

**SISTEM PENELUSURAN RUTE DAN KONTROL TERPUSAT
UNTUK TUGAS PEMETAAN
MENGUNAKAN *DRONE* JAMAK**

TUGAS AKHIR



Oleh

Shania Argiliana NIM: 13320053

**PROGRAM STUDI TEKNIK FISIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI BANDUNG
2024**

ABSTRAK

SISTEM PENELUSURAN RUTE DAN KONTROL TERPUSAT UNTUK TUGAS PEMETAAN MENGUNAKAN *DRONE* JAMAK

Oleh
Shania Argiliana NIM: 13320053

(Program Studi Teknik Fisika)

Drone jamak dapat digunakan untuk melakukan tugas pemetaan, sehingga diperlukan akurasi yang baik dalam penelusuran rute pemetaan. Selain akurasi, *drone* harus melaksanakan tugas dengan waktu tempuh sesingkat-singkatnya. Untuk mencapai tersebut, *drone* akan dikontrol dengan menggunakan kontrol terpusat yang terhubung pada *access point*. Kontrol tersebut menyebabkan *drone* dianggap sebagai individu yang berbeda, sehingga *drone* akan bersifat independen. Meningkatkan akurasi penelusuran posisi dilakukan dengan mendeteksi beberapa marka fidusial persegi (MFP) dalam proses penentuan posisi. Posisi dari *drone* diasumsikan dengan titik pusat kamera, sehingga perhitungan posisi *drone* lebih akurat. Selain itu, algoritma pemisahan diterapkan juga pada *drone* untuk menghindari tabrakan saat menjalankan tugas pemetaan. Algoritma tersebut akan memisahkan *drone* sesuai dengan jarak aman yang telah ditetapkan, yaitu 200 cm. Pada penelitian ini, jumlah *drone* yang digunakan berjumlah 2 *drone*. Dari hasil penelitian, *drone* melakukan sebuah pengujian penelusuran rute searah sumbu x dan sumbu y. Berdasarkan pengujian tersebut, nilai RMSE yang dihasilkan mengalami penurunan. Saat *drone* bergerak searah sumbu x positif, nilai RMSE yang diperoleh adalah 5,2 cm. Nilai tersebut lebih rendah dari hasil penelitian sebelumnya yang bernilai 19,3 cm. Dalam menjalankan tugas pemetaan, *drone* akan melakukan penelusuran rute dan didapat nilai RMSE untuk *drone* A sebesar 12,3 cm dan *drone* B sebesar 22,5 cm dengan waktu tempuh 51 detik. Dengan demikian, Teknik pendeteksian beberapa marka meningkatkan kemampuan penelusuran rute dan *drone* dapat melakukan tugas pemetaan dengan baik, meskipun *drone* bekerja pada area yang terbatas, yaitu 5m x 4m.

Kata kunci: *drone*, kontrol terpusat, penelusuran rute, marka fidusial persegi

ABSTRACT

POINT-TRACING AND CENTRALIZED CONTROL SYSTEMS FOR MAPPING TASKS USING MULTIPLE DRONES

By

Shania Argiliana NIM: 13320053

(Engineering Physics Study Program)

Multiple drones can be used for mapping tasks, so high accuracy is needed in trajectory-tracing for mapping. In addition to accuracy, drones must complete their tasks in the shortest possible time. To achieve this, the drones will be controlled using a centralized control system connected to an access point. This control system treats each drone as a distinct individual, thus making the drones independent. Enhancing the accuracy of position tracing is done by detecting several square fiducial markers (SFM) in the positioning process. The position of the drone is assumed to be at the center of the camera, leading to more precise position calculations. Furthermore, a separation algorithm is also applied to the drones to avoid collisions while performing mapping tasks. This algorithm will separate the drones according to a predetermined safe distance, which is 200 cm. In this research, the number of drones used amounted to 2 drones. According to the research findings, the drones underwent a trajectory-tracing test along the x-axis and y-axis. Based on this test, the resulting RMSE value decreased. When the drones moved along the positive x-axis, the RMSE value obtained was 5,2 cm, lower than the previous research result of 19,3 cm. While performing mapping tasks, the drones traced the route and obtained RMSE values of 12,3 cm for drone A and 22,5 cm for drone B, with a travel time of 51 seconds. Thus, the technique of detecting multiple markers enhances the trajectory-tracing capability, and the drones can perform mapping tasks well even though they operate in a confined area of 5 m x 4 m.

Keywords: *drone, centralized control, trajectory-tracing, square fiducial marker.*

**SISTEM PENELUSURAN RUTE DAN KONTROL TERPUSAT
UNTUK TUGAS PEMETAAN
MENGUNAKAN *DRONE* JAMAK**

HALAMAN PENGESAHAN

Oleh

Shania Argiliana NIM: 13320053

(Program Studi Teknik Fisika)

Institut Teknologi Bandung

Menyetujui

Tim Pembimbing

Rabu, 26 Juni 2024


Pembimbing 1

Pembimbing 2



(Ir. Estiyanti Ekawati, M.T., Ph.D.)

NIP. 196908052008012020



(Faqihza Mukhlis, S.T., M.T., Ph.D.)

NOPEG. 121110001

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa yang telah memberikan rahmat beserta karunia-Nya, serta diberi-Nya kemudahan dalam menyelesaikan penulisan Laporan Tugas Akhir yang berjudul “Sistem Penelusuran Rute dan Kontrol Terpusat untuk Tugas Pemetaan Menggunakan *Drone* Jamak”. Penulisan Laporan Tugas Akhir ini merupakan salah satu syarat untuk menyelesaikan pendidikan tahap sarjana pada Program Studi Sarjana Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Bandung.

Laporan Tugas Akhir ini dapat terselesaikan dengan baik, tidak lepas dari bimbingan dan dukungan dari banyak pihak, sehingga dalam kesempatan ini penulis dengan segala hormat mengucapkan terima kasih kepada:

1. Ir. Estiyanti Ekawati, M.T., Ph.D., selaku dosen pembimbing pertama yang telah meluangkan waktu dan pikirannya untuk membimbing penulis dalam menyelesaikan penelitian dan penyusunan Laporan Tugas Akhir ini,
2. Faqihza Mukhlis, S.T., M.T., Ph.D selaku dosen pembimbing kedua yang telah meluangkan waktu dan pikirannya untuk membimbing penulis dalam menyelesaikan penelitian dan penyusunan Laporan Tugas Akhir ini,
3. Ir. R. Sugeng Joko Sarwono, MT., Ph.D., selaku dosen wali yang telah membimbing penulis selama menempuh pendidikan di Program Studi Sarjana Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Bandung,
4. Kedua orang tua yang selalu memberikan doa dan dukungan kepada penulis,
5. Seluruh Dosen dan Staf di Program Studi Sarjana Teknik Fisika yang telah membekali banyak ilmu pengetahuan yang bermanfaat selama penulis menjalani studi di Program Studi Sarjana Teknik Fisika,
6. Rekan-rekan di laboratorium CITA atas kebersamaan selama penelitian ini dilaksanakan,
7. Teman-teman S1 Teknik Fisika Angkatan 2020 yang memberikan dukungan dan semangat kepada penulis, dan

8. Semua pihak yang telah membantu hingga terselesaikannya Laporan Tugas Akhir ini yang tidak dapat penulis sebutkan satu per satu.

Semoga Allah SWT senantiasa memberikan limpahan rahmat dan karunia-Nya kepada semua pihak yang terlibat dalam proses penulisan Laporan Tugas Akhir ini.

Akhir kata penulis memohon maaf apabila terdapat banyak kekurangan dalam penulisan Laporan Tugas Akhir ini. Oleh karena itu, saran dan kritik yang membangun sangat diharapkan guna penyempurnaan dan pengembangan Tugas Akhir ini ke arah yang lebih baik. Semoga segala yang tertuang dalam Laporan Tugas akhir ini memberikan manfaat bagi kita semua baik sekarang maupun dimasa yang akan datang.

Bandung, Mei 2024

Penulis

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
HALAMAN PENGESAHAN.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xi
DAFTAR SINGKATAN DAN LAMBANG	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Permasalahan.....	2
1.3 Tujuan Penelitian.....	3
1.4 Ruang Lingkup Permasalahan.....	3
1.4.1 Batasan	3
1.4.2 Asumsi.....	4
1.5 Alur Penelitian.....	4
1.6 Sistematika Penulisan.....	5
BAB II TEORI	7
2.1 Model Sistem <i>Drone</i>	7
2.1.1 Persamaan Dinamika <i>Drone</i>	7
2.1.2 Persamaan Dinamika <i>Drone</i> Menggunakan FOPTD	10
2.2 Pengontrol PD	11
2.3 Marka Fidusial Persegi.....	12
2.4 <i>Moving Average</i>	13
2.5 Jaringan Nirkabel	13
2.5.1 TCP/IP	14
2.5.2 UDP.....	14
2.6 Algoritma Pemisahan	15
2.7 Jenis-Jenis Jalur Pemetaan pada <i>Drone</i>	15
BAB III METODE.....	17
3.1 Arsitektur Komunikasi dan Pengaturan Pengujian	17

3.1.1	Komunikasi <i>Done</i>	17
3.1.2	Pengaturan Pengujian.....	18
3.2	Karpet MAT.....	19
3.3	Sistem Pemosisian.....	19
3.3.1	Pendeteksian Marka ArUco.....	20
3.3.2	Konversi Marka ArUco Menjadi Titik Koordinat.....	20
3.3.3	Penentuan Posisi <i>Drone</i>	21
3.4	Perancangan Algoritma Pemisahan pada <i>Drone</i> Jamak.....	25
3.5	Perancangan Penelusuran Titik pada Tugas Pemetaan.....	27
3.6	Pengujian Penelusuran <i>Drone</i>	27
BAB IV	HASIL DAN ANALISIS.....	31
4.1	Sistem Pemosisian.....	31
4.2	Identifikasi Karakteristik <i>Drone</i>	33
4.2.1	Hasil Identifikasi Karakteristik <i>Drone</i>	33
4.2.2	Hasil Penalaan Parameter Pengontrol PD.....	35
4.3	Pengujian Individu.....	36
4.3.1	Penelusuran Rute Searah Sumbu X.....	36
4.3.2	Penelusuran Rute Searah Sumbu Y.....	40
4.4	Pengujian Berkelompok.....	41
4.4.1	Pengujian Algoritma Pemisahan untuk Menghindari Tabrakan	41
4.4.2	Pengujian untuk Tugas Pemetaan.....	43
BAB V	PENUTUP.....	48
5.1	Kesimpulan.....	48
5.2	Saran.....	48
DAFTAR PUSTAKA	49
LAMPIRAN A.	SPESIFIKASI <i>DRONE</i> PARROT ARDRONE 2.0.....	51
LAMPIRAN B.	SPESIFIKASI <i>ROUTER</i> WI-FI.....	52
LAMPIRAN C.	HASIL IDENTIFIKASI KARAKTERISTIK KECEPATAN <i>DRONE</i>.....	53
C.1	Grafik Karakteristik Kecepatan <i>Drone</i> A.....	53
C.2	Grafik Karakteristik Kecepatan <i>Drone</i> B.....	55

C.3 Grafik Pengontrol PD untuk <i>Drone A</i>	57
C.4 Grafik Pengontrol PD untuk <i>Drone A</i>	59
LAMPIRAN D. PENELUSURAN POSISI <i>DRONE</i>	60
D.1 Penelusuran Posisi <i>Drone A</i>	60
D.2 Penelusuran Posisi <i>Drone B</i>	63

DAFTAR GAMBAR

Gambar 1.1 Diagram tulang ikan rumusan permasalahan.	3
Gambar 1.2. Diagram alir penelitian.	5
Gambar 2.1 Skema gaya yang ada pada <i>drone</i> dalam kerangka sistem <i>drone</i>	7
Gambar 2.2. Diagram blok pengontrol PD pada <i>drone</i>	12
Gambar 2.3. Ilustrasi marka ArUco.	12
Gambar 2.4. Proses pengolahan citra marka (a) penangkapan marka, (c) Pengembangan citra, (c) pembagian sel citra, dan (d) informasi bit hitam putih [6].	12
Gambar 2.5. Ilustrasi algoritma pemisahan dua individu.	15
Gambar 2.6. Ilustrasi jalur pada tugas pemetaan dengan pola (a) berliku-liku, (b) pilin, dan (c) S.	16
Gambar 3.1. Arsitektur komunikasi.	17
Gambar 3.2. Pengaturan pengujian (a) sudut hadap <i>drone</i> pada karpet dan (b) ketinggian <i>drone</i> saat terbang.	18
Gambar 3.3. Desain karpet MAT.	19
Gambar 3.4. Diagram alir pendeteksian MFP ArUco.	20
Gambar 3.5. Diagram alir konversi kode unik MFP menjadi koordinat pada karpet MAT.	20
Gambar 3.6. Ilustrasi jumlah baris dan kolom dari marka ArUco.	21
Gambar 3.7. Hasil deteksi MFP pada ketinggian 80 cm.	22
Gambar 3.8. Hasil pendeteksian empat marka MFP pada <i>drone</i>	22
Gambar 3.9. Diagram alir pendeteksian beberapa MFP saat <i>drone</i> bergerak.	23
Gambar 3.10. Hasil deteksi dua marka MFP saat <i>drone</i> bergerak searah sumbu x ⁻	23
Gambar 3.11. Diagram alir perancangan algoritma pemisahan.	26
Gambar 3.12. Diagram alir perancangan algoritma kohesi.	26
Gambar 3.13. Diagram alir perancangan penelusuran titik pada tugas pemetaan.	27
Gambar 3.14. Jalur pengujian individu (a) lurus terhadap sumbu x dan (b) lurus terhadap sumbu y.	28
Gambar 3.15. Jalur pengujian algoritma pemisahan.	29
Gambar 3.16. Jalur pengujian secara berkelompok untuk tugas pemetaan.	30
Gambar 4.1. Marka MFP yang terdeteksi oleh drone (a) saat diam dan (b) saat <i>drone</i> bergerak dengan kecepatan 50 cm/s.	31
Gambar 4.2. Ilustrasi deteksi beberapa MFP (a) MFP yang terdeteksi dan menunjukan posisi riil <i>drone</i> dan (b) posisi <i>drone</i> dari hasil pengolahan pada UI.	32
Gambar 4.3. Hasil identifikasi pergerakan ke kiri dari <i>drone</i> A.	33
Gambar 4.4. Respons sistem input step untuk <i>drone</i> A arah maju	35
Gambar 4.5. Grafik respons sistem <i>drone</i> A arah maju dengan pengontrol PD.	36
Gambar 4.6. Ilustrasi pengujian individu.	36

Gambar 4.7. Penelusuran posisi dengan arah maju pada sumbu x (a) rekaman posisi <i>drone</i> saat bergerak maju, (b) profil posisi pada sumbu x dan (c) profil posisi sumbu y.....	38
Gambar 4.8. Penelusuran posisi pada penelitian Giga dengan arah maju pada sumbu x (a) posisi pada sumbu x dan (b) posisi sumbu y.	40
Gambar 4.9. Ilustrasi pengujian berkelompok.	41
Gambar 4.10. Rekaman posisi <i>drone</i> pada pengujian algoritma pemisahan (a) pada $t = 3$ detik, (b) pada $t = 5$ detik, dan (c) pada $t = 11$ detik.....	43
Gambar 4.11. Penelusuran posisi <i>drone</i> A dan <i>drone</i> B selama 20 detik.....	44
Gambar 4.12. Hasil pengujian tugas pemetaan (a) posisi <i>drone</i> pada koordinat <i>cartesius</i>	46

DAFTAR TABEL

Tabel 4.1. Hasil identifikasi kecepatan <i>drone</i> A dan <i>drone</i> B.	34
Tabel 4.2. Parameter pengontrol PD untuk <i>drone</i> A dan <i>drone</i> B	35
Tabel 4.3. Perbandingan nilai RMSE pada penelitian 2022 dan penelitian 2024..	39
Tabel 4.4. Waktu tempuh pengujian penelusuran posisi tiap <i>drone</i> (detik).....	41
Tabel 4.5. Data hasil tugas pemetaan.....	47

DAFTAR SINGKATAN DAN LAMBANG

SINGKATAN	Nama	Pemakaian pertama kali pada halaman
MFP	Marka Fidusial Persegi	2
CITA	<i>Center for Instrumentation Technology and Automation</i>	4
FOPTD	<i>First Order Process with Time Delay</i>	9
SMA	<i>Simple Moving Average</i>	12
TCP/IP	<i>Transmission Control Protocol/Internet Protocol</i>	12
WiFi	<i>Wireless Fidelity</i>	12
UDP	<i>User Datagram Protocol</i>	13

LAMBANG

ϕ	Sudut guling	6
θ	Sudut angguk	6
ψ	Sudut belok	6
m	Massa <i>drone</i>	6
I	Matriks identitas	6
V	Vektor linear	6
F	Gaya resultan	6
τ	Momen resultan	6
Ω	Kecepatan sudut	6
R	Matriks inersia	7
J	Momen inersia dari <i>drone</i>	7
\bar{u}	Gaya mekanik yang memberikan gerak rotasi pada <i>drone</i>	7
k_i	Faktor yang menghubungkan kecepatan sudut dengan motor	8
K	Penguatan	9
τ	Konstanta waktu	9
t_d	Waktu tunda	9
K_p	Penguatan proporsional	10
$e(t)$	Error	10
$u(t)$	Masukan	10
T_d	Waktu derivatif	10
π	Rangkaian aksi setiap agen	15
\bar{x}	Nilai rata-rata koordinat x	20
\bar{y}	Nilai rata-rata koordinat y	20
r_{pixel}	Radius dua titik dalam satuan piksel	22
α_x	Faktor konversi satuan piksel ke cm untuk koordinat x	22

α_y	Faktor konversi satuan piksel ke cm untuk koordinat y	22
(x_1, y_1)	Koordinat dari pendeteksian beberapa MFP	22
(x_2, y_2)	Koordinat titik pusat kamera <i>drone</i> pada bingkai kamera	22
x^+	Pergerakan <i>drone</i> arah sumbu x positif	23
x^-	Pergerakan <i>drone</i> arah sumbu x negatif	23
y^+	Pergerakan <i>drone</i> arah sumbu y positif	23
y^-	Pergerakan <i>drone</i> arah sumbu y negatif	23

BAB II

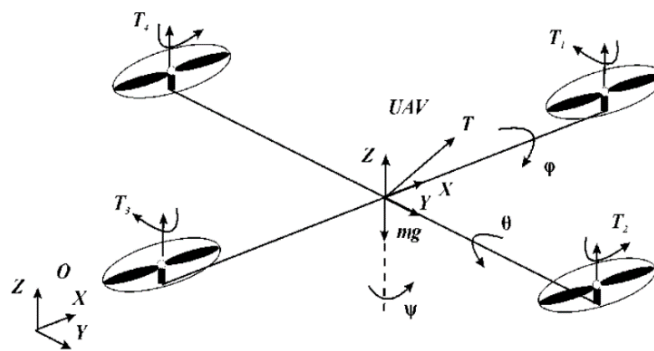
TEORI

Teori yang akan digunakan pada penelitian adalah persamaan dinamika *drone* yang merepresentasikan model sistem *drone*, persamaan fungsi transfer *first order plus time delay* (FOPTD) sebagai pendekatan model sistem *drone*, komunikasi yang digunakan pada *drone*, pengontrol PD pada *drone*, pendeteksian marka fidusial persegi (MFP) untuk penentuan posisi *drone*, dan algoritma pemisahan untuk menghindari tabrakan.

2.1 Model Sistem *Drone*

2.1.1 Persamaan Dinamika *Drone*

Drone merupakan kendaraan terbang nirawak yang dikembangkan dari konsep robot terbang. *Drone* memiliki empat motor dengan dua pasang motor yang bergerak berlawanan arah. Selain itu, *drone* dapat dikontrol secara manual dengan pengontrol jarak jauh menggunakan kontrol otomatis [8]. Dinamika *drone* dapat digambarkan pada koordinat tetap yang terhubung dengan bumi (kerangka O) dan koordinat bergerak yang bertepatan dengan pusat gravitasi *drone* (kerangka UAV). Pada Gambar 2.1, variabel ϕ, θ, ψ merupakan sudut guling, angguk, dan belok yang digambarkan pada koordinat bumi (kerangka O) [9].



Gambar 2.1 Skema gaya yang ada pada *drone* dalam kerangka sistem *drone* [9].

Persamaan dinamika dari *drone* dalam ruang 3D dapat dituliskan sebagai berikut.

$$\begin{bmatrix} mI & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\Omega} \end{bmatrix} + \begin{bmatrix} \Omega \times mV \\ \Omega \times J\Omega \end{bmatrix} = \begin{bmatrix} F \\ \tau \end{bmatrix} \quad (2.1)$$

Dengan $V = (V_x, V_y, V_z)^T = (\dot{x}, \dot{y}, \dot{z})^T$ merupakan vektor linier, $\Omega = (\Omega_\phi, \Omega_\theta, \Omega_\psi)^T = (\dot{\phi}, \dot{\theta}, \dot{\psi})^T$ merupakan kecepatan sudut, m merupakan massa *drone*, $J \in \mathbb{R}^{3 \times 3}$ merupakan matriks inersia objek yang dikontrol, I merupakan matriks identitas, F merupakan gaya resultan, τ merupakan momen resultan yang bekerja pada pusat massa *drone*. Dengan mengabaikan gaya eksternal, gaya yang dihasilkan untuk menentukan pergerakan *drone* di bidang vertikal dapat dituliskan sebagai berikut.

$$F^T = [0 \quad 0 \quad u] \quad (2.2)$$

Dalam sistem koordinat Euler, orientasi *drone* relatif terhadap sistem koordinat inersial XYZ sehingga dapat direpresentasikan oleh matriks rotasi R .

$$F_V = R(\psi, \theta, \phi)F \quad (2.3)$$

Dengan matriks rotasi R dituliskan seperti berikut.

$$R(\psi, \theta, \phi) = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\theta & C_\psi S_\theta C_\phi + S_\psi S_\theta \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\theta & S_\psi S_\theta C_\phi - C_\psi S_\theta \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \quad (2.4)$$

Pada matriks di atas, $C_\alpha = \cos \alpha$ dan $S_\alpha = \sin \alpha$. Apabila Persamaan (2.3) – (2.4) dilakukan kombinasi, maka akan didapatkan hasil persamaan berikut.

$$m\ddot{x} = u_1(C_\psi S_\theta S_\phi + S_\psi C_\theta) \quad (2.5)$$

$$m\ddot{y} = u_2(S_\psi S_\theta S_\phi + C_\psi C_\theta) \quad (2.6)$$

$$m\ddot{z} = u_3(C_\theta C_\phi) - mg \quad (2.7)$$

u_1, u_2, u_3, u_4 merupakan gaya mekanik yang memberikan gerakan rotasi pada *drone* yang direpresentasikan sebagai berikut.

$$u_1 = k_1 \Omega_1^2 + k_2 \Omega_2^2 + k_3 \Omega_3^2 + k_4 \Omega_4^2 \quad (2.8)$$

$$u_2 = k_1 \Omega_1^2 + k_2 \Omega_2^2 - k_3 \Omega_3^2 - k_4 \Omega_4^2 \quad (2.9)$$

$$u_3 = k_4 \Omega_4^2 - k_2 \Omega_2^2 \quad (2.10)$$

$$u_4 = k_1 \Omega_1^2 - k_3 \Omega_3^2 \quad (2.11)$$

k_i pada Persamaan (2.8) – (2.11) merupakan faktor yang menghubungkan momen dengan kecepatan sudut ω_i dari motor ke- i . Kemudian, misalkan J_1, J_2, J_3 adalah momen inersia dari *drone* di sumbu X, Y, dan Z yang melewati pusat massa, $\dot{\phi}, \dot{\theta}, \dot{\psi}$ merupakan kecepatan sudut yang berkaitan dengan sumbu X, Y, dan Z, serta gaya eksternal dari *drone* dapat diabaikan, maka persamaan diferensial kecepatan sudut dapat direpresentasikan dalam bentuk berikut.

$$J_1 \ddot{\phi}(t) = (J_2 - J_3) \dot{\theta}(t) \dot{\psi}(t) \quad (2.12)$$

$$J_2 \ddot{\theta}(t) = (J_3 - J_1) \dot{\psi}(t) \dot{\phi}(t) \quad (2.13)$$

$$J_3 \ddot{\psi}(t) = (J_1 - J_2) \dot{\phi}(t) \dot{\theta}(t) \quad (2.14)$$

Persamaan (2.12) – (2.14) di atas merupakan persamaan dinamika Euler. Ketika aksi dari motor $M_1 - M_4$ dipertimbangkan, maka Persamaan (2.12) – (2.14) dapat direpresentasikan sebagai berikut.

$$J_1 \ddot{\phi}(t) = (J_2 - J_3) \dot{\theta}(t) \dot{\psi}(t) - J_r \dot{\theta}(t) \Omega + d\bar{u}_3(t) \quad (2.15)$$

$$J_2 \ddot{\theta}(t) = (J_3 - J_1) \dot{\psi}(t) \dot{\phi}(t) - J_r \dot{\psi}(t) \Omega + d\bar{u}_4(t) \quad (2.16)$$

$$J_3 \ddot{\psi}(t) = (J_1 - J_2) \dot{\phi}(t) \dot{\theta}(t) + \bar{u}_2(t) \quad (2.17)$$

Pada persamaan di atas, d merupakan jarak antara pusat massa *drone* dan motor \mathfrak{S}_i , sedangkan J_r merupakan inersia motor. Momen J_1 dan J_2 pada ketinggian tetap di sumbu Z dapat dinyatakan sebagai berikut.

$$J_1 = J_2 = J \quad (2.18)$$

Dengan demikian, dinamika rotasi dari *drone* dari Persamaan (2.15) – (2.17) dapat direpresentasikan sebagai berikut.

$$\ddot{\phi}(t) = \frac{1}{J}[(J - J_3)\dot{\theta}(t)\dot{\psi}(t) - J_r\dot{\theta}(t)\Omega + d\bar{u}_3(t)] \quad (2.19)$$

$$\ddot{\theta}(t) = \frac{1}{J}[(J - J_1)\dot{\psi}(t)\dot{\phi}(t) - J_r\dot{\psi}(t)\Omega + d\bar{u}_4(t)] \quad (2.20)$$

$$\ddot{\psi}(t) = \frac{1}{J}[\bar{u}_2(t)] \quad (2.21)$$

Persamaan (2.14) – (2.17) dan Persamaan (2.19) – (2.21) merupakan persamaan orde tinggi yang menghubungkan input u_1, u_2, u_3, u_4 ke sudut guling, angguk, belok dan akhirnya ke XYZ.

2.1.2 Persamaan Dinamika *Drone* Menggunakan FOPTD

Karena inersia *drone* relatif besar, maka respons dinamika kecepatan terhadap input tegangan aktuator *drone* cenderung tidak beresilasi, sehingga menyerupai profil *first order plus time delay* (FOPTD). FOPTD merupakan model pendekatan yang terdiri dari komponen dasar orde pertama dan waktu tunda [10 – 11]. Dalam domain waktu, pendekatan FOPTD dapat dirumuskan secara umum sebagai berikut.

$$\frac{1}{\tau} \frac{d}{dt} y(t) + y(t) = Ku(t - t_d) \quad (2.25)$$

Dengan demikian, persamaan dinamika dari *drone* dapat dinyatakan sebagai berikut dalam domain waktu menggunakan FOPTD.

$$\tau \left(\frac{d}{dt} v_x(t) \right) + v_x(t) = Ku(t - t_d) \quad (2.26)$$

$$\tau \left(\frac{d}{dt} v_y(t) \right) + v_y(t) = Ku(t - t_d) \quad (2.27)$$

K merupakan penguatan, τ merupakan konstanta waktu, dan t_d merupakan waktu tunda. Kemudian, Persamaan (2.26) – (2.27) ditransformasi ke dalam domain laplace dan direpresentasikan sebagai berikut.

$$G(s) = \frac{K}{\tau s + 1} e^{-t_d s} \quad (2.28)$$

Persamaan (2.28) tersebut menghubungkan respons sistem dalam bentuk fungsi transfer. Untuk keperluan perancangan kontrol, Persamaan (2.28) perlu diubah dengan menggunakan pendekatan Padé orde pertama, sehingga fungsi FOPTD

Persamaan (2.28) dapat disimulasikan pada sistem. Berikut merupakan penurunan fungsi FOPTD dengan menggunakan pendekatan Padé.

$$e^{-T_d s} \approx \frac{1 - \frac{T_d}{2}s}{1 + \frac{T_d}{2}s} \quad (2.29)$$

Dengan mensubstitusi Persamaan (2.29) ke dalam Persamaan (2.28), bentuk fungsi FOPTD dapat diubah menjadi Persamaan (2.30) dalam bentuk pendekatan Padé, seperti berikut.

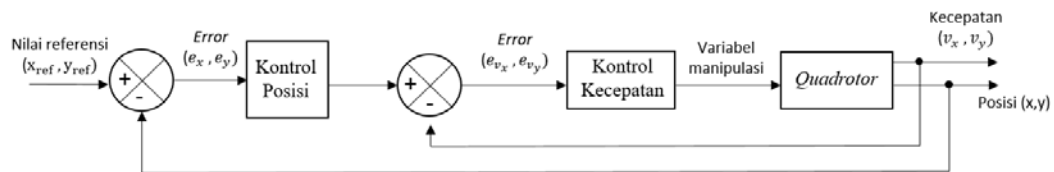
$$G(s) = \frac{K}{\tau s + 1} \times \frac{1 - \frac{T_d}{2}s}{1 + \frac{T_d}{2}s} \quad (2.30)$$

2.2 Pengontrol PD

Pengontrol PID merupakan jenis pengontrol klasik yang digunakan dalam sistem kontrol otomatis untuk mengatur suatu sistem. Salah satu jenis pengontrol klasik tersebut adalah pengontrol PD [11 – 13]. Pengontrol proporsional derivatif (PD) merupakan pengontrol yang terdiri dari dua parameter, yaitu penguatan proporsional K_p dan waktu derivatif T_d . Pengontrol derivatif digunakan untuk mempercepat respons awal suatu sistem, tetapi tidak memperkecil kesalahan pada keadaan tunaknya. Dengan demikian, pengontrol derivatif digunakan dengan pengontrol lainnya. Pengontrol PD dapat dituliskan pada persamaan (2.31).

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (2.31)$$

Gambar 2.2 merupakan diagram blok pengontrol PD yang akan digunakan pada sistem *drone*. Parameter pengontrol PD dirancang dengan menggunakan metode *root locus*. Selain itu, pengontrol PD dipilih untuk menghilangkan *undershoot* yang dihasilkan dari Persamaan (2.31). Gambar 2.2 menunjukkan bahwa *drone* telah memiliki komponen integral, sehingga hasil dari pengontrol PD akan terintegralkan dan menjadi *set poin* untuk mengatur kecepatan *drone*.



Gambar 2.2. Diagram blok pengontrol PD pada *drone*.

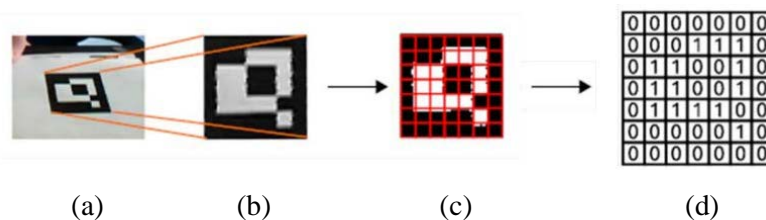
2.3 Marka Fidusial Persegi

Marka fidusial persegi (MFP) merupakan marka berbentuk persegi yang memiliki pola tertentu di setiap markanya dan mengandung kode unik tertentu, seperti pada Gambar 2.3. Dengan demikian, marka ini digunakan untuk mengidentifikasi dan memposisikan objek secara akurat [7]. Selain itu, MFP memiliki empat titik korespondensi yang dapat berfungsi sebagai estimasi pose kamera. Pada penelitian ini, pustaka pendeteksi yang digunakan adalah pustaka ArUco[14].



Gambar 2.3. Ilustrasi marka ArUco [15].

Deteksi marka MFP dilakukan dengan dua proses, yaitu deteksi marka dan identifikasi informasi marka. Proses deteksi marka MPF dapat diilustrasikan seperti pada Gambar 2.4.



Gambar 2.4. Proses pengolahan citra marka (a) penangkapan marka, (c) Pengambilan citra, (c) pembagian sel citra, dan (d) informasi bit hitam putih [6].

Pada proses Gambar 2.4 (a) – (b), citra yang ditangkap oleh kamera akan dianalisis untuk menemukan bentuk-bentuk persegi yang menjadi kandidat marka [14 – 15]. Hal pertama yang dilakukan adalah proses ambang pada citra masukan menggunakan metode ambang lokal untuk memperhitungkan variabel iluminasi. Kemudian, kontur diekstraksi dari hasil citra sebelumnya dan dilakukan penyaringan. Citra yang dihasilkan dari proses penyaringan tersebut akan dilakukan pendekatan poligon untuk mendapatkan bentuk persegi, sehingga citra selain

bentuk segi empat poligon akan dibuang. Selain itu, penyaringan ekstra juga diterapkan untuk menghapus kontur yang terlalu kecil atau terlalu besar dan menghapus kontur yang terlalu dekat satu sama lain.

Setelah proses deteksi marka, proses identifikasi informasi marka akan dilakukan dengan menganalisis kode internal pada setiap marka [14 – 15]. Proses identifikasi diawali dengan mengekstraksi bit-bit yang ada pada setiap marka. Untuk mendapatkan tanda dalam bentuk kanonik, transformasi perspektif diterapkan terlebih dahulu pada proses tersebut. Citra kanonik yang dihasilkan akan dilakukan pengembangan menggunakan metode Otsu untuk memisahkan bit putih dan hitam, seperti Gambar 2.4 (b). Selanjutnya, citra tersebut akan dibagi menjadi sel-sel berbeda sesuai dengan ukuran tanda dan ukuran tepi, seperti Gambar 2.24 (c). Kemudian, jumlah piksel hitam atau putih di setiap sel dihitung untuk menentukan bit putih atau hitam. Pada tahap terakhir, Gambar 2.4 (d), bit-bit tersebut dianalisis sesuai dengan kamus yang tersedia pada marka ArUco.

2.4 *Moving Average*

Moving average merupakan salah satu konsep untuk melakukan analisis data dengan cara merata-ratakan serangkaian data yang terus berubah seiring waktu. Salah satu jenis *moving average* sederhana adalah *Simple Moving Average* (SMA) [16]. Persamaan untuk menghitung SMA dapat dituliskan seperti Persamaan (2.33).

$$SMA = \frac{x_1 + x_2 + \dots + x_n}{n} \quad (2.33)$$

dengan x_1, x_2, \dots, x_n merupakan nilai data setiap periode dan n adalah jumlah periode waktu yang digunakan untuk menghitung SMA. Pada penelitian ini, SMA digunakan untuk mengolah marka MFP menjadi titik koordinat *drone*. Pada penelitian ini, nilai n yang digunakan adalah empat.

2.5 Jaringan Nirkabel

Jaringan nirkabel didefinisikan sebagai jaringan transmisi yang tidak menggunakan kabel sebagai media dalam menghubungkan sumber data dan penerima data [17]. Jaringan nirkabel memungkinkan banyak pengguna terhubung secara bersamaan ke dalam jaringan. Jaringan nirkabel memiliki *Access Point* yang bekerja dengan

menggunakan *transceiver radio* untuk menciptakan koneksi [18]. Sinyal radio yang ditransmisikan akan diidentifikasi oleh perangkat klien. Setelah saluran komunikasi dikonfirmasi, akses ke jaringan menjadi tersedia. *Access Point* nirkabel mengikuti standar protokol IEEE 802.11, umumnya standar ini diterapkan dalam WiFi atau *Wireless Fidelity*.

2.5.1 TCP/IP

TCP/IP merupakan singkatan dari *Transmission Control Protocol/Internet Protocol* [19]. TCP/IP adalah kumpulan protokol komunikasi yang dapat digunakan untuk menghubungkan perangkat-perangkat dalam jaringan komputer. Dengan demikian, satu komputer dengan TCP/IP dapat berkomunikasi dengan komputer lain pada satu lingkup jaringan nirkabel. TCP/IP digunakan untuk mengirimkan data berupa angka. TCP/IP terdiri dari dua protokol utama, yaitu TCP dan IP.

Protokol TCP (*Transmission Control Protocol*) bertanggung jawab untuk memastikan pengiriman data yang andal antar perangkat dalam jaringan [20]. TCP akan membagi data menjadi paket-paket kecil, memastikan setiap paket yang datang dengan benar, dan menyusun kembali paket-paket tersebut. Selain itu, TCP berfungsi untuk memastikan bahwa data yang dikirim menjadi paket-paket kecil tersebut tidak hilang, rusak, atau urutan paket tertukar.

Protokol IP (*Internet Protocol*) merupakan protokol yang berfungsi untuk menangani pengiriman paket data antar perangkat dalam jaringan [20]. IP akan mengatur alamat unik (*IP address*) dari setiap perangkat yang terhubung ke jaringan dan memastikan bahwa paket data dikirimkan ke tujuan yang tepat berdasarkan alamat unit tersebut.

2.5.2 UDP

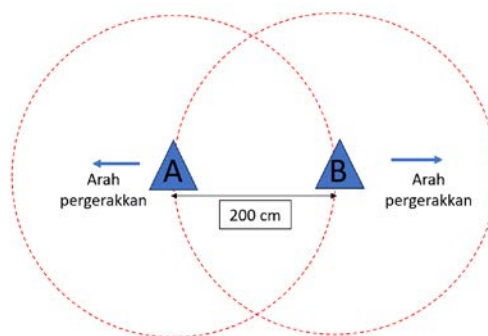
UDP atau *User Datagram Protocol* adalah protokol transmisi data dalam jaringan yang berbeda dengan TCP/IP [21]. UDP bersifat sederhana karena data yang dikirim berupa datagram dan tidak memiliki informasi verifikasi, sehingga memiliki kemungkinan data akan rusak saat pengiriman. Akan tetapi, pengiriman data yang dilakukan pada protokol UDP lebih cepat dibandingkan protokol TCP/IP.

Data yang dikirimkan menggunakan protokol UDP kepada komputer dapat berupa citra.

2.6 Algoritma Pemisahan

Algoritma pemisahan merupakan salah satu aturan dasar yang digunakan dalam simulasi perilaku kelompok hewan, seperti kawanan burung yang disebut “*boids*” [22]. Algoritma pemisahan digunakan untuk menghindari tabrakan antar individu dalam satu kelompok. Dengan demikian, algoritma pemisahan dapat memastikan bahwa setiap individu menjaga jarak aman dengan individu lain saat menjalankan tugas dalam satu kelompok. Hal tersebut yang memungkinkan algoritma pemisahan dapat menghindari kepadatan yang berlebih dan tabrakan yang tidak diinginkan.

Perancangan algoritma pemisahan dilakukan dengan beberapa langkah, langkah pertama adalah melakukan identifikasi tetangga terdekat [23]. Pada langkah tersebut, individu akan mencari individu lain yang berada dalam radius. Langkah kedua yang dilakukan adalah menghitung target posisi baru dan arah pergerakan. Pada langkah ini, individu akan menghitung posisi target baru untuk menghindari individu lain. Apabila pada posisi target baru tidak ada individu lain, maka individu akan bergerak ke arah tersebut. Dengan demikian, setiap individu akan memeriksa individu lain dalam radius tertentu yang disebut sebagai zona pemisahan. Gambar 2.5 merupakan ilustrasi terkait algoritma pemisahan



Gambar 2.5. Ilustrasi algoritma pemisahan dua individu.

2.7 Jenis-Jenis Jalur Pemetaan pada *Drone*

Dalam menjalankan tugas pemetaan, *drone* jamak dapat bergerak dengan mengikuti beberapa jalur pemetaan [3]. Berikut ini merupakan beberapa jenis jalur pemetaan yang dapat diterapkan pada *drone* jamak.

a. Jalur Berliku-liku

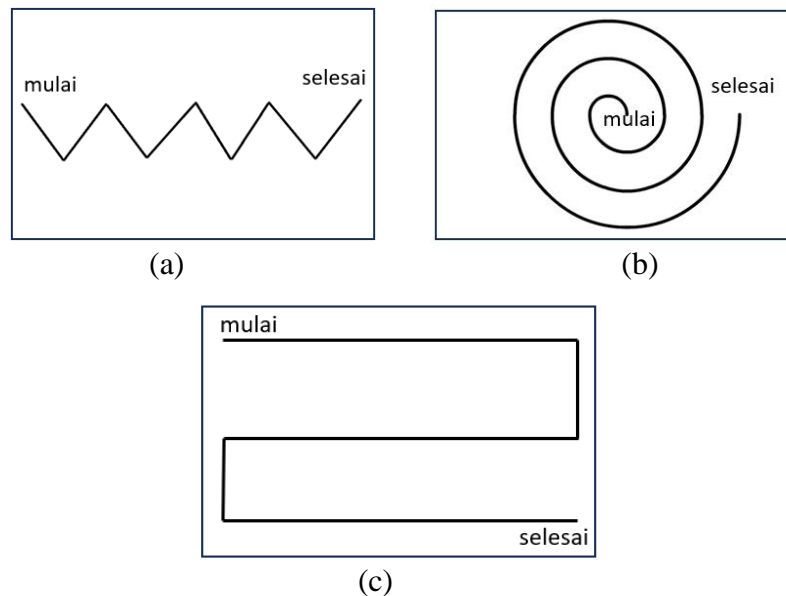
Jalur Berliku-liku merupakan jalur yang berbentuk pola berliku-liku dan merupakan salah satu pola yang efektif untuk tugas-tugas sederhana yang membutuhkan penutupan area secara menyeluruh. Akan tetapi, jalur tersebut tidak cocok untuk area yang kompleks. Gambar 2.4 (a) merupakan ilustrasi dari jalur berliku-liku pada tugas pemetaan.

b. Jalur Pilin

Jalur pilin merupakan jalur pemetaan yang menyebabkan *drone* terbang dengan pola pilin dari pusat menuju keluar area atau sebaliknya. Gambar 2.4 (b) merupakan ilustrasi dari jalur pilin pada tugas pemetaan.

c. Jalur Pola S

Jalur pola S merupakan jalur pemetaan yang menyerupai huruf “S”, sehingga *drone* bergerak secara bergantian dari satu sisi ke sisi lainnya. Jalur pola S digunakan untuk pemetaan hutan atau survei agrikultur. Selain itu, jalur pola S dapat memberikan hasil pemetaan yang efisien tanpa memerlukan banyak data. Gambar 2.4 (c) merupakan ilustrasi dari jalur spiral pada tugas pemetaan.



Gambar 2.6. Ilustrasi jalur pada tugas pemetaan dengan pola (a) berliku-liku, (b) pilin, dan (c) S.

Pada penelitian ini, jenis jalur yang akan digunakan sebagai penelusuran rute untuk *drone* jamak adalah jalur pola S. Jalur tersebut dipilih karena keterbatasan area tugas pemetaan serta untuk melihat kinerja pergerakan *drone* yang telah diidentifikasi, yaitu pergerakan searah sumbu x dan sumbu y.

BAB III

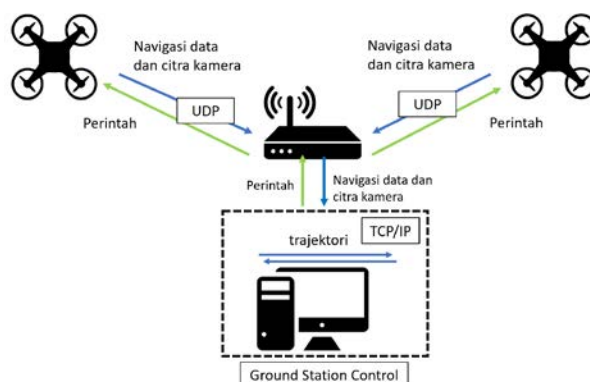
METODE

Bab ini membahas metode penelitian, arsitektur sistem dan pengaturan pengujian, pemilihan pengontrol, pendeteksian beberapa MFP untuk posisi *drone*, dan langkah-langkah pengujian yang dilakukan pada penelitian.

3.1 Arsitektur Sistem dan Pengaturan Pengujian

3.1.1 Komunikasi *Drone*

Objek penelitian yang digunakan pada penelitian adalah *quadrotor* Parrot AR Drone 2.0 sebanyak 2 *drone*. Spesifikasi dari *drone* tersebut ditunjukkan pada Lampiran A. Komunikasi antar *drone* dirancang dengan menggunakan kontrol terpusat yang menganggap setiap *drone* sebagai individu yang berbeda, sehingga tiap *drone* memiliki program yang identik. Akan tetapi, hal yang membedakan dari masing-masing *drone* adalah titik koordinat yang akan dicapai oleh tiap *drone*, masing-masing *drone* juga harus mengetahui posisi dari *drone* lain untuk menghindari tabrakan pada pemetaan. Oleh karena itu, komunikasi antar *drone* untuk berbagi informasi posisi dan komunikasi antara perangkat komputasi (program) dengan *drone* untuk kontrol terpusat diperlukan pada penelitian ini. Gambar 3.1 merupakan arsitektur komunikasi yang digunakan pada penelitian. *Drone* akan terhubung pada *access point* berupa *router* Wi-Fi dengan spesifikasi *router* dapat dilihat pada Lampiran B.



Gambar 3.1. Arsitektur komunikasi.

Pada Gambar 3.1, TCP/IP digunakan untuk melakukan pertukaran data berupa jarak dan posisi antar *drone*. Sementara itu, UDP akan digunakan untuk mengirimkan

BAB IV

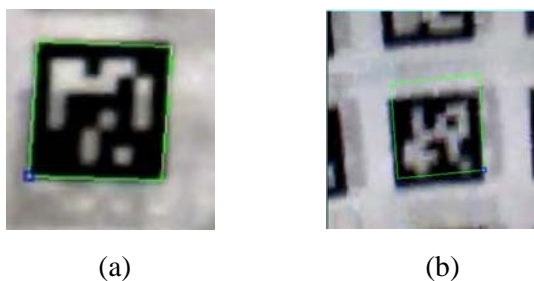
HASIL DAN ANALISIS

Pada bab ini akan dijelaskan terkait hasil dan analisis yang diperoleh dari penelitian yang telah dilakukan, di antaranya identifikasi sistem pemosisian *drone* dengan menentukan kecepatan maksimum *drone* untuk mendeteksi MFP, identifikasi kecepatan *drone* untuk memperoleh parameter pengontrol, pengujian penelusuran posisi *drone* dengan pergerakan ke arah yang berbeda, pengujian algoritma pemisahan, dan pengujian penelusuran titik berbasis kontrol terpusat untuk tugas pemetaan pada *drone*.

4.1 Sistem Pemosisian

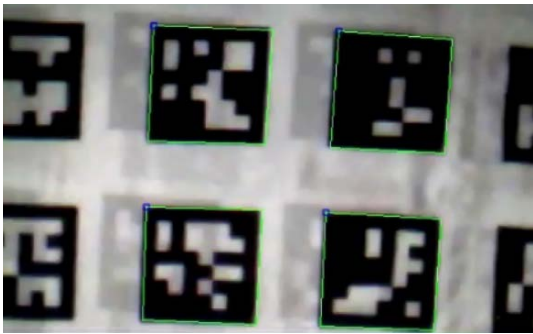
Proses pengambilan citra marka menggunakan kamera bawah *drone* saat *drone* berada pada kondisi terbang dan memberikan perintah maju. Pada penelitian [6], pembacaan MFP dapat terdeteksi hingga nilai masukan kecepatan sebesar 1 m/s, sehingga nilai kecepatan di atas 1 m/s tidak dapat mendeteksi MFP [6]. Pada penelitian [6], ukuran marka yang dideteksi oleh *drone* berukuran 60 cm x 60 cm, sementara pada penelitian ini berukuran 12,5 cm x 12,5 cm. Oleh karena itu, *drone* akan mengambil citra marka dengan kecepatan 50 cm/s.

Pada Gambar 4.1 (a), citra marka MFP yang terdeteksi saat *drone* diam memiliki resolusi yang baik. Akan tetapi, saat *drone* bergerak dengan kecepatan 50 cm/s, citra marka MFP yang ditangkap oleh kamera *drone* tidak sebaik marka yang terdeteksi saat *drone* dalam kondisi diam. Citra marka MFP pada saat *drone* bergerak ditunjukkan pada Gambar 4.1 (b).

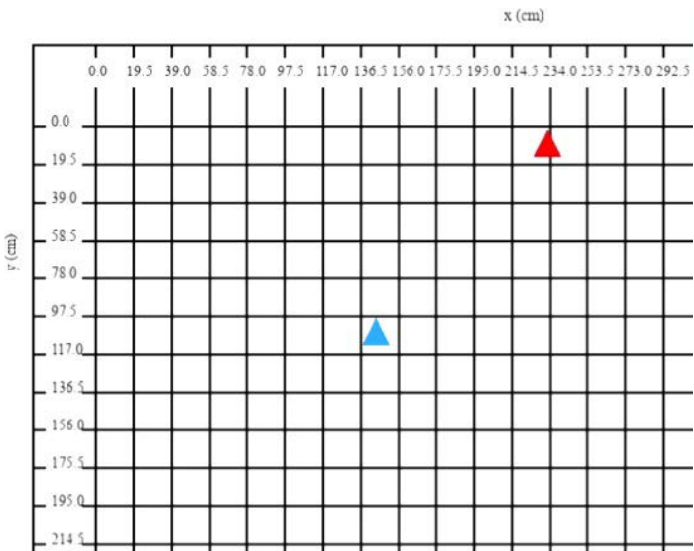


Gambar 4.1. Marka MFP yang terdeteksi oleh drone (a) saat diam dan (b) saat *drone* bergerak dengan kecepatan 50 cm/s.

Meskipun posisi *drone* sesuai perhitungan, yaitu (146,25;107,25). Dengan demikian, resolusi citra marka yang terdeteksi rendah, *drone* tetap mendapatkan informasi ID dari marka tersebut. Pendeteksian beberapa marka dapat memberikan informasi posisi *drone* yang presisi. Gambar 4.2 (a) merupakan hasil deteksi beberapa marka dan koordinat yang diperoleh. Pada proses deteksi beberapa marka tersebut, posisi *drone* berada pada 4 ID dengan koordinat masing-masing adalah (136,5;97,5), (156;97,5), (136,5;117), dan (156;117). Dengan pendeteksian beberapa marka MFP, posisi dari drone yang diperoleh adalah (146,25;107,25). Gambar (4.2) (b) merupakan tampilan UI sinkronisasi posisi *drone* dari deteksi beberapa MFP. Segitiga berwarna biru merepresentasikan *drone* pada kondisi melayang. Berdasarkan tampilan UI, posisi *drone* berada pada koordinat (146,25;107,25) dan sesuai dengan posisi riil.



(a)



(b)

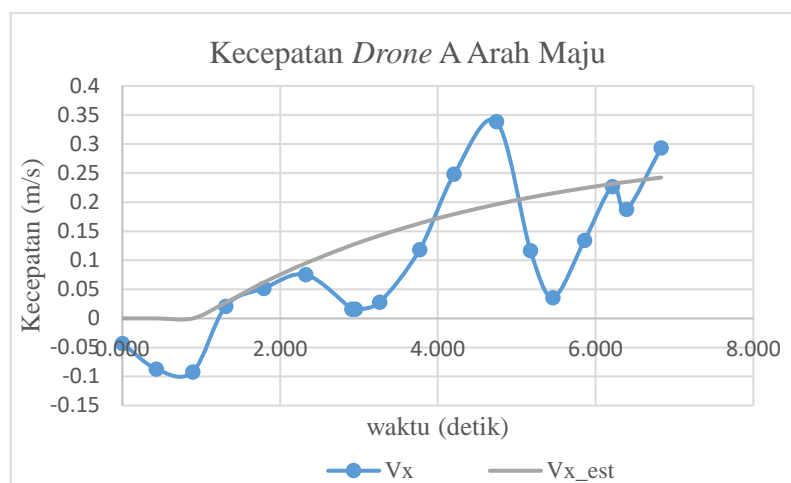
Gambar 4.2. Ilustrasi deteksi beberapa MFP (a) MFP yang terdeteksi dan menunjukan posisi riil *drone* dan (b) posisi *drone* dari hasil pengolahan pada UI.

4.2 Identifikasi Karakteristik *Drone*

Identifikasi karakteristik dilakukan pada kecepatan *drone* untuk menentukan parameter pengontrol yang akan digunakan pada penelitian ini. Identifikasi karakteristik dilakukan pada masing-masing *drone* dengan kondisi ruangan yang sama dan pergerakan arah yang sama.

4.2.1 Hasil Identifikasi Karakteristik *Drone*

Proses identifikasi pergerakan *drone* dilakukan dengan cara memberikan perintah bergerak searah sumbu x dan sumbu y. Pada penelitian ini dilakukan identifikasi pergerakan *drone* pada nilai kecepatan 50 cm/s. Nilai tersebut dipilih untuk memenuhi kebutuhan tugas *drone* dalam menangkap citra pemetaan, sehingga kecepatan yang dimasukkan pada *drone* harus sekecil mungkin. Respons kecepatan yang didapatkan untuk proses identifikasi memiliki bentuk serupa dengan FOPTD pada penelitian [6], sehingga identifikasi respons dapat dilakukan dengan pendekatan FOPTD. Gambar 4.3 merupakan respons sistem dengan nilai kecepatan 0,100 dari *drone* A. Data lengkap identifikasi kecepatan *drone* A dan *drone* B pada Lampiran C.



Gambar 4.3. Hasil identifikasi pergerakan ke kiri dari *drone* A.

Parameter FOPTD untuk setiap arah pergerakan *drone* A ditampilkan pada Tabel 4.1. Fungsi alih kecepatan dengan arah maju pada *drone* A diekspresikan pada Persamaan (4.1). Untuk pergerakan arah lain, parameter K , t_d , dan τ dapat disubstitusikan ke Persamaan (2.29).

Tabel 4.1. Hasil identifikasi kecepatan *drone* A dan *drone* B.

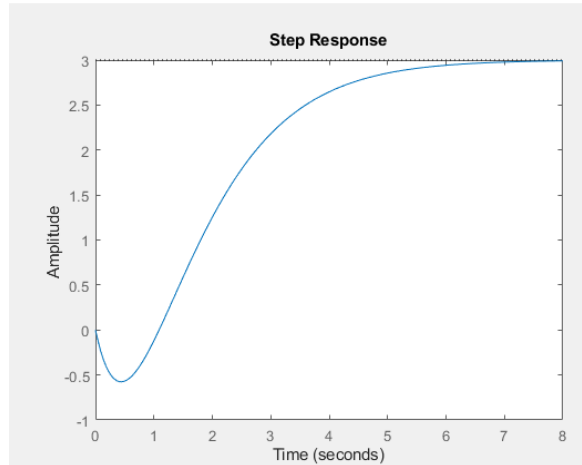
Arah	Drone A				Drone B			
	Parameter FOPTD			SSE	Parameter FOPTD			SSE
	K	τ	t_d		K	τ	t_d	
Maju	2,998	3,543	0,974	0,112	2,999	0,976	1,692	0,524
Mundur	1,681	0,280	0,075	0,070	3,541	0,858	0,564	1,289
Ke Kanan	1,430	0,957	0,560	0,148	1,182	1,000	0,425	0,342
Ke Kiri	6,655	0,523	0,742	1,579	2,353	0,723	0,555	0,152

$$G(s) = \frac{2,998}{3,543 s + 1} e^{-0,974 s} \quad (4.1)$$

Persamaan (3.2) diubah menggunakan pendekatan Padé. Pendekatan Padé dilakukan untuk melakukan pendekatan pada bagian waktu tunda dari fungsi transfer FOPTD. Dengan demikian, Persamaan (4.1) menggunakan pendekatan Padé dapat dituliskan sebagai berikut.

$$G(s) = \frac{2,998}{4,543 s + 1} \times \frac{1 - \frac{0,974}{2} s}{1 + \frac{0,974}{2} s} \quad (4.2)$$

Persamaan (4.2) diberi input step untuk melihat respons fungsi transfer dalam simulasi. Gambar 4.3 merupakan grafik respons input step dari fungsi transfer Persamaan (4.2). Berdasarkan Gambar 4.4 tersebut, sistem memiliki *undershoot* yang diakibatkan oleh komponen *zero* dari fungsi transfer. Komponen *zero* tersebut merupakan bagian dari waktu tunda, Gambar 4.3 pun menunjukkan adanya waktu tunda pada sistem selama 0,974 detik.



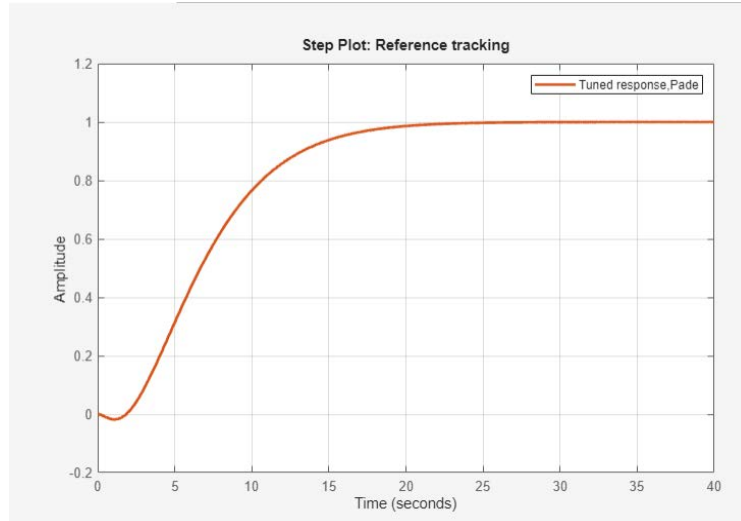
Gambar 4.4. Respons sistem input step untuk *drone* A arah maju

4.2.2 Hasil Penalaan Parameter Pengontrol PD

Persamaan (4.2) digunakan sebagai model sistem yang digunakan untuk perancangan pengontrol. Pada penelitian ini pengontrol yang digunakan merupakan pengontrol PD untuk menghilangkan *undershoot*. Dengan menggunakan Persamaan (4.2) yang merupakan pendekatan Padé, maka proses perancangan pengontrol PD dapat dilakukan dengan menggunakan metode *root locus*. Dengan demikian, parameter pengontrol PD untuk masing-masing *drone* dapat diperoleh, seperti pada Tabel 4.2. Hasil pengontrolan dari *drone* A untuk bergerak maju ditunjukkan oleh Gambar 4.4. Berdasarkan Gambar 4.5, *overshoot* yang dimiliki oleh sistem dapat tereduksi sehingga keluaran tersebut memberikan hasil menuju satu sesuai input yang diberikan.

Tabel 4.2. Parameter pengontrol PD untuk *drone* A dan *drone* B

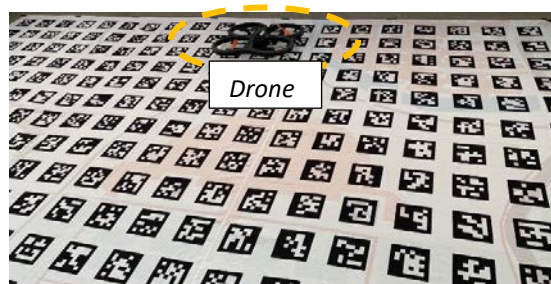
Arah	<i>Drone A</i>		<i>Drone B</i>	
	K_p	T_d	K_p	T_d
Maju	0,487	0,7905	0,501	0,846
Mundur	0,11	0,038	0,217	0,282
Kanan	0,597	0,479	0,639	0,213
Kiri	0,106	0,371	0,302	0,362



Gambar 4.5. Grafik respons sistem *drone* A arah maju dengan pengontrol PD.

4.3 Pengujian Individu

Proses penelusuran posisi *drone* dilakukan secara individu dengan memberikan perintah kepada *drone* untuk terbang mengikuti titik referensi yang diberikan dan *drone* akan mendarat jika titik referensi terakhir telah dicapai. Gambar 4.6 merupakan ilustrasi terkait pengujian individu dengan asumsi tidak ada gangguan angin dari luar.



Gambar 4.6. Ilustrasi pengujian individu.

Pengujian penelusuran posisi dilakukan dengan menggunakan sistem pemosisian deteksi beberapa MFP yang telah dirancang sebelumnya pada Bab III. Titik referensi yang ditentukan membuat pergerakan *drone* searah sumbu x dan sumbu y. Jumlah titik referensi yang diberikan kepada *drone* bergantung dengan arah pergerakan *drone* karena kondisi ruangan dan ukuran karpet MAT yang terbatas.

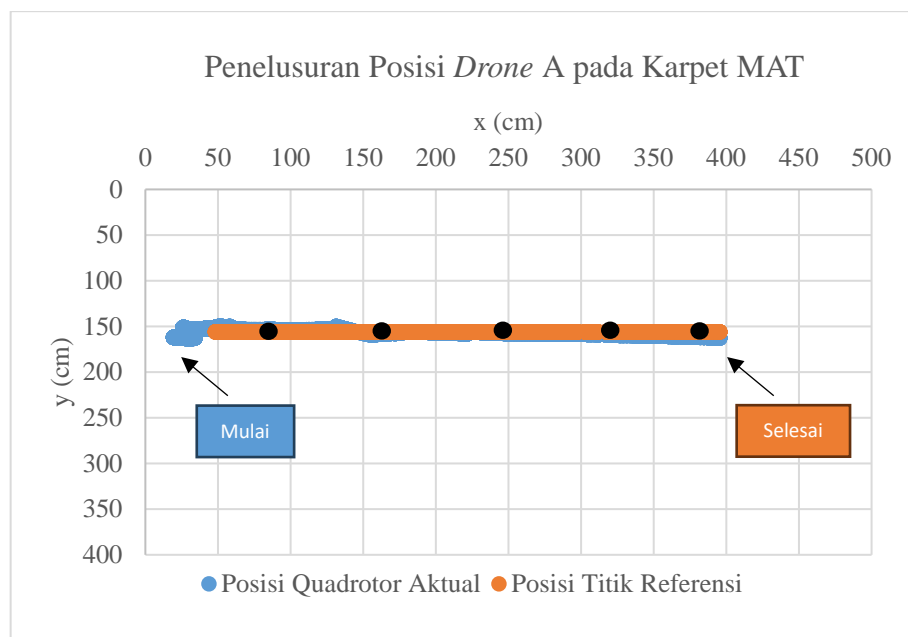
4.3.1 Penelusuran Rute Searah Sumbu X

Gambar 4.7 (a) merupakan hasil pengujian pergerakan maju pada *drone* A searah sumbu x. Berdasarkan Gambar 4.7 (a), pergerakan arah maju pada *drone* dilakukan

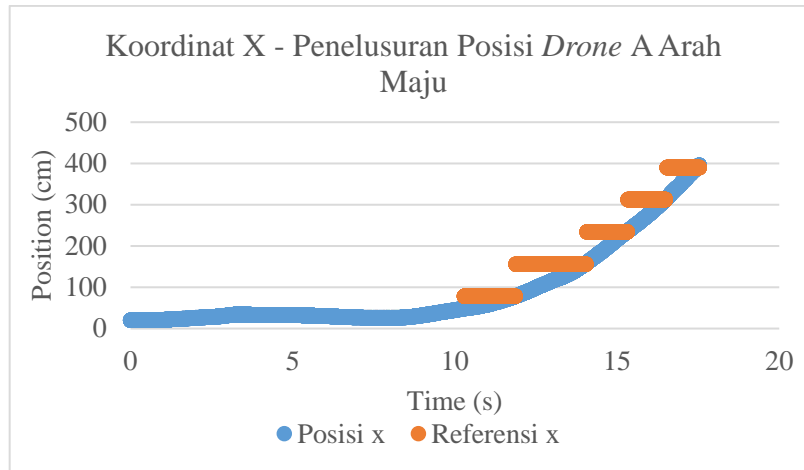
dengan memberikan lima titik target, yaitu (78;156), (156;156), (234;156), (312;156), dan (390;156). Pengujian dilaksanakan setelah *drone* masuk kondisi melayang dan siap menerima perintah pergerakan. Pada Gambar 4.6, posisi awal *drone* tidak di titik $x = 0$, tetapi di titik $x = 19,5$. Hal tersebut dilakukan agar *drone* terhindar dari tabrakan.

Pada proses pengujian ini, *drone* menghadap sejajar dengan sumbu x , sehingga pergerakan maju *drone* ditandai dengan bertambahnya nilai koordinat pada sumbu x , seperti Gambar 4.7 (b). Proses pergantian posisi dilakukan sama seperti pada penelitian [6], bahwa pergantian titik referensi tidak dilakukan dengan jeda, melainkan berubah ketika mencapai radius 30,0 cm dari titik target terkini. Untuk menyelesaikan penelusuran titik yang diberikan pada pergerakan maju, waktu yang dibutuhkan *drone* A menjalankan tugas tersebut adalah 17 detik.

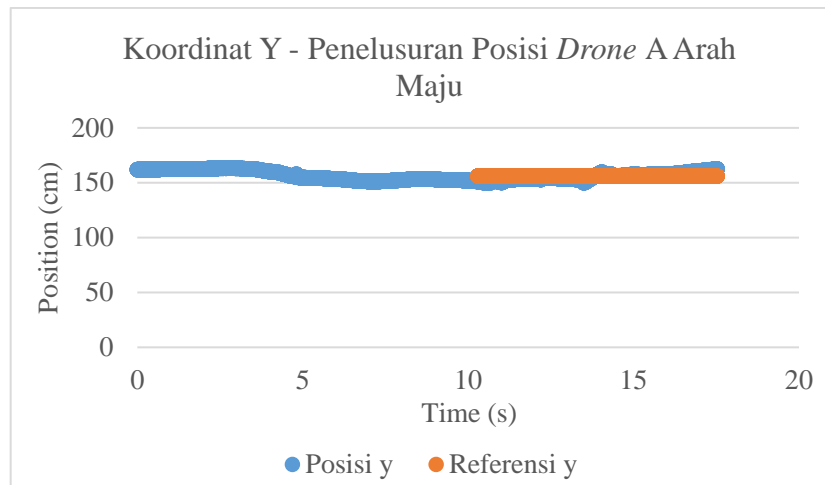
Gambar 4.7 (c) merupakan posisi *drone* A pada sumbu y . Pada Gambar 4.7 (c) *drone* melayang di sekitar $y = 156$ dan menerima perintah untuk menjaga koordinat y .



(a)



(b)



(c)

Gambar 4.7. Penelusuran posisi dengan arah maju pada sumbu x (a) rekaman posisi *drone* saat bergerak maju, (b) profil posisi pada sumbu x dan (c) profil posisi sumbu y.

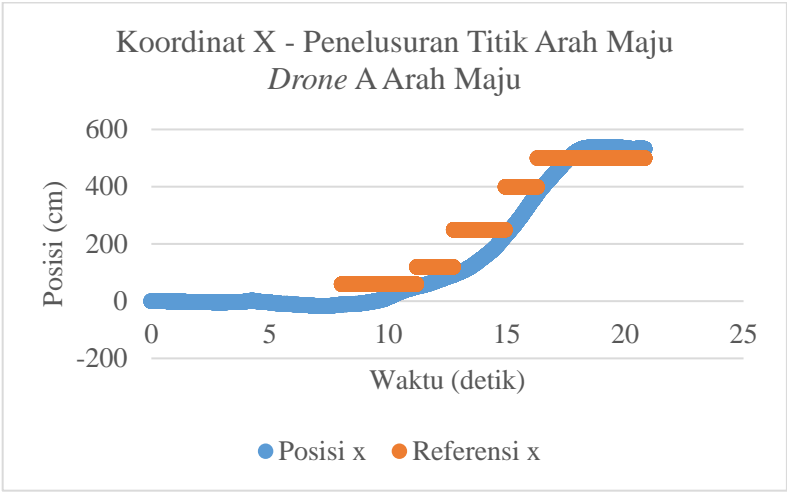
Untuk membandingkan hasil penelusuran penelitian ini dengan penelitian [6] dapat dilihat perbandingan nilai RMSE pada Tabel 4.3. Tabel 4.3 menunjukkan bahwa pada penelitian ini nilai RMSE penelusuran posisi *drone* lebih rendah dibandingkan penelitian [6], sebagai contoh nilai RMSE *drone* A saat bergerak searah sumbu x positif (maju) nilai RMSE awalnya 19,3 cm dan pada penelitian ini menurun menjadi sebesar 5,2 cm.

Tabel 4.3. Perbandingan nilai RMSE pada penelitian 2022 dan penelitian 2024.

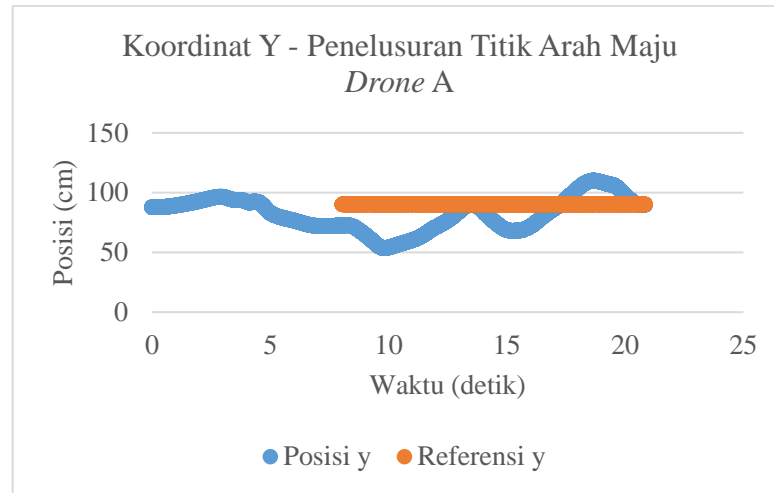
		Nilai RMSE (cm)			
		Maju	Mundur	Kanan	Kiri
Penelitian 2022	Drone A	19,3	13,4	9,9	6,6
	Drone B	14,4	10,1	8,7	9,5
Penelitian 2024	Drone A	5,2	7,8	7,7	6,3
	Drone B	5,0	6,7	8,5	8,6

Apabila posisi sumbu y pada *drone* A Gambar 4.7 (c) dibandingkan dengan Gambar 4.8 (c) yang merupakan posisi *drone* A pada sumbu y dari hasil penelitian [6] dengan nilai posisi y = 90, dapat dilihat bahwa posisi *drone* A pada sumbu y Gambar 4.7 (b) lebih terjaga pada rentang nilai yang diberikan jika dibandingkan dengan posisi *drone* pada Gambar 4.8 (b).

Hal tersebut dikarenakan sistem pemosisian yang digunakan tidak lagi menggunakan kombinasi sistem pemosisian dan sistem pengontrol PD, melainkan menggunakan pendeteksian beberapa MFP dan penggunaan titik acuan yang merupakan titik tengah kamera sebagai titik pusat *drone*. Proses penelusuran posisi pada pergerakan arah mundur diberikan titik referensi yang merupakan kebalikan dari titik referensi pergerakan maju *drone*, yaitu (312;156), (234;156), (156;156), (19,5;156). Grafik pergerakan arah mundur *drone* A dan drone B dapat dilihat pada Lampiran D.



(a)



(b)

Gambar 4.8. Penelusuran posisi pada penelitian Giga dengan arah maju pada sumbu x (a) posisi pada sumbu x dan (b) posisi sumbu y [6].

4.3.2 Penelusuran Rute Searah Sumbu Y

Proses penelusuran posisi pada pergerakan arah kanan dan kiri dilakukan dengan menyimpan posisi *drone* tetap menghadap sumbu x. Oleh karena itu, pergerakan *drone* ke arah kanan dan kiri ditandai dengan berubahnya nilai koordinat pada sumbu y. Titik referensi yang digunakan untuk pengujian arah kanan dan kiri lebih sedikit dibandingkan pengujian arah maju. Hal tersebut dikarenakan ukuran koordinat sumbu y yang lebih kecil dibandingkan koordinat x pada karpet MAT yang dirancang, seperti pada Gambar 3.9. Selain itu, jumlah titik referensi dikurangi untuk menghindari tabrakan dengan dinding dan mengurangi efek turbulensi akibat baling-baling. Pergerakan searah sumbu y pada *drone* A dapat dilihat pada Lampiran D. Dengan demikian, waktu tempuh *drone* A untuk melakukan pergerakan ke arah kanan sebesar 15 detik dan waktu tempuh pergerakan ke arah kiri sebesar 14 detik.

Pengujian penelusuran posisi dilakukan juga pada *drone* B dengan langkah-langkah yang sama, seperti pada *drone* A. Perbandingan hasil pengujian untuk kedua *drone* ditunjukkan pada Tabel 4.4. Berdasarkan hasil tersebut, waktu tempuh dari tiap *drone* berbeda dikarenakan karakteristik *drone* yang berbeda. Pergerakan *drone* searah sumbu y memiliki jarak tempuh yang lebih dekat dibandingkan dengan arah searah sumbu x, sehingga memberikan waktu tempuh yang lebih cepat juga. Akan tetapi, pada *drone* B, waktu tempuh ke arah kanan lebih lambat dibandingkan ke

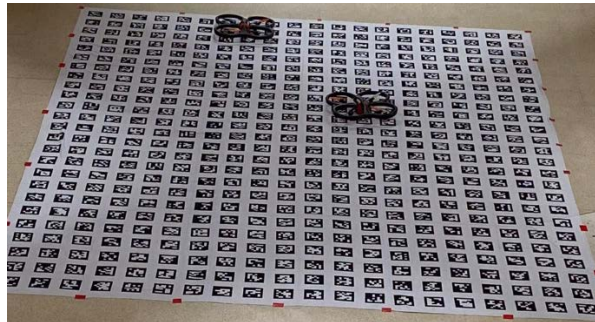
arah kiri karena kecepatan *drone* saat bergerak ke arah kanan lebih kecil dibandingkan saat bergerak mundur.

Tabel 4.4. Waktu tempuh pengujian penelusuran posisi tiap *drone* (detik).

	Arah Pergerakan <i>Drone</i> (detik)			
	Maju	Mundur	Kanan	Kiri
<i>Drone A</i>	17	17	15	14
<i>Drone B</i>	23	19	20	18

4.4 Pengujian Berkelompok

Pengujian berkelompok dilakukan dengan dua pengujian, yaitu pengujian algoritma pemisahan untuk melihat respons *drone* dalam menghindari tabrakan dan pengujian tugas pemetaan. Gambar 4.9 merupakan ilustrasi pengujian berkelompok, kedua *drone* terbang di atas karpet MAT yang menunjukkan koordinat posisi dalam satuan cm.



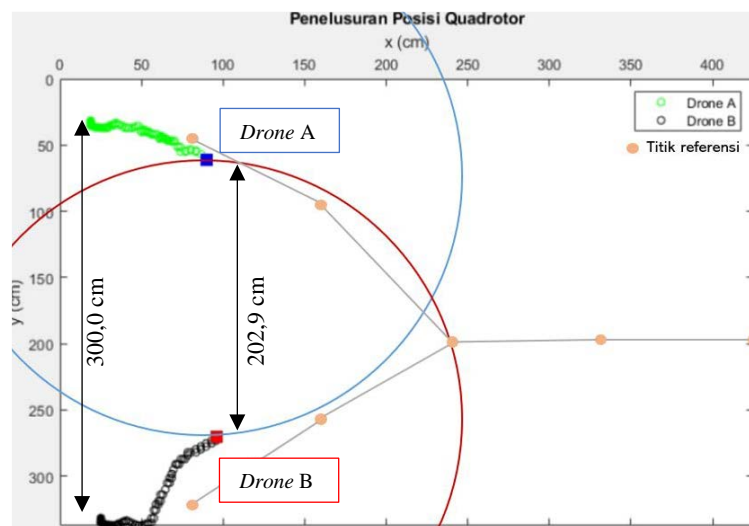
Gambar 4.9. Ilustrasi pengujian berkelompok.

4.4.1 Pengujian Algoritma Pemisahan untuk Menghindari Tabrakan

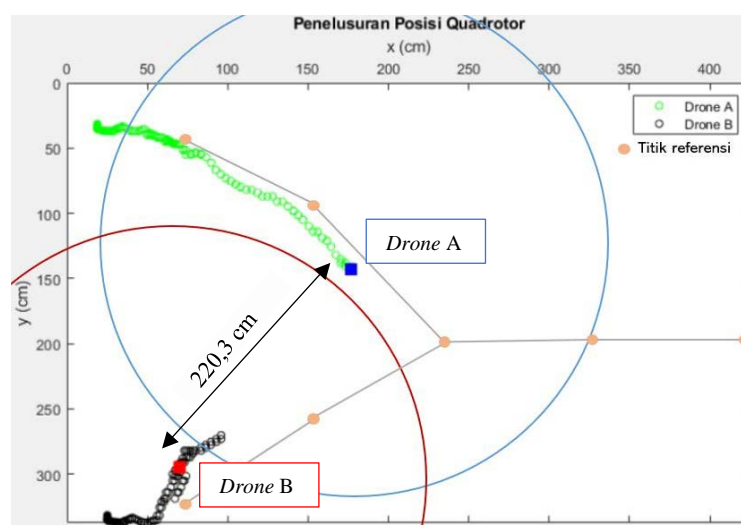
Proses pengujian algoritma pemisahan dilakukan dengan memberikan jalur yang beresiko tabrakan untuk kedua *drone*. Pada pengujian ini, jarak minimum antar titik pusat *drone A* dan *drone B* ditetapkan sebesar 200 cm. Dengan demikian, ketika kedua *drone* berada pada jarak kurang dari 200 cm, maka kedua *drone* akan menghindar menjauhi satu sama lain. Sebaliknya, jarak maksimum antar *drone* ditetapkan sebesar 400 cm. Gambar 4.8 merupakan hasil dari pengujian algoritma pemisahan. Ketika kedua *drone* berada pada jarak 200 cm, *drone* akan memberikan arah pergerakan untuk menghindar. Pada proses pengujian, *drone A* akan tetap

bergerak menuju titik C dan *drone* B akan bergerak berlawanan arah (mundur) ke titik G.

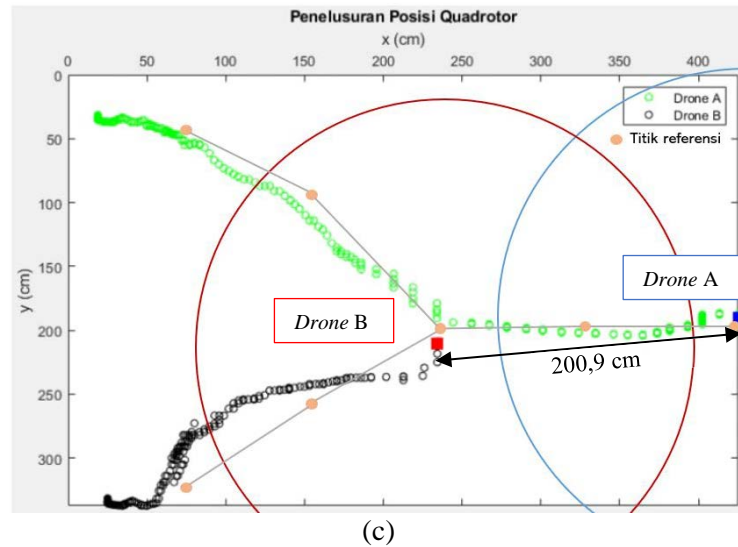
Respons pergerakan tersebut dapat dilihat pada Gambar 4.7. Pada Gambar 4.10 (a), jarak antar *drone* mendekati jarak minimum yang diperbolehkan pada $t = 3$ detik dan posisi *drone* B yang berada pada posisi (92,9;275,5). Oleh karena itu, *drone* B bergerak mundur ke posisi (68,5;300) saat $t = 5$ detik, seperti Gambar 4.10 (b) dan jarak antar *drone* menjadi 220,3 cm. Karena jarak antar *drone* lebih dari jarak minimum, *drone* B kembali bergerak mengikuti rute yang diberikan, yaitu menuju titik C apabila *drone* A telah melewati titik C dan jarak antar *drone* lebih dari 200 cm, seperti Gambar 4.10 (c).



(a)



(b)

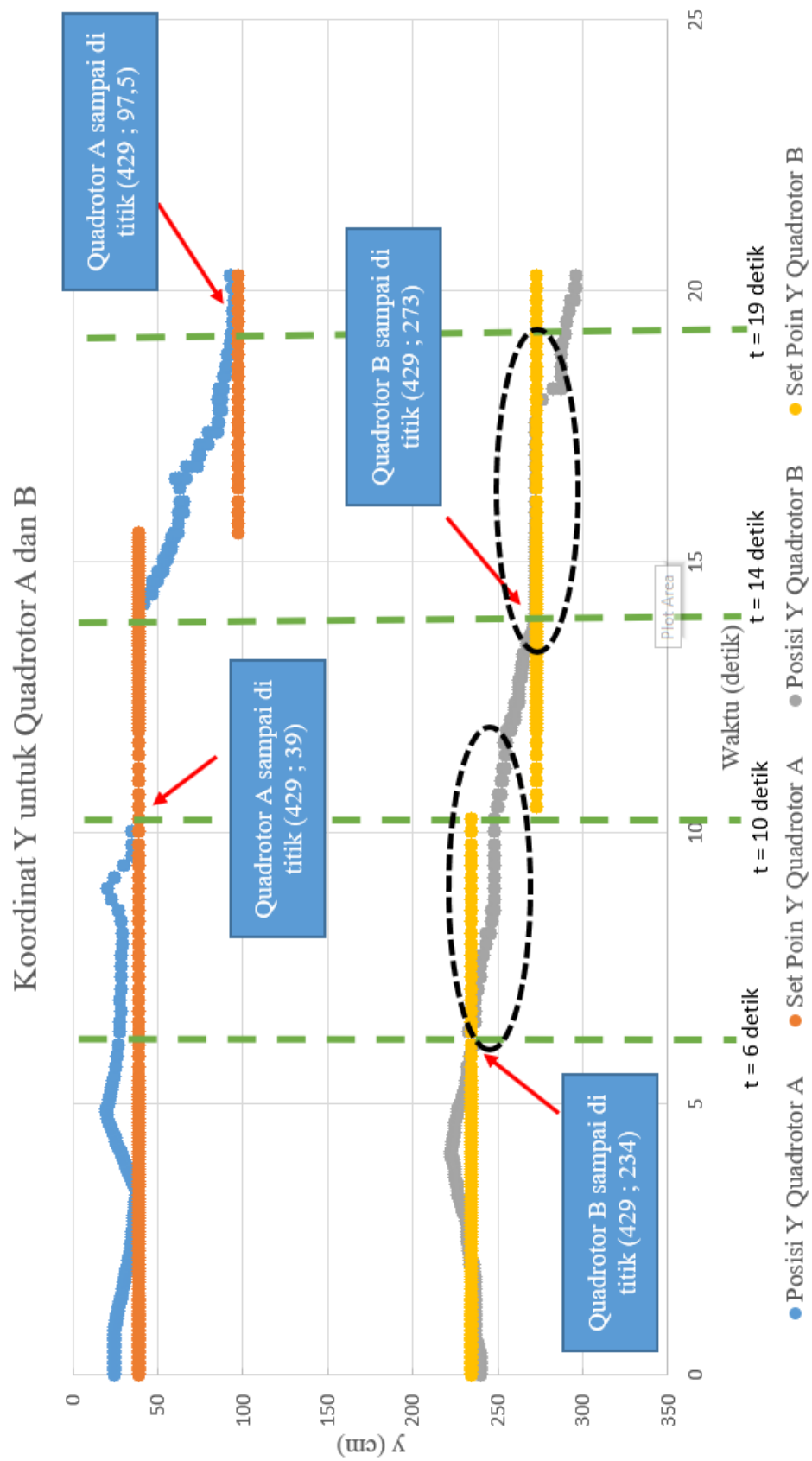


Gambar 4.10. Rekaman posisi *drone* pada pengujian algoritma pemisahan (a) pada $t = 3$ detik, (b) pada $t = 5$ detik, dan (c) pada $t = 11$ detik.

4.4.2 Pengujian untuk Tugas Pemetaan

Gambar 4.11 merupakan data yang diambil hingga detik 20 saat kedua *drone* mengalami pergeseran pada sumbu y . Gambar 4.11 menunjukkan bahwa *drone B* sampai lebih dahulu pada titik referensi (429;234) pada detik 6 dan *drone A* belum sampai di titik (429;39). Oleh karena itu, *drone B* akan menunggu dengan melayang di titik (429;234) hingga *drone A* sampai di titik (429;39). Berdasarkan gambar tersebut, *drone A* mencapai titik (429;39) pada detik 10.

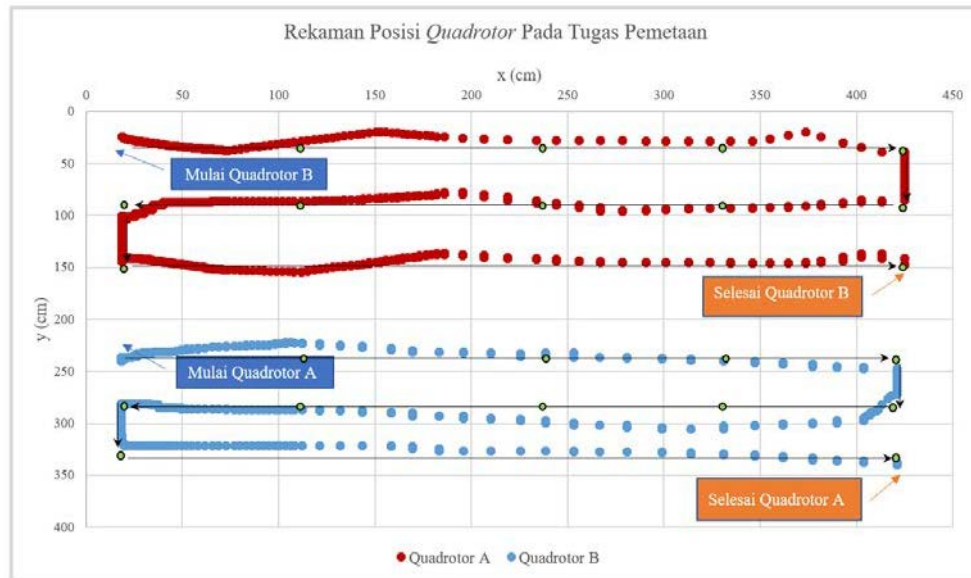
Dengan demikian, skenario pergerakan *drone* pada saat melakukan tugas pemetaan pada *drone* jamak untuk menghindari tabrakan dan menjaga jarak aman antar *drone* pada nilai 200 cm hingga 400 cm dapat digunakan pada penelitian ini. Ketika *drone A* dan *drone B* sudah mengalami pergerakan ke sumbu y dengan titik referensi masing-masing (429;97,5) dan (429;273), maka kedua *drone* akan bergerak bersama-sama pada arah sumbu x .



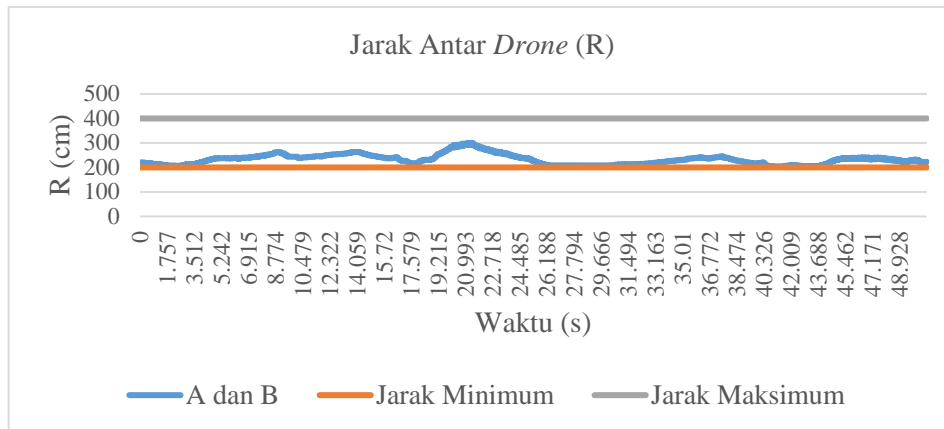
Gambar 4.11. Penelusuran posisi *drone* A dan *drone* B selama 20 detik.

Kedua *drone* akan melakukan tugas pemetaan dengan mengikuti titik target yang diberikan sebagai jalur pemetaan. Radius toleransi yang ditetapkan terhadap target sebesar 30,0 cm. Gambar 4.12 merupakan hasil penelusuran posisi *drone* jamak saat melakukan tugas pemetaan. Gambar 4.12 menunjukkan posisi *drone* tidak tepat berada di posisi awal, yaitu (19,5;39) untuk *drone* A dan (19,5;234) untuk *drone* B. Berdasarkan Gambar 4.12, kedua *drone* mengikuti jalur sesuai titik referensi yang diberikan, meskipun *drone* B mengalami pergeseran dari titik referensi yang diberikan, yaitu pada posisi (281;302). Pergeseran yang dialami oleh *drone* B diakibatkan adanya turbulensi dari baling-baling setiap *drone* serta jalur yang diberikan sebagai referensi *drone* melakukan pemetaan berbentuk titik. Penggunaan jalur referensi berupa titik menyebabkan *drone* bergerak tidak mengejar lurus mengikuti jalur referensi yang ditentukan. Akan tetapi, pergeseran tersebut berada pada radius toleransi.

Tugas pemetaan akan selesai jika kedua *drone* menemukan titik referensi terakhir. Apabila hanya satu *drone* saja yang baru sampai di titik referensi terakhir, *drone* tersebut akan berada pada kondisi melayang menunggu *drone* lain mencapai titik referensi terakhir juga. Pada penelitian ini, *drone* A berhasil menemukan titik referensi terakhir, yaitu titik (429;156) pada detik 49. Sedangkan, *drone* B berhasil menemukan titik referensi terakhir, yaitu titik (429;351) pada detik 51. Dengan demikian, *drone* akan berada pada kondisi melayang mulai dari detik 49 hingga detik 51, sehingga kedua *drone* akan mendarat pada detik 51,317 sesuai dengan algoritma yang dirancang. Selain itu, Gambar 4.9 (b) menunjukkan bahwa jarak antar *drone* berada pada nilai rata-rata 232,2 cm.



(a)



(b)

Gambar 4.12. Hasil pengujian tugas pemetaan (a) posisi *drone* pada koordinat *cartesius* dan (b) jarak tiap *drone* relatif terhadap waktu.

Untuk mengetahui performa antar *drone* dalam melakukan tugas pemetaan sesuai dengan jalur yang ditentukan, maka perhitungan performa tersebut menggunakan RMSE, seperti penelitian sebelumnya. Perhitungan RMSE dilakukan dengan menghitung galat antara posisi *drone* secara aktual dengan titik referensi yang diberikan. Dengan demikian, hasil yang didapat oleh *drone* A adalah 12,3 cm dan *drone* B adalah 22,5 cm. Selain itu, jarak tempuh tiap *drone* untuk melakukan tugas pemetaan jika dihitung mulai berdasarkan posisi awal hingga akhir dapat dilihat pada Tabel 4.5. Dengan demikian, pengujian untuk tugas untuk kedua *drone* pada penelitian ini memberikan performa yang baik dengan luas area tugas yang terbatas, yaitu sebesar 420 cm x 350 cm.

Tabel 4.5. Data hasil tugas pemetaan.

	Jarak Tempuh	Waktu Tempuh	Error_d	RMSE
<i>Drone A</i>	1405,5 cm	49 detik	151,4 cm ²	12,3 cm
<i>Drone B</i>	1405,5 cm	51 detik	507,4 cm ²	22,5 cm

BAB V

KESIMPULAN

Bab ini membahas mengenai simpulan dan saran terkait penelitian ini.

5.1 Kesimpulan

Kesimpulan penelitian yang diperoleh adalah sebagai berikut.

1. Teknik pendeteksian beberapa marka MFP meningkatkan kemampuan penelusuran rute pada ketinggian terbang 80 cm. Pada penelitian ini, nilai RMSE untuk *drone* A bergerak searah sumbu x positif bernilai 5,2 cm dan pada penelitian [6] bernilai 19,3 cm.
2. *Drone* dapat menghindari tabrakan terhadap *drone* lainnya dengan menggunakan algoritma pemisahan. Ketika jarak kedua *drone* berada pada jarak minimum, yaitu kurang dari atau sama dengan 200 cm, maka algoritma pemisahan akan berjalan.
3. Performa *drone* dihitung dengan menggunakan nilai RMSE. Pada penelitian, tugas pemetaan dilakukan dengan jarak tempuh sebesar 1405,5 cm dan waktu tempuh 51 detik dihitung dari posisi awal hingga posisi mendarat. Nilai RMSE yang diperoleh untuk masing-masing *drone* sebesar 12,3 cm untuk *drone* A dan 22,5 cm untuk *drone* B. Berdasarkan hasil pengujian, *drone* mampu melakukan tugas pemetaan dengan baik, meskipun *drone* bekerja pada area yang terbatas, yaitu 5 m x 4 m.

5.2 Saran

Beberapa hal yang disarankan sebagai arah pengembangan penelitian selanjutnya, yaitu:

1. Kontrol untuk turbulensi antar *drone*, sehingga jarak antar *drone* dapat diperkecil.
2. Mengontrol sudut hadap *drone* searah pergerakan *drone*, sehingga pengolahan citra marka menjadi koordinat posisi *drone* dapat presisi.
3. Mengembangkan Solusi MAPF untuk koordinasi antar *drone*, sehingga mendapatkan rute pemetaan yang lebih kompleks dan efisien.

DAFTAR PUSTAKA

- [1] Tahir, A., Böling, J., Haghbayan, M.H., Toivonen, H.T., dan Plosila, J. (2019): Swarm of unmanned aerial vehicles – a survey, *Journal of Industrial Information Integration*, **16**, 1 – 7 .
- [2] Faessler, M., Franchi, A., dan Scaramuzza, D. (2018): Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories. *IEEE Robotics and Automation Letters*, **3**, 620 – 626 .
- [3] Lin, Y., & Saripalli, S. (2017): Path Planning using 3D dubins curve for unmanned aerial vehicles. 2017 *International Conference on Unmanned Aircraft Systems (ICUAS)*, 322 – 329 .
- [4] Yu, J., Yan, C., dan Huang, M. (2019): Research of consistency problem for quadrotor UAV system with leader-follower, *Proceedings - 2019 Chinese Automation Congress, CAC 2019*, 616 – 621.
- [5] K. M. Kabore and S. Güler. (2022): Distributed formation control of drones with onboard perception, *IEEE/ASME Transactions on Mechatronics*, **27**, 3121 – 3131.
- [6] Pratama, G.V. (2021): Kontrol formasi quadrotor menggunakan metode cyclic leader follower SFM sebagai media pelacakan lokasi, Tesis Program Magister, Institut Teknologi Bandung.
- [7] B. Li, J. Wu, X. Tan and B. Wang. (2020): ArUco marker detection under occlusion using convolutional neural network. *International Conference on Automation, Control and Robotics Engineering (CACRE)*, 706 – 711.
- [8] Reynolds, C. W. (1987). “Flocks herds, and schools a distributed behavioral model.” *Computer Graphics*, **21**, 25 – 34.
- [9] Ekawati, E., Joelianto, E., dan Mukhlis F. (2012): Nonlinear predictive control design for quadrotor trajectory using sequential dynamic operability framework, *UNSYS Digital*, 8.
- [10] Kuchеров, D., Kozub, A., Sushchenko, O., dan Skrynkovskyy, R. (2021): Stabilizing the spatial position of a quadrotor by the backstepping procedure, *Indonesian Journal of Electrical Engineering and Computer Science*, **23**, 1188 – 1199.
- [11] Rafifandi, R., Asri, D. L., Ekawati, E., dan Budi, E. M. (2019): Leader–follower formation control of two quadrotor UAVs, *SN Applied Sciences*, **1**, 1 – 12.
- [12] Ogata, K. *Modern Control Engineering*. Prentice Hall, 2010.
- [13] Y. Y. Nazaruddin. *Sistem Kontrol Otomatik: Analisis dan Desain*, I. Bandung: ITB Press, 2022.
- [14] Mondéjar-Guerra, V., Garrido-Jurado, S., Muñoz-Salinas, R., Marín-Jiménez, M. J., dan Medina-Carnicer, R. (2018): Robust identification of fiducial markers in challenging conditions, *Expert Systems with Applications*, **93**, 336–345.

- [15] *ArUco Marker Detection Using Open CV*, data diperoleh melalui situs internet: https://www.delftstack.com/howto/python/opencv-aruco-markers/#google_vignette. Diunduh pada tanggal 15 Desember 2023.
- [16] *Detection od ArUco Markers*, data diperoleh melalui situs internet: https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html. Diunduh pada tanggal 15 Desember 2023.
- [17] Hansun, Seng. (2013): A new approach of moving average method in time series analysis, *IEEE*, 1 – 4.
- [18] Budin, S. dan Riadi I. (2019): Traffic shaping menggunakan metode HTB (hierarchical token bucket) pada jaringan nirkabel, *Buletin Ilmiah Sarjana Teknik Elektro*, **1**, 144 – 152 .
- [19] Nazir, R., laghari, A. A., Kumar, K., David, S., dan Ali, M. (2021): Survey on wireless network security, *Archives of Computational Methods in Engineering*.
- [20] Forouzan, B. A. (2005): *TCP/IP Protocol Suite (3 ed.)*, USA: McGraw-Hill, Inc.
- [21] Kessler, G. C. (2010): *An Overview of TCP/IP Protocols and the Internet, InterNIC*.
- [22] AL-Dhief, F. T., Sabri, N., Latiff, N. M. A., Malik, N., Abbas, M., Albader, A., Mohammed, M. A., AL-Haddad, R. N., Salman, Y. D., dan Khanapi, M. (2018): Performance comparison between TCP and UDP protocols in different simulation scenarios, *International Journal of Engineering & Technology*, **7**, 172–176.
- [23] Maruyama, N., Saito, D., Hashimoto, Y. et al. (2019): Dynamic organization of flocking behaviors in a large-scale boids model. *J Comput Soc Sc* **2**, 77–84.
- [25] Piskorski, S., Brulez, N., Eline, P., dan D’Haeyer, F. (2012): Ar. Drone develepor guide, *Parrot, sdk*, **1**.

LAMPIRAN A. SPESIFIKASI *DRONE* PARRROT ARDRONE 2.0

No.	Kategori	Spesifikasi
1.	Kamera	Horizontal: HD 720p 30 fps (sensor CMOS dengan lensa lebar 90 Derajat) Vertikal: QVGA 360p 30fps
2.	Tipe Perangkat Input	USB 2.0 <i>high speed for extensions</i>
3.	Penyimpanan	1 Gbit DDR2 RAM 200MHz
4.	Tipe Baterai	<i>Lithium Polymer Battery</i>
5.	Kapasitas Baterai	(3 sel, 11,1 V, 1000 mAh) 4 motor <i>brushless</i> , (35000 rpm, <i>power</i> : 15W) <i>Discharging capacity</i> : 10 C <i>Battery charging time</i> : 90 minutes <i>Flying time</i> : about 12 minutes
6.	Dimensi	517 mm x 451 mm
7.	Massa	Total massa 380 gram dengan <i>outdoor hull</i> dan 420 gram dengan <i>indoor hull</i>
8.	Lainnya	1 GHz 32 bit ARM Cortex A8 <i>processor</i> dengan 800MHz video DSP Linux 2.6.32 <i>3 axis gyroscope</i> 2000°/second precision <i>3 axis accelerometer</i> ± 50 mg precision <i>3 axis magnetometer</i> 6° precision Kecepatan : 5 m/s

LAMPIRAN B. SPESIFIKASI *ROUTER* WI-FI

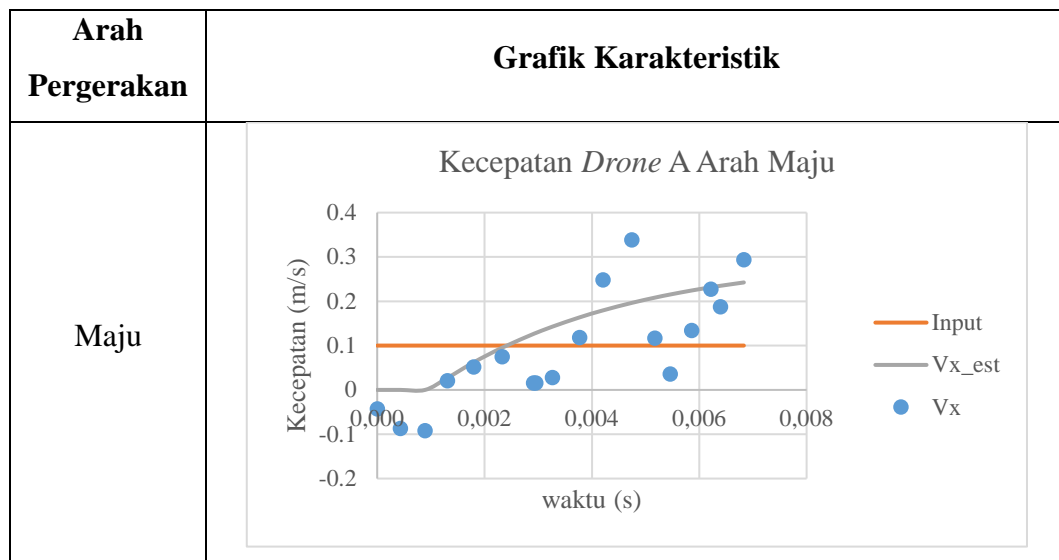
No.	Kategori	Spesifikasi
1.	<i>Standards</i>	Wi-Fi 4 IEEE 802.11n/b/g 2,4 GHz
2.	Kecepatan Wi-Fi	N300 2,4 GHz: 300 Mbps
3.	<i>Processor</i>	Single-Core CPU
4.	Daya	5 V dan 0,6 A
5.	Protokol	IPv4 IPv6
6.	Dimensi	115 mm x 106,7 mm x 24,3 mm
7.	Data Transmisi	CE: < 20 dBm (2,4GHz) FCC: < 30 dBm
8.	Mode Kerja	<i>Router Mode</i> <i>Access Point Mode</i> <i>Range Extender Mode</i> <i>WISP Mode</i>
9.	Lainnya	Keamanan Jaringan: SPI Firewall, <i>Access Control</i> , IP & MAC Binding, Application Layer Gateway Quality of Service: <i>QoS by Device</i>

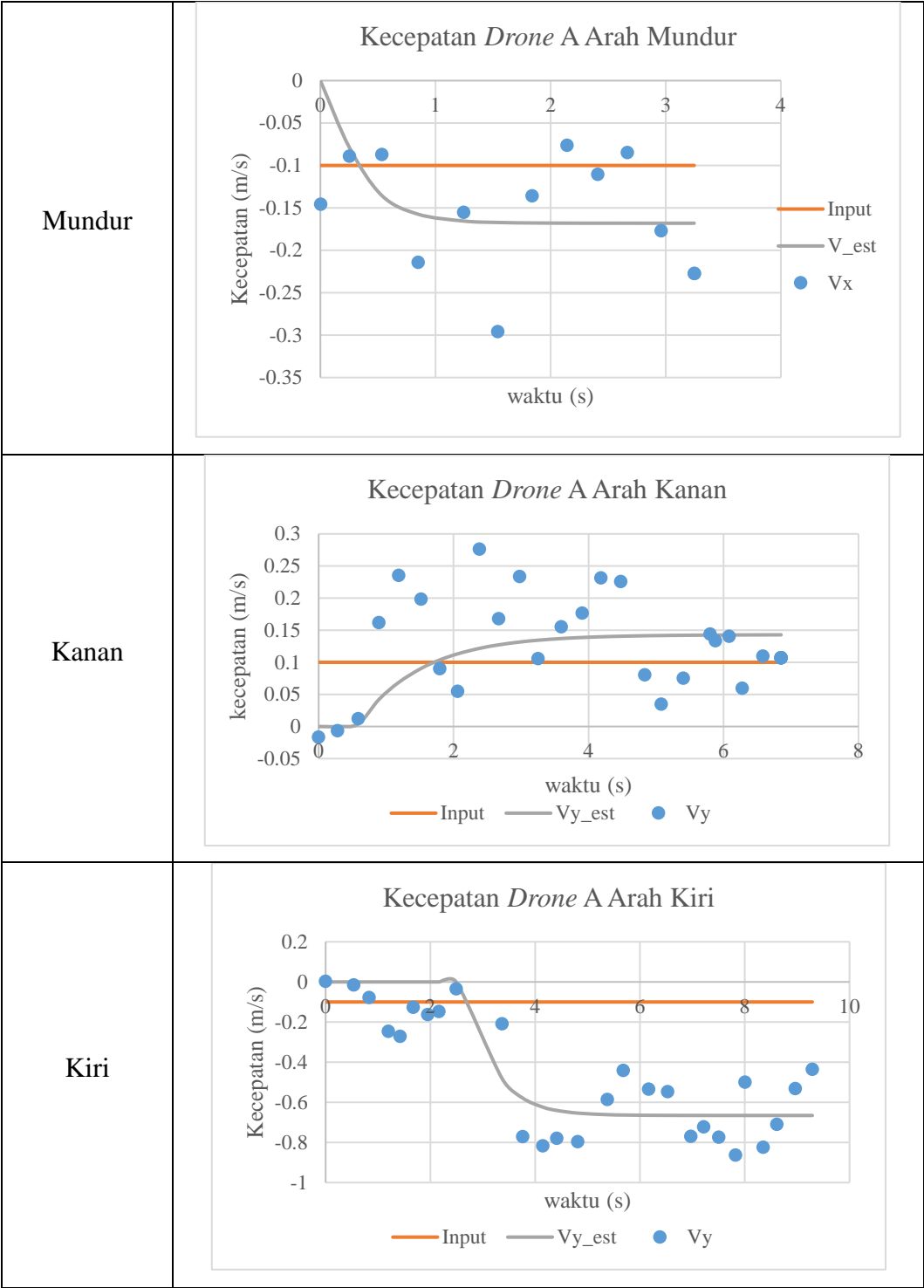
LAMPIRAN C. HASIL IDENTIFIKASI KARAKTERISTIK *Drone*

Identifikasi karakteristik dilakukan pada *drone* A dan *drone* B dengan kondisi ruangan dan arah pergerakan yang sama. Proses identifikasi dilakukan untuk merancang pengontrol yang akan digunakan pada *drone*. Berikut merupakan hasil identifikasi karakteristik *drone* A dan *drone* B. Data keseluruhan dapat diakses melalui bit.ly/DataKarakteristikQuadrotor.

C.1 Grafik Karakteristik Kecepatan *Drone* A

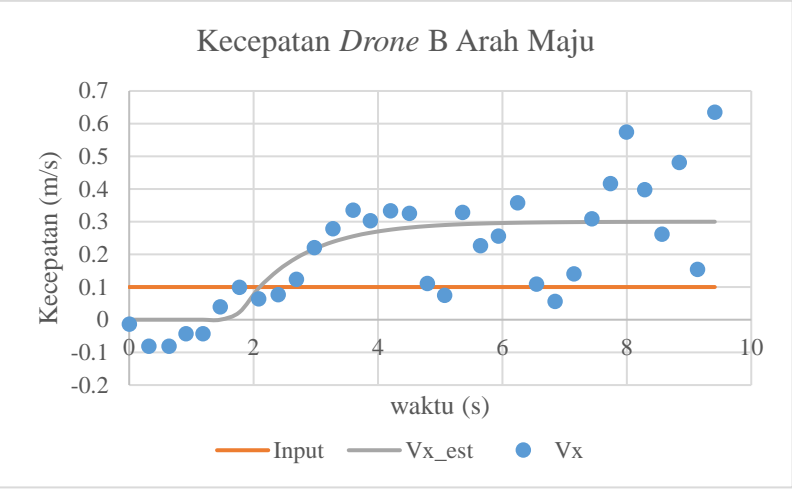
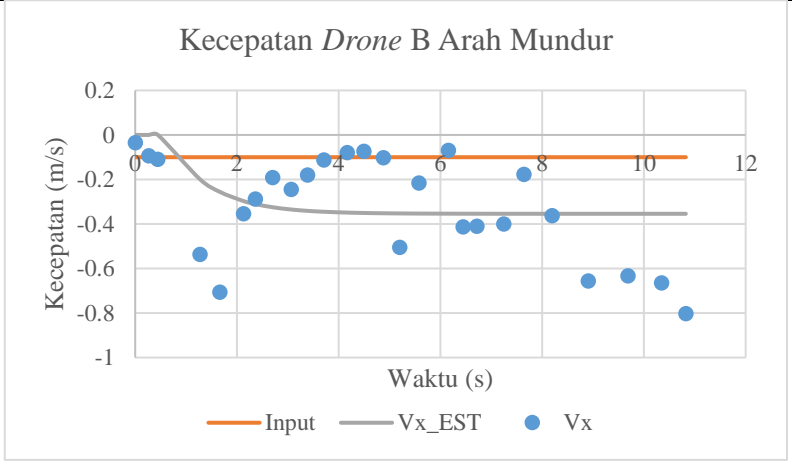
Berdasarkan data yang diperoleh dari identifikasi kecepatan *drone* akan didapatkan parameter fungsi transfer orde satu atau FOPTD. Dengan menggunakan fungsi transfer tersebut, dapat ditentukan karakteristik dan pengontrol yang dapat digunakan pada penelitian ini. Berikut grafik karakteristik untuk *drone* A.

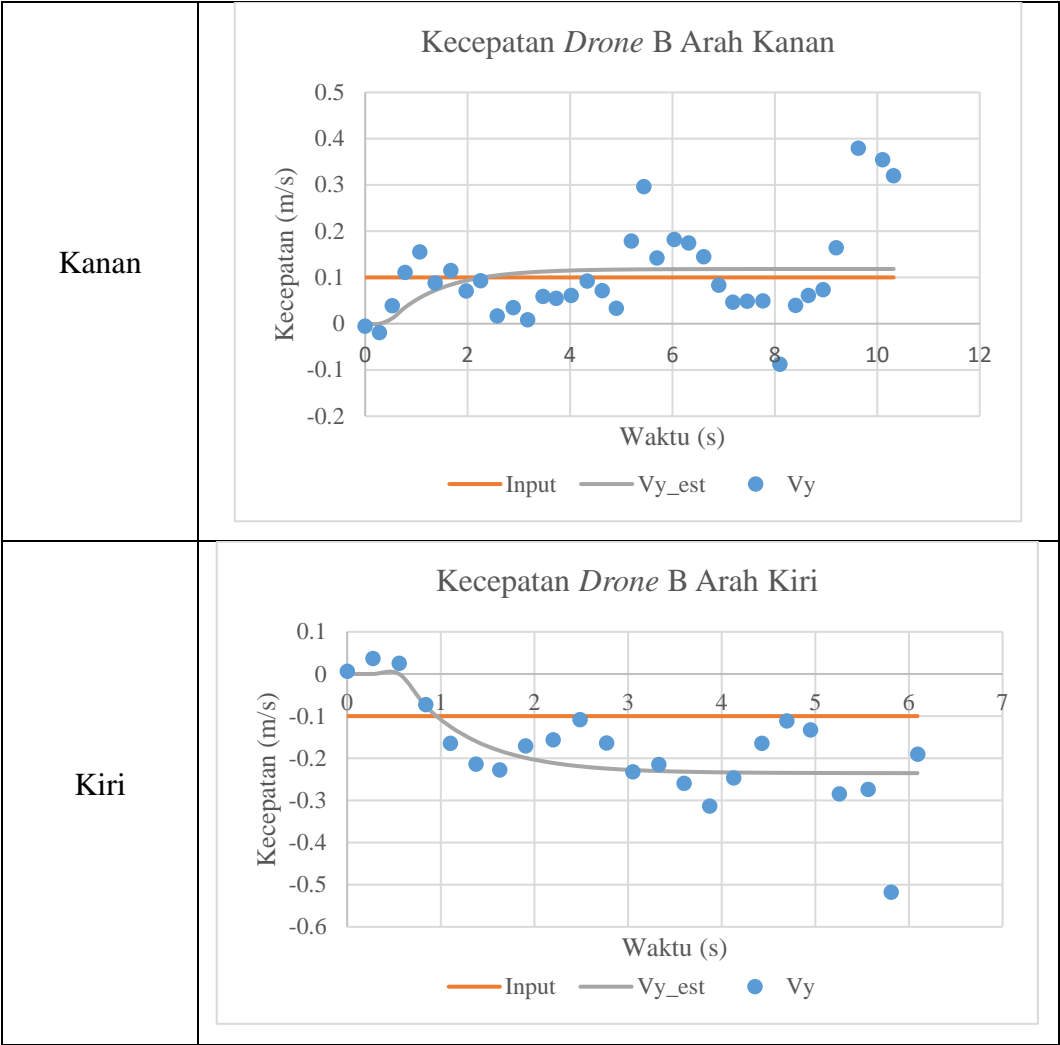




C.2 Grafik Karakteristik Kecepatan Drone B

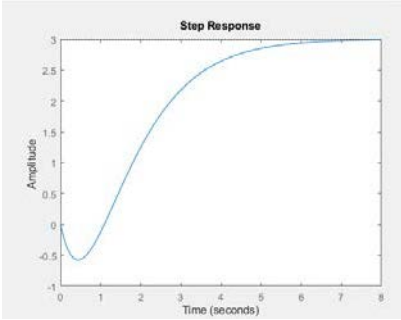
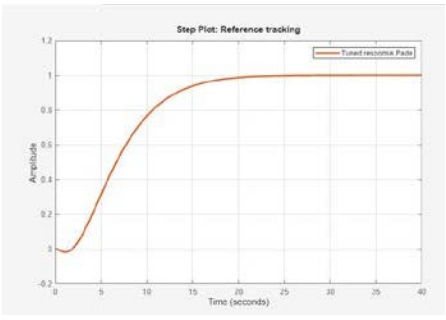
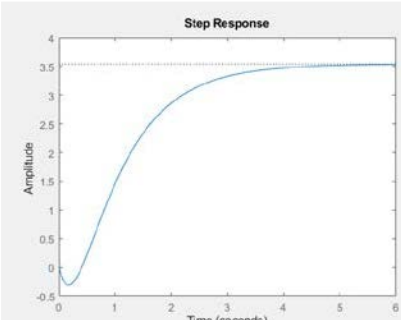
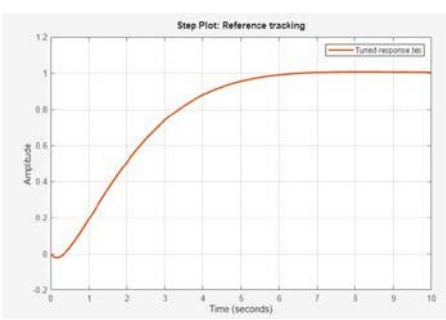
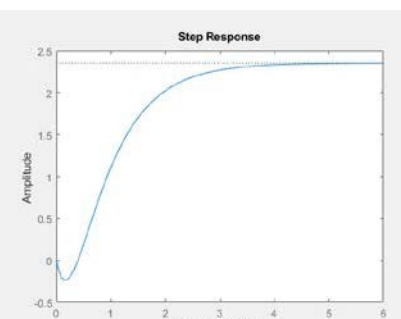
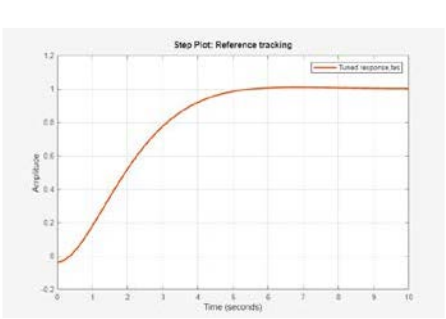
Berdasarkan data yang diperoleh dari identifikasi kecepatan *drone* akan didapatkan parameter fungsi transfer orde satu atau FOPTD. Dengan menggunakan fungsi transfer tersebut, dapat ditentukan karakteristik dan pengontrol yang dapat digunakan pada penelitian ini. Berikut grafik karakteristik untuk *drone* B.

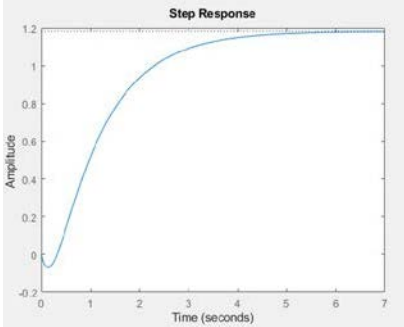
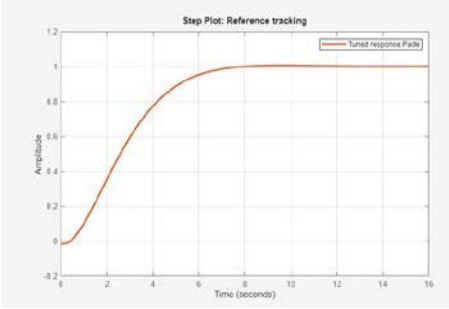
Arah Pergerakan	Grafik Karakteristik
Maju	<div><p>Kecepatan Drone B Arah Maju</p></div>
Mundur	<div><p>Kecepatan Drone B Arah Mundur</p></div>



C.3 Grafik Pengontrol PD untuk Drone A

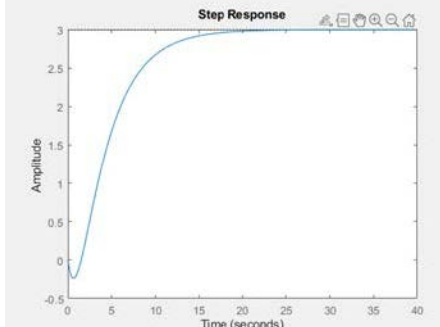
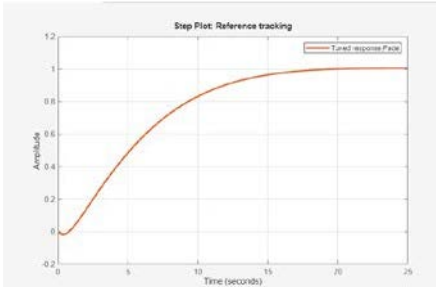
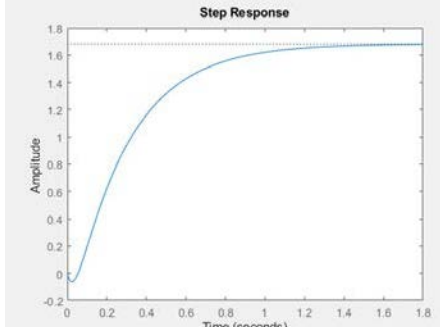
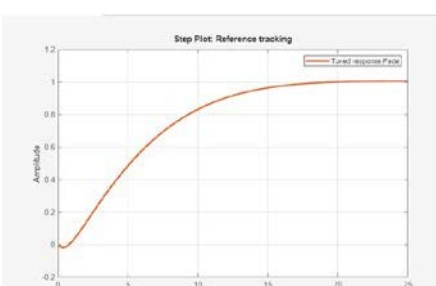
Berdasarkan data yang diperoleh dari identifikasi kecepatan *drone* akan dan parameter fungsi transfer, akan dilakukan perancangan pengontrol PD untuk masing-masing *drone*. Berikut merupakan diagram sistem setelah menggunakan pengontrol PD pada *drone* A.

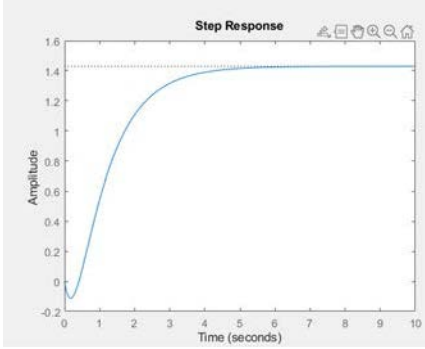
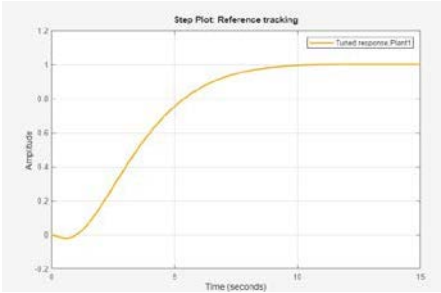
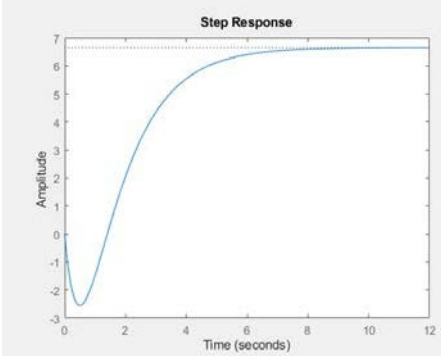

Arah	Respons Sistem Awal	Respons Sistem Menggunakan Pengontrol PD
Maju		
Mundur		
Kiri		

Arah	Respons Sistem Awal	Respons Sistem Menggunakan Pengontrol PD
Kanan		

C.4 Grafik Pengontrol PD untuk Drone B

Berdasarkan data yang diperoleh dari identifikasi kecepatan *drone* akan dan parameter fungsi transfer, akan dilakukan perancangan pengontrol PD untuk masing-masing *drone*. Berikut merupakan diagram sistem setelah menggunakan pengontrol PD pada *drone* B.

Arah	Respons Sistem Awal	Respons Sistem Menggunakan Pengontrol PD
Maju		
Mundur		

Arah	Respons Sistem Awal	Respons Sistem Menggunakan Pengontrol PD
Kanan		
Kiri		

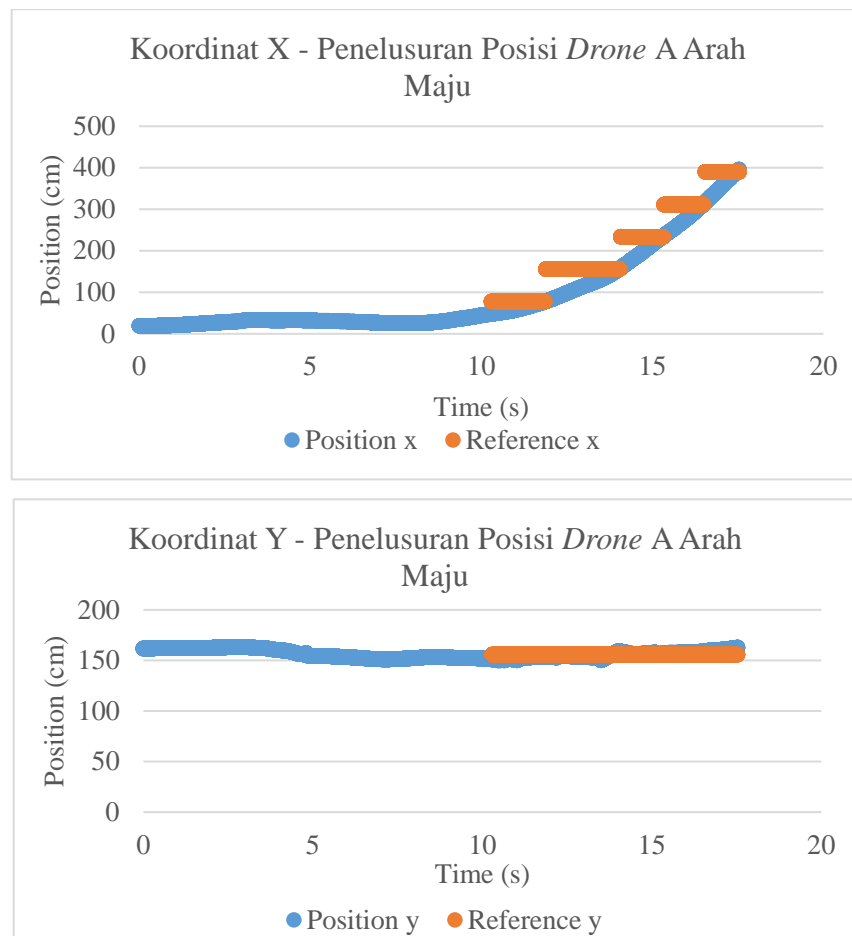
LAMPIRAN D. PENELUSURAN POSISI *DRONE*

Penelusuran posisi *drone* dilakukan dengan memberikan titik referensi dengan pergerakan ke arah maju, kanan, dan kiri. Penelusuran posisi dilakukan untuk mengetahui pergerakan *drone* sesuai jalur yang diberikan. Pengujian ini dilakukan pada tiap *drone* dan dilihat dari respons pergerakan *drone*. Di bawah ini merupakan hasil penelusuran posisi untuk *drone* A dan *drone* B. Data keseluruhan dapat diakses melalui bit.ly/PenelusuranPosisiQuadrotor.

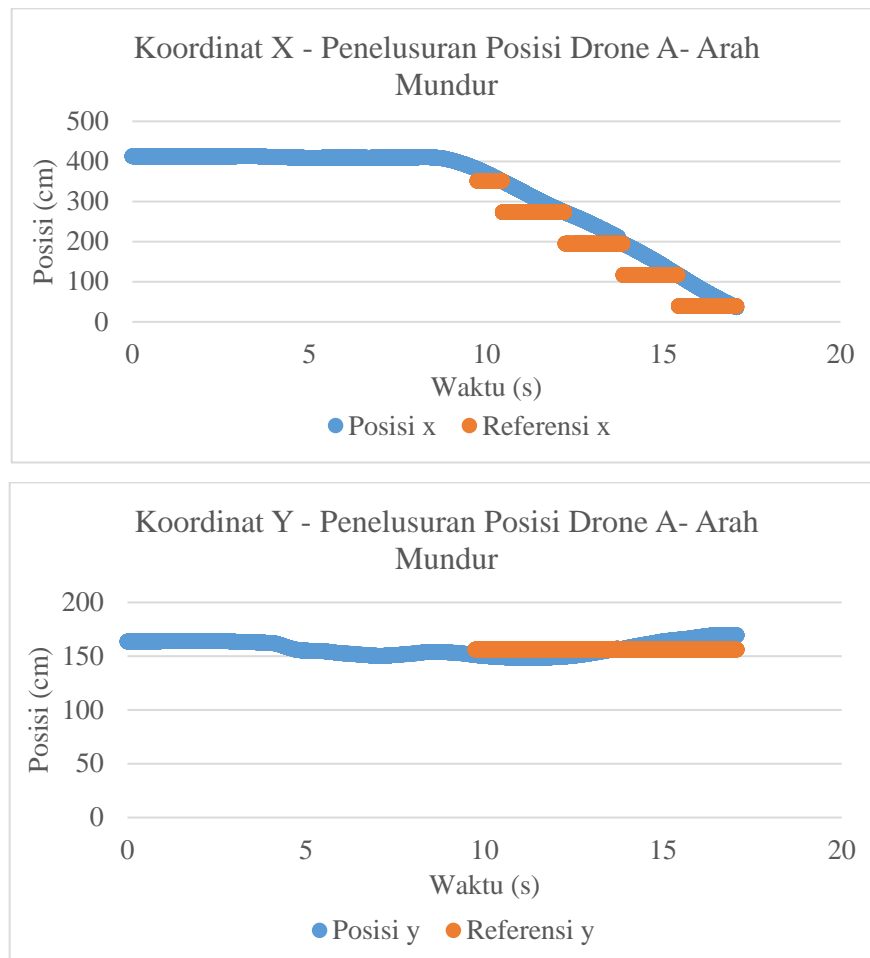
D.1 Penelusuran Posisi *Drone* A

Penelusuran posisi pada *drone* A dilakukan pada koordinat *cartesius* dengan menempatkan *drone* sejajar dengan sumbu x. Selain itu, *drone* akan bergerak maju ke arah sumbu x positif, sehingga ketika *drone* maju, maka nilai koordinat x akan bertambah. Pergerakan kanan dan kiri akan dilakukan ke arah sumbu y. Berikut merupakan penelusuran posisi untuk *drone* A.

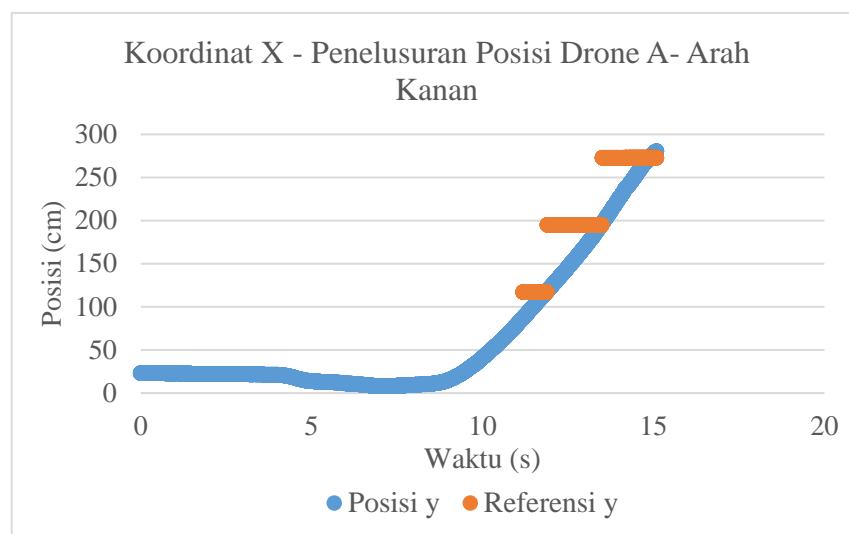
1. Pergerakan Arah Maju

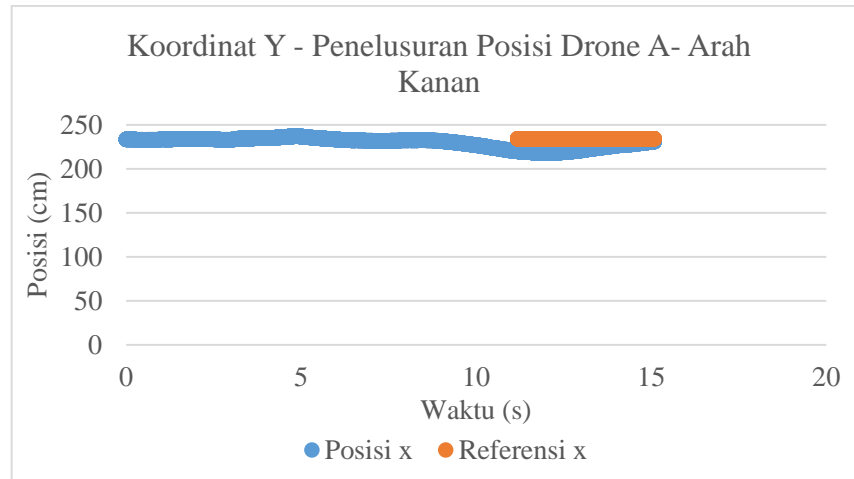


2. Pergerakan Arah Mundur

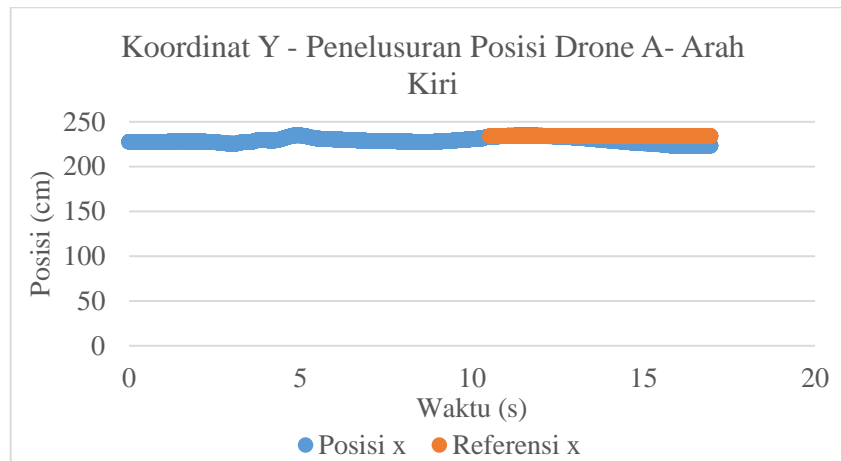
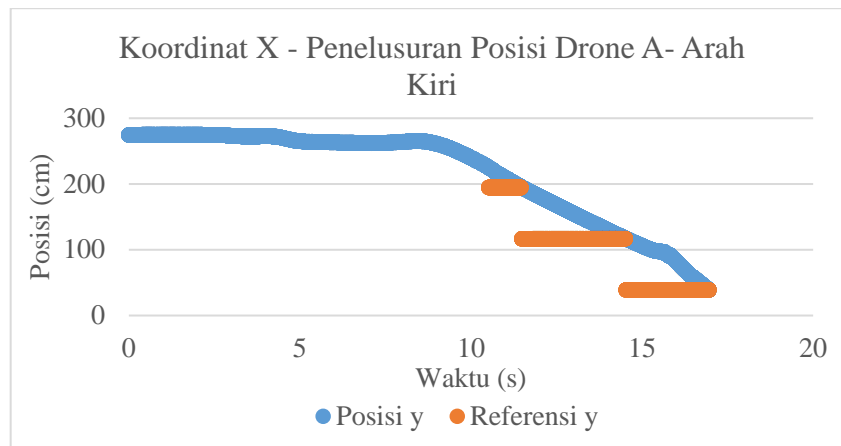


3. Pergerakan Arah Kanan





4. Pergerakan Arah Kiri

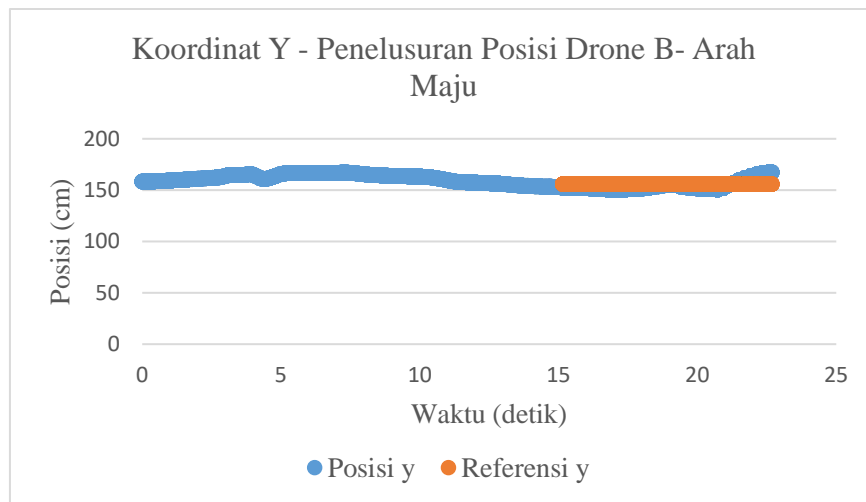
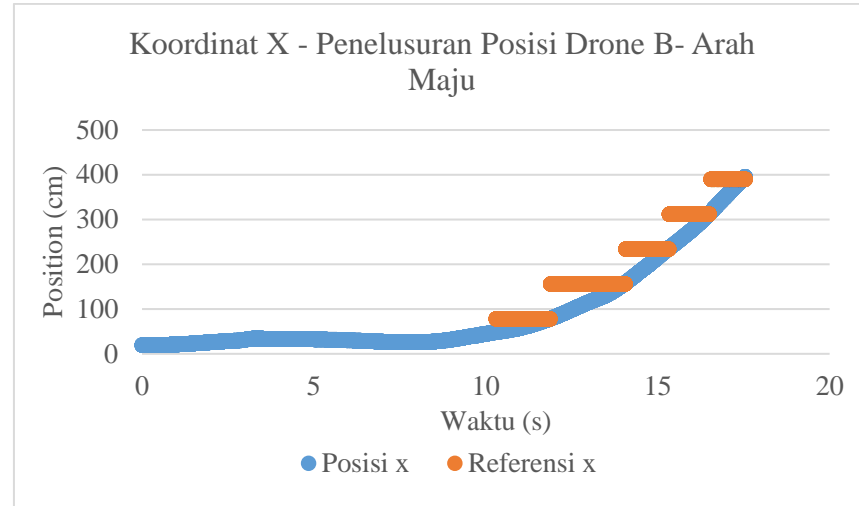


D.2 Penelusuran Posisi Drone B

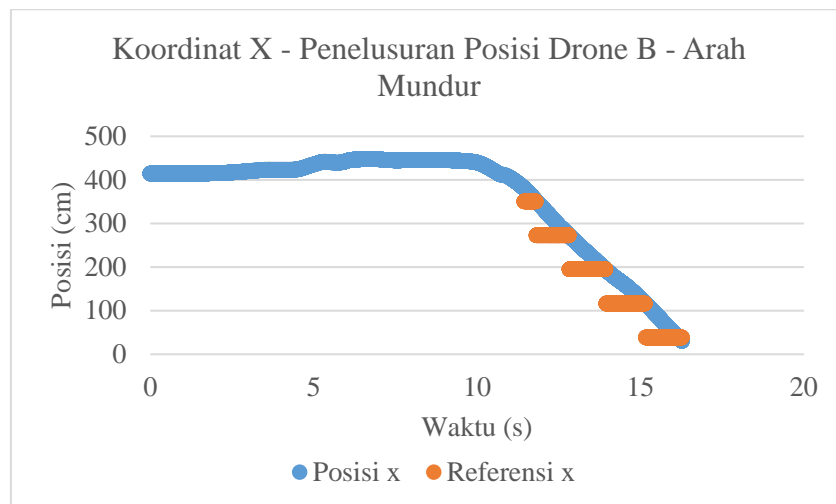
Penelusuran posisi pada *drone* B dilakukan pada koordinat *cartesius* dengan menempatkan *drone* sejajar dengan sumbu x. Selain itu, *drone* akan bergerak maju ke arah sumbu x positif, sehingga ketika *drone* maju, maka nilai koordinat x akan

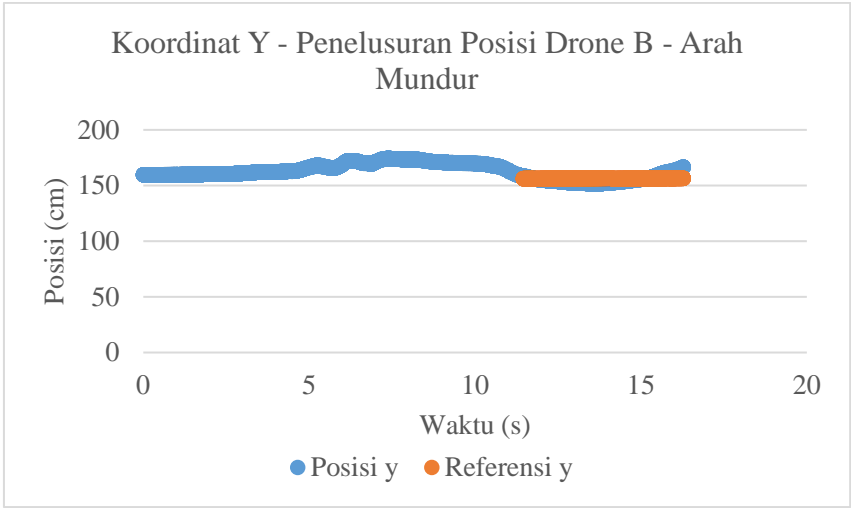
bertambah. Pergerakan kanan dan kiri akan dilakukan ke arah sumbu y. Berikut merupakan penelusuran posisi untuk *drone* B.

1. Pergerakan Arah Maju

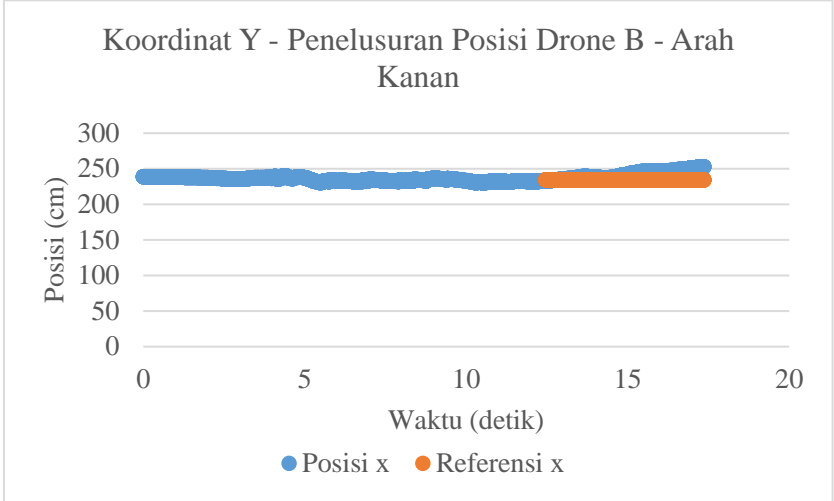
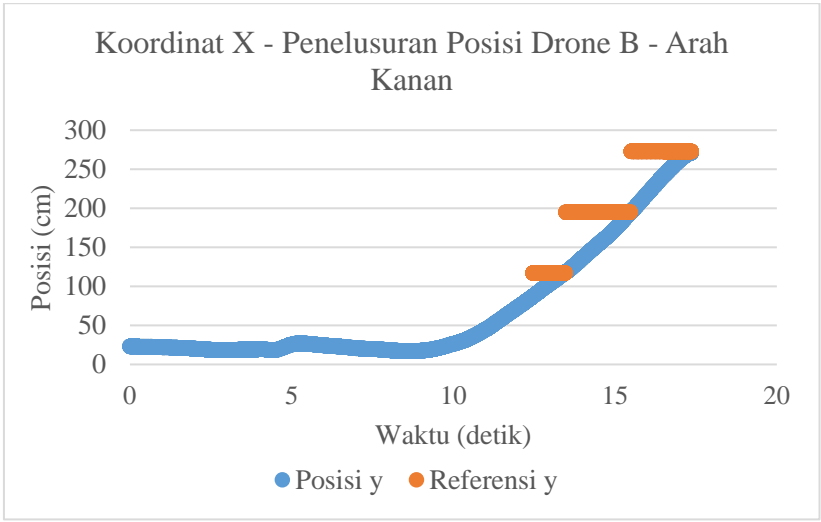


2. Pergerakan Arah Mundur





3. Pergerakan Arah Kanan



4. Pergerakan Arah Kiri

