kesalahan, dan T adalah waktu *sampling*. Diagram blok pengendali PID pada quadrotor dapat dilihat pada Gambar 4.



Gambar 4. Pengendali PID pada Quadrotor

Pada sistem dinamik quadrotor, terdapat enam buah *state* seperti ditunjukkan pada Persamaan (1). Akan tetapi hanya terdapat empat sinyal kendali yaitu berupa tegangan untuk keempat rotor penggerak sesuai dengan Persamaan (4), Persamaan (8), Persamaan (9), dan Persamaan (10). Dengan demikian diperoleh persamaan sinyal kendali sikap quadrotor sebagai berikut:

$$\left. \begin{aligned} U_z(kT) &= \left[g + Kp_z e_z(kT) + Ki_z T \sum_{k=1}^n e_z(kT) + Kd_z \frac{e_z(kT) - e_z(kT-T)}{T} \right] \frac{m}{C_\phi C_\theta} \\ U_\phi(kT) &= \left[Kp_\phi e_\phi(kT) + Ki_\phi T \sum_{k=1}^n e_\phi(kT) + Kd_\phi \frac{e_\phi(kT) - e_\phi(kT-T)}{T} \right] I_{xx} \\ U_\theta(kT) &= \left[Kp_\theta e_\theta(kT) + Ki_\theta T \sum_{k=1}^n e_\theta(kT) + Kd_\theta \frac{e_\theta(kT) - e_\theta(kT-T)}{T} \right] I_{yy} \\ U_\psi(kT) &= \left[Kp_\psi e_\psi(kT) + Ki_\psi T \sum_{k=1}^n e_\psi(kT) + Kd_\psi \frac{e_\psi(kT) - e_\psi(kT-T)}{T} \right] I_{zz} \end{aligned} \right\} \quad (12)$$

dimana U_z adalah sinyal kendali untuk gaya angkat, U_ϕ , U_θ , U_ψ adalah sinyal kendali untuk sikap *roll*, *pitch*, dan *yaw*. Persamaan sinyal kendali U_z dipengaruhi oleh gaya gravitasi, massa, dan *cosinus* sudut *roll* dan *pitch*. Persamaan sinyal kendali sikap *roll*, *pitch*, dan *yaw* dikalikan dengan momen inersia pada masing-masing sumbu orientasi.

Hubungan antara persamaan sinyal kendali ini dengan kecepatan masing-masing rotor dapat dituliskan sebagai berikut:

$$\left. \begin{aligned} \omega_1^2 &= \frac{U_z}{4k} - \frac{U_\theta}{2kl} - \frac{U_\psi}{4b} \\ \omega_2^2 &= \frac{U_z}{4k} - \frac{U_\phi}{2kl} + \frac{U_\psi}{4b} \\ \omega_3^2 &= \frac{U_z}{4k} + \frac{U_\theta}{2kl} - \frac{U_\psi}{4b} \\ \omega_4^2 &= \frac{U_z}{4k} + \frac{U_\phi}{2kl} + \frac{U_\psi}{4b} \end{aligned} \right\} \quad (13)$$

dengan ω_i adalah kecepatan motor ke- i , k adalah konstanta gaya angkat, l adalah panjang lengan quadrotor, dan b koefisien *drag*.

Pada pengendali posisi quadrotor, terdapat *state* posisi x (longitude) dan y (latitude). Keluaran dari pengendali ini adalah nilai referensi sudut sikap *roll*, *pitch*, dan *yaw* pada quadrotor. Sinyal kendali yang dihasilkan memenuhi persamaan berikut

$$U_x(kT) = \left[Kp_x e_x(kT) + Ki_x T \sum_{k=1}^n e_x(kT) + Kd_x \frac{e_x(kT) - e_x(kT-T)}{T} \right] \frac{m}{C_\psi S_\theta C_\phi + S_\psi S_\phi} \quad (14)$$

$$U_y(kT) = \left[Kp_y e_y(kT) + Ki_y T \sum_{k=1}^n e_y(kT) + Kd_y \frac{e_y(kT) - e_y(kT-T)}{T} \right] \frac{m}{S_\psi S_\theta C_\phi - C_\psi S_\phi} \quad (15)$$

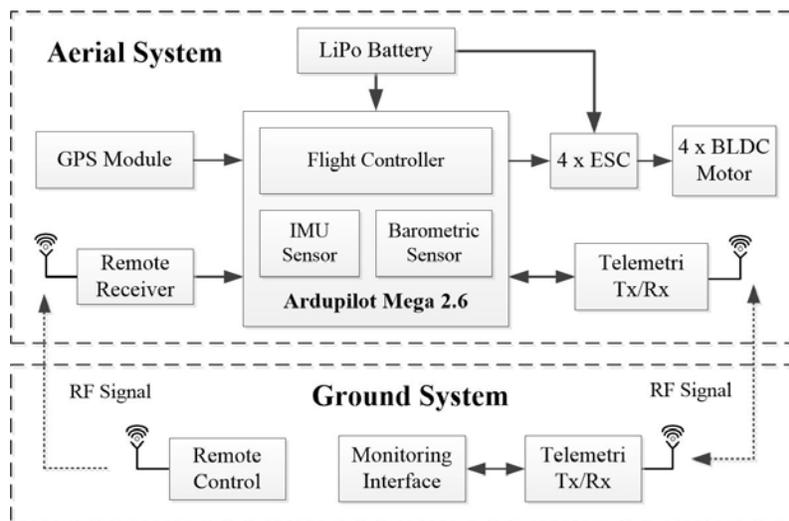
Dari persamaan sinyal kendali yang telah diuraikan, terdapat konstanta PID sebanyak 18 dimana setiap tiga konstanta digunakan untuk pengendalian enam buah *state* pada quadrotor.

2.5 Konfigurasi Perangkat Keras

Sebuah quadrotor dirancang dan diimplementasikan dalam bentuk perangkat keras untuk menguji pengendali PID dengan misi *waypoint tracking*. Spesifikasi dari perangkat keras quadrotor yang dirancang dapat dilihat pada Tabel 1 dengan diagram blok konfigurasi ditampilkan pada Gambar 5.

Tabel 1. Spesifikasi Perangkat Keras Quadrotor

Perangkat keras	Spesifikasi
<i>Flight Controller</i>	Ardupilot Mega 2.6, Gyroscope, Accelerometer, Barometric Pressure, GPS Compatible
Motor BLDC	1000 KV/12A
<i>Electronic Speed Control (ESC)</i>	30A
Kerangka	450 mm x 55 mm, 300 g
Baterai	3 cell @3,7V 2200 mAh



Gambar 5. Konfigurasi Perangkat Keras Quadrotor

Secara umum, terdapat dua sistem konfigurasi perangkat keras yang dirancang, yaitu sistem di udara (*aerial*) dan di darat (*ground*). Pada sistem di udara, quadrotor dirancang menggunakan *flight controller* APM 2.6 yang telah terintegrasi sensor IMU dan Barometer. Sensor IMU yang terdiri dari *Gyroscope* dan *Accelerometer* berfungsi untuk mendeteksi sudut dan kecepatan sudut sikap quadrotor, sedangkan sensor barometer digunakan untuk mendeteksi ketinggian quadrotor. Modul GPS digunakan sebagai sensor posisi *longitude* dan *latitude* quadrotor. Pada bagian *output*, terdapat empat unit *Electronic Speed Controller* (ESC) dan *Brushless DC Motor* (BLDC) yang dilengkapi propeler sebagai sistem penggerak quadrotor. Pada bagian sistem antarmuka di darat, terdapat *remote control* yang dapat dioperasikan oleh pilot melalui komunikasi frekuensi radio. Antarmuka data telemetri Mission Planner digunakan untuk menampilkan data sikap, ketinggian, dan posisi quadrotor saat melakukan misi. Selain itu, antarmuka ini juga dapat digunakan untuk pengaturan parameter pengendalian quadrotor.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Perangkat Keras

Implementasi perangkat keras quadrotor saat di darat dan ketika terbang dapat dilihat pada Gambar 6.



Gambar 6. Implementasi Quadrotor

Pada bagian sisi kiri terdapat quadrotor dengan APM 2.6 sebagai *flight controller*, *remote control* untuk pengendalian manual, dan perangkat antarmuka *Mission Planner*. Pada bagian sisi kanan merupakan quadrotor saat terbang melakukan misi *waypoint tracking*.

3.2 Pengujian Pengendali PID

Pengujian pengendalian *waypoint tracking* dilakukan untuk mengetahui kinerja pengendali PID ketika diimplementasikan pada quadrotor. Adapun konstanta PID yang digunakan dapat dilihat pada Tabel 2. Konstanta ini diperoleh dari nilai *default* pada antarmuka *flight controller* Mission Planner dan telah dilakukan sedikit *tuning* dengan metode *trial and error*.

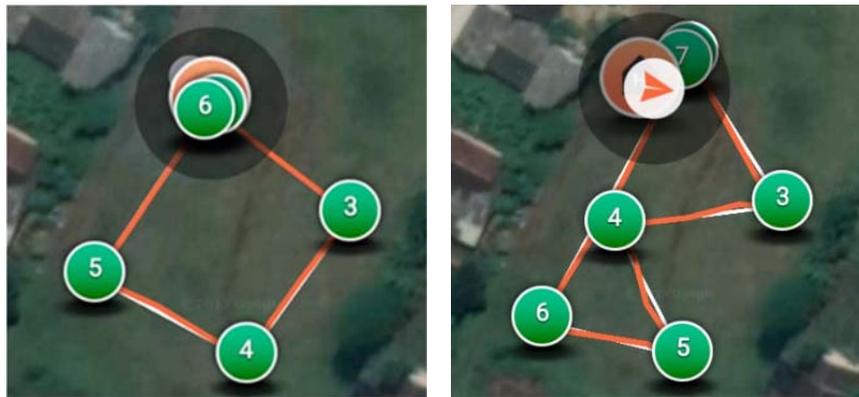
Tabel 2. Konstanta PID

Konstanta	State					
	x	y	z	ϕ	θ	ψ
Kp	1,0	1,0	5,0	0,15	0,15	0,2
Ki	0,0	0,0	0,0	0,1	0,1	0,02
Kd	0,0	0,0	0,0	0,04	0,04	0,0

Dari konstanta yang ditentukan terlihat bahwa pengendalian hanya dilakukan pada empat *state* saja yaitu z , ϕ , θ , dan ψ . Sesuai dengan blok diagram pada Gambar 1, pengendalian

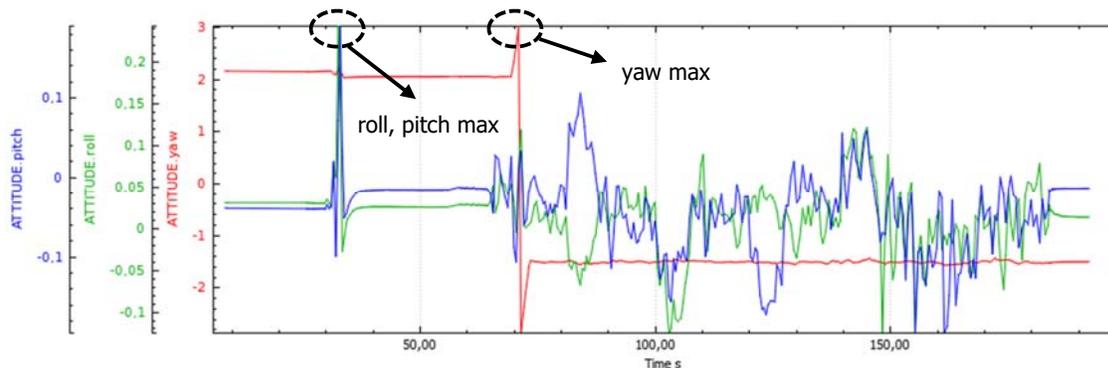
dilakukan pada sikap terbang quadrotor untuk menuju posisi koordinat x dan y. Pada pengendalian posisi x dan y hanya menggunakan konstanta P sebesar 1 yang akan menghasilkan keluaran berupa sudut sikap sebagai referensi pengendalian sikap terbang. Konstanta ini masih dapat diubah dan dioptimasi kembali agar diperoleh kinerja pengendalian yang terbaik.

Pengujian dilakukan melalui dua skenario *waypoint* yaitu persegi dan zig-zag. Kedua skenario *waypoint* ini dibuat menggunakan antarmuka Mission Planner yang kemudian diunggah ke *flight controller* APM 2.6 melalui modul telemetri. Hasil pengujian menggunakan kedua skenario ini disajikan pada Gambar 7. Berdasarkan hasil pengujian terlihat bahwa quadrotor mampu menuju *waypoint* yang telah ditentukan secara *autonomous*. Hasil ini ditunjukkan oleh garis berwarna *orange* yang mampu menjejaki garis berwarna putih. Garis berwarna *orange* merupakan data koordinat quadrotor, sedangkan garis berwarna putih merupakan nilai *setpoint path* perjalanan quadrotor. Titik berwarna hijau merupakan *waypoint* yang ditetapkan dimana untuk skenario persegi terdapat empat titik sedangkan untuk skenario zig-zag terdapat lima titik.

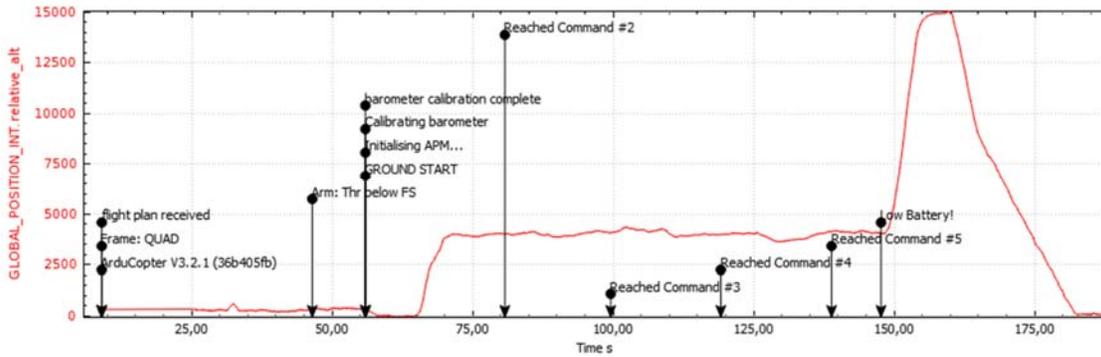


Gambar 7. Pengendalian *Waypoint Tracking* Bentuk Persegi dan Zig-zag

Data sikap dan ketinggian quadrotor saat melakukan misi menggunakan *waypoint* persegi ditunjukkan pada Gambar 8 dan Gambar 9.



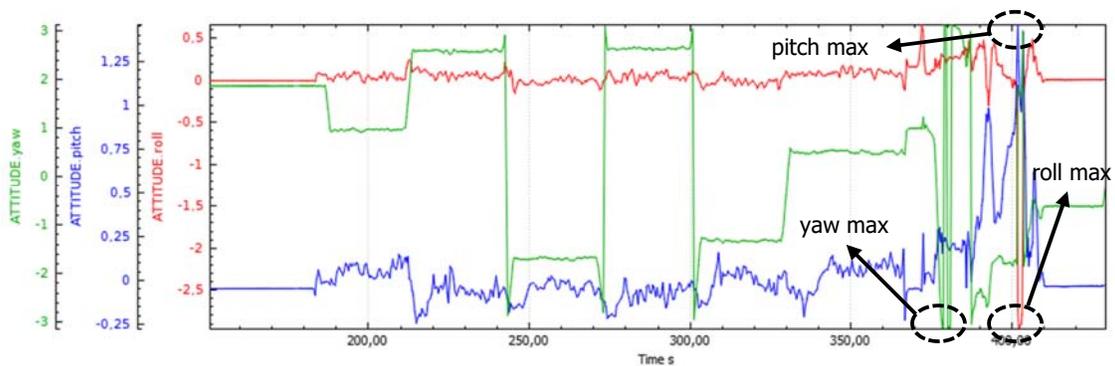
Gambar 8. Respon Sikap Terbang pada Misi *Waypoint Tracking* Bentuk Persegi



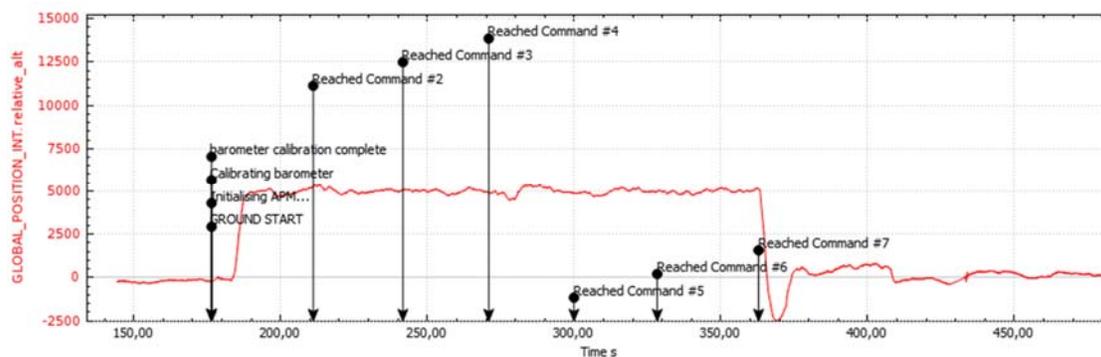
Gambar 9. Ketinggian Quadrotor pada Misi *Waypoint Tracking* Bentuk Persegi

Dari hasil pengujian terlihat bahwa sikap terbang quadrotor relatif stabil dengan simpangan terbesar untuk sudut *roll* sebesar $0,25^\circ$, sudut *pitch* sebesar $0,2^\circ$, dan sudut *yaw* sebesar 3° seperti ditunjukkan pada tanda lingkaran terputus di grafik. Dari hasil tersebut juga terlihat sikap *yaw* yang lebih stabil jika dibandingkan dengan sikap *roll* dan *pitch*. Hal ini terjadi karena pada saat perpindahan posisi koordinat dari satu titik ke titik lain, sikap *yaw* tidak memiliki pengaruh yang besar. Perubahan sikap *roll* dan *pitch* yang menyebabkan quadrotor mampu bergerak dari satu titik ke titik lain. Untuk data ketinggian, quadrotor sebelumnya melakukan proses inialisasi dari mulai perencanaan *waypoint*, kalibrasi ketinggian, dan kalibrasi pengendali. Setelah itu quadrotor melakukan lepas landas menuju *waypoint* 1. Dari grafik terlihat quadrotor mampu mempertahankan ketinggian sekitar 4,2 meter saat bergerak dari *waypoint* 1 hingga ke *waypoint* 4. Kondisi ini diperlihatkan dari histori perintah *reached command #2* sampai dengan perintah *reached command #5*. Setelah itu quadrotor kembali mendarat secara *autonomous*.

Data sikap dan ketinggian quadrotor saat melakukan misi menggunakan *waypoint* zig-zag ditunjukkan pada Gambar 10 dan Gambar 11.



Gambar 10. Respon Sikap Terbang pada Misi *Waypoint Tracking* Bentuk Zig-zag



Gambar 11. Ketinggian Quadrotor pada Misi *Waypoint Tracking* Bentuk Zig-zag

Dari hasil pengujian terlihat bahwa sikap terbang quadrotor juga relatif stabil dengan simpangan terbesar untuk sudut *roll* sebesar -3° , sudut *pitch* sebesar $1,5^\circ$, dan sudut *yaw* sebesar -3° seperti ditunjukkan pada tanda lingkaran di grafik. Dari hasil tersebut juga terlihat sikap *yaw* berubah lebih stabil jika dibandingkan dengan perubahan sudut *roll* dan *pitch* saat berpindah dari satu titik ke titik lain. Sikap *roll* dan *pitch* tidak memiliki simpangan yang terlalu besar saat quadrotor bergerak pada *waypoint*. Untuk data ketinggian, quadrotor sebelumnya melakukan proses inialisasi dan kemudian melakukan lepas landas menuju *waypoint* 1 sampai dengan *waypoint* 5 dengan ketinggian sekitar 5 meter. Kondisi ini diperlihatkan dari histori perintah *reached command #2* sampai dengan perintah *reached command #7*.

3.3 Diskusi

Faktor yang harus dipertimbangkan dalam merancang sebuah quadrotor di antaranya adalah biaya rendah dan kesederhanaan sistem kendali. Perancangan sebuah quadrotor untuk kebutuhan riset dan pengembangan sampai saat ini masih dibutuhkan meskipun telah bermunculan berbagai jenis quadrotor dengan fitur yang canggih. Penelitian ini merupakan keberlanjutan dari studi sebelumnya dimana sebuah quadrotor berbiaya rendah dengan pengendali PID dirancang dengan menghabiskan biaya sekitar dua juta rupiah (Al Tahtawi & Yusuf, 2019). Pada penelitian ini quadrotor tersebut dikembangkan dengan mengganti *flight controller*, menambahkan modul GPS, menambahkan modul telemetri, serta mengganti *remote control*. Pengembangan yang telah dilakukan menghabiskan biaya sekitar dua juta rupiah. Penambahan biaya tersebut diikuti oleh peningkatan kemampuan quadrotor seperti *waypoint tracking* yang telah diuji pada penelitian ini.

4. KESIMPULAN

Sebuah quadrotor berbiaya rendah telah berhasil dirancang dan diimplementasikan dengan misi *waypoint tracking* menggunakan pengendali PID. Quadrotor dirancang agar mampu terbang secara *autonomous* menuju ke beberapa titik koordinat yang telah ditentukan. Pengendali PID digunakan pada *flight controller* APM 2.6 untuk menstabilkan sikap terbang quadrotor dalam menjalankan misinya. Hasil pengujian menunjukkan bahwa quadrotor mampu terbang dan bergerak secara *autonomus* pada dua buah skenario *waypoint* yaitu persegi dan zig-zag. Nilai maksimum sudut sikap untuk kedua skenario tersebut sebesar 3° pada sikap *yaw* di *waypoint* persegi dan -3° pada sikap *roll* dan *yaw* di *waypoint* zig-zag. Ketinggian yang dihasilkan pada skenario *waypoint* persegi adalah 4,2 meter, sedangkan pada *waypoint* zig-zag adalah 5 meter. Dari sisi biaya, quadrotor yang dirancang memiliki biaya yang relatif lebih murah jika dibandingkan dengan quadrotor komersial meskipun

memerlukan pengembangan fitur dan pengendalian yang lebih handal. Penelitian selanjutnya adalah mengembangkan quadrotor untuk misi pemetaan pencemaran udara serta integrasi dengan teknologi *Internet of Things*.

UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada Direktorat Riset dan Pengabdian Masyarakat (DRPM) Kemeristekdikti Republik Indonesia atas dukungan dana pada penelitian ini dengan nomor kontrak 880A/POLSMI/PN/III/2019.

DAFTAR RUJUKAN

- Al Tahtawi, A. R., & Yusuf, M. (2017). Implementasi Sistem Kendali Lepas Landas Quadrotor Menggunakan Pengendali PID. *Prosiding Seminar Nasional Energi dan Teknologi (SINERGI)*, (pp. 160–170).
- Al Tahtawi, A. R., & Yusuf, M. (2019). Low-cost quadrotor hardware design with PID control system as flight controller. *Telkomnika (Telecommunication Computing Electronics and Control)*, 17(4), 1923–1930.
- Bai, Y., Liu, H., Shi, Z., & Zhong, Y. (2012). Robust flight control of quadrotor unmanned air vehicles. *Jiqiren/Robot*, 34(5), 519–524.
- Boudjedir, H. (2012). Adaptive Neural Network for a Quadrotor Unmanned Aerial Vehicle. *International Journal in Foundations of Computer Science & Technology*, 2(4), 1–13.
- Cabecinhas, D., Naldi, R., Marconi, L., Silvestre, C., & Cunha, R. (2012). Robust take-off for a quadrotor vehicle. *IEEE Transactions on Robotics*, 28(3), 734–742.
- Hussein, A., Al-Kaff, A., De La Escalera, A., & Armingol, J. M. (2015). Autonomous indoor navigation of low-cost quadcopters. *10th IEEE Int. Conf. on Service Operations and Logistics, and Informatics, SOLI 2015 - In Conjunction with ICT4ALL 2015*, (pp. 133–138).
- Indrawati, V., Prayitno, A., & Kusuma, T. A. (2015). Waypoint navigation of AR. Drone quadrotor using fuzzy logic controller. *Telkomnika (Telecommunication Computing Electronics and Control)*, 13(3), 930–939.
- Li, J., & Li, Y. (2011). Dynamic analysis and PID control for a quadrotor. *2011 IEEE International Conference on Mechatronics and Automation, ICMA 2011*, (pp. 573–578).
- Luukkonen, T. (2011). Modelling and Control of Quadrotor. *School of Science, Aalto University*.
- Pham, H. X., La, H. M., Feil-Seifer, D., & Nguyen, L. V. (2018). Autonomous UAV Navigation Using Reinforcement Learning, *IEEE International Symposium on Safety, Security, and*

- Rescue Robotics (SSRR)*, (pp. 1-6).
- Qian, G. M., Pebrianti, D., Chun, Y. W., Hao, Y. H., & Bayuaji, L. (2017). Waypoint navigation of quad-rotor MAV. *2017 7th IEEE International Conference on System Engineering and Technology*, (pp. 38–42).
- Reizenstein, A. (2017). Position and Trajectory Control of a Quadcopter Using PID and LQ Controllers, *Master Thesis Linkoping University*.
- Tran, H. K., & Nguyen, T. N. (2018). Flight Motion Controller Design using Genetic Algorithm for a Quadcopter. *Measurement and Control (United Kingdom)*, *51*(3), 59–64.
- Valeria, E. R., Caldera, R. E., Lara, S. C., & Guichard, J. (2013). LQR control for a quadrotor using unit quaternions, *International Conference on Electronics, Communications, and Computing (CONIELECOMP)*, (pp. 172–178).
- Villa, T., Gonzalez, F., Miljevic, B., Ristovski, Z. D., & Morawska, L. (2016). An overview of small unmanned aerial vehicles for air quality measurements: Present applications and future perspectives. *Sensors (Switzerland)*, *16*(7), 12–20.
- Zhang, D., Li, C., & Zhang, Y. (2017). Dual-hand gesture controlled quadcopter robot. *Chinese Control Conference, CCC*, (pp. 6869–6874).