



TUGAS AKHIR (AE43250)

**IMPLEMENTASI *OBSTACLE AVOIDANCE* AUV BERBASIS
STEREO DEPTH DAN APF PADA KRBAI**

Syahrizal Aditya Kuncoro
NRP. 0922040096

Dosen Pembimbing
Ryan Yudha Adhitya, S.ST., M.T.
Zindhu Maulana Ahmad Putra, S.ST., M.Tr.T.

PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK KELISTRIKAN KAPAL
POLITEKNIK PERKAPALAN NEGERI SURABAYA
SURABAYA
2026

(Halaman ini sengaja dikosongkan)



PPNS POLITEKNIK
PERKAPALAN
NEGERI SURABAYA

TUGAS AKHIR (AE43250)

IMPLEMENTASI *OBSTACLE AVOIDANCE* AUV BERBASIS *STEREO DEPTH* DAN APF PADA KRBAI

Syahrizal Aditya Kuncoro
NRP. 0922040096

Dosen Pembimbing
Ryan Yudha Adhitya, S.ST., M.T.
Zindhu Maulana Ahmad Putra, S.ST., M.Tr.T.

PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK KELISTRIKAN KAPAL
POLITEKNIK PERKAPALAN NEGERI SURABAYA
SURABAYA
2026

(Halaman ini sengaja dikosongkan)

PPNS

PPNS

LEMBAR PENGESAHAN

PPNS

PPNS

TUGAS AKHIR

IMPLEMENTASI *OBSTACLE AVOIDANCE* AUV BERBASIS *STEREO DEPTH* DAN *APF* PADA *KRBAI*

PPNS

PPNS

PPNS

PPNS

PPNS

PPNS

Disusun Oleh:

Syahrizal Aditya Kuncoro

0922040096

Diajukan Untuk Memenuhi Salah Satu Syarat Kelulusan
Program Studi D4 Teknik Otomasi
Jurusan Teknik Kelistrikan Kapal
POLITEKNIK PERKAPALAN NEGERI SURABAYA

PPNS

PPNS

PPNS

PPNS

PPNS

PPNS

Disetujui oleh Tim penguji Tugas Akhir Tanggal Ujian
Periode Wisuda

: 22 Juli 2026

: Oktober 2026

Menyetujui,

PPNS

PPNS

PPNS

PPNS

PPNS

PPNS

Dosen Penguji

1. Ir. Joko Endrasmono, M.T.
2. Noorman Rinanto, S.T., M.T., Ph.D.
3. Zindhu Maulana Ahmad Putra, S.ST., M.Tr.T.

NIDN

(0009096402)

(0014107607)

(0023099403)

Tanda Tangan

(.....)

(.....)

(.....)

Dosen Pembimbing

1. Ryan Yudha Adhitya, S.ST., M.T.
2. Zindhu Maulana Ahmad Putra, S.ST., M.Tr.T.

NIDN

(0016069101)

(0023099403)

Tanda Tangan

(.....)

(.....)

Menyetujui
Ketua Jurusan,



Isa Rachman, S.T., M.T.
NIP. 198008162008121001

Mengetahui
Koordinator Program Studi,

Agus Khumaidi, S.ST., M.T.
NIP. 199308172020121004

(Halaman ini sengaja dikosongkan)

 PPNS	PERNYATAAN BEBAS PLAGIAT	No. : F. WD I. 021 Date : 3 Nopember 2015 Rev. : 01 Page : 1 dari 1
--	-------------------------------------	--

Yang bertanda tangan di bawah ini:

Nama : Syahrizal Aditya Kuncoro

NRP : 0922040096

Jurusan/Prodi : Teknik Kelistrikan Kapal / D4 Teknik Otomasi

Dengan ini menyatakan dengan sesungguhnya bahwa :

Tugas Akhir yang akan saya kerjakan dengan judul :

IMPLEMENTASI *OBSTACLE AVOIDANCE* AUV BERBASIS *STEREO DEPTH* DAN APF PADA KRBAI

Adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain.

Apabila dikemudian hari terbukti terdapat plagiat dalam karya ilmiah tersebut, maka saya bersedia menerima sanksi sesuai ketentuan peraturan yang berlaku.

Demikian surat pernyataan ini saya buat dengan penuh tanggung jawab.

Surabaya, 5 Agustus 2026
Yang membuat pernyataan,



(Syahrizal Aditya Kuncoro)
NRP. 0922040096

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR



Dengan mengucapkan syukur Alhamdulillah, segala puji penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul **“Implementasi Obstacle Avoidance AUV Berbasis Stereo Depth dan APF pada KRBAI”** dengan baik dan lancar. Sholawat serta salam senantiasa tercurah kepada junjungan Nabi Muhammad SAW, yang telah membimbing umat manusia dari zaman kegelapan menuju era yang terang benderang penuh ilmu pengetahuan.

Tugas Akhir ini disusun sebagai salah satu syarat untuk menyelesaikan studi pada Program Sarjana Terapan D-IV Teknik Otomasi di Politeknik Perkapalan Negeri Surabaya serta untuk memperoleh gelar Sarjana Terapan Teknik Otomasi. Penulis menyadari sepenuhnya bahwa tersusunnya laporan ini tidak lepas dari dukungan, bimbingan, serta doa dari berbagai pihak yang telah memberikan kontribusi baik secara langsung maupun tidak langsung. Oleh karena itu, penulis ingin menyampaikan banyak rasa terima kasih dan penghargaan yang sebesar-besarnya khususnya kepada :

1. Allah SWT, karena berkat limpahan nikmat sehat, nikmat ilmu, nikmat waktu, serta limpahan rahmat dan hidayah-Nya sehingga Penulis dapat menyelesaikan Tugas Akhir ini tepat waktu.
2. Orang tua penulis yaitu Bapak Rudy Kuncoro dan Ibu Yeni Evita Kristanti yang selalu mendoakan, memberikan dukungan dan motivasi, support penuh ketika suka dan duka, serta mengingatkan penulis untuk tetap semangat dalam mengerjakan tugas akhir ini hingga selesai.
3. Bapak Rachmad Tri Soelistijono, S.T., M.T. yaitu selaku Direktur Politeknik Perkapalan Negeri Surabaya.
4. Bapak Isa Rachman, S.T., M.T. selaku Ketua Jurusan Teknik Kelistrikan Kapal.
5. Bapak Agus Khumaidi, S.ST., M.T. selaku Koordinator Program Studi Teknik Otomasi Politeknik Perkapalan Negeri Surabaya.

6. Bapak Ryan Yudha Adhitya, S.ST., M.T. selaku Dosen Pembimbing I yang dengan penuh kesabaran telah memberikan motivasi, arahan, serta bimbingan kepada penulis selama proses penyusunan Tugas Akhir ini. Berbagai masukan dan ilmu yang diberikan menjadi bekal yang sangat berarti sehingga penulis dapat menyelesaikan Tugas Akhir ini.
7. Bapak Zindhu Maulana Ahmad Putra, S.ST., M.Tr.T. selaku Dosen Pembimbing II, penulis mengucapkan terima kasih yang sebesar-besarnya atas bimbingan dan dukungan yang telah diberikan selama 3 tahun ini. Penulis mendapat banyak sekali ilmu serta pengalaman dari beliau.
8. Terima kasih yang sebesar-besarnya kepada para dosen penguji. Kritikan dan saran yang diberikan sangat berarti, telah membuka wawasan saya dan mendorong perbaikan signifikan dalam penulisan laporan ini.
9. Tak lupa, kepada Tim Robot AEROships dan HYDROships, tempat penulis bernaung selama tiga tahun. Terima kasih atas setiap momen suka dan duka, setiap tantangan yang penulis dan tim hadapi bersama, dan setiap dukungan yang tak pernah putus. Tim ini bukan hanya sekadar kelompok kerja, melainkan inspirasi dan motivasi yang tak pernah habis bagi saya dalam berkarya.
10. Semua pihak lain yang tidak dapat disebutkan satu persatu. Terimakasih banyak atas semua bantuan dan semangat yang diberikan bagi penulis.

Semoga Allah SWT memberikan balasan yang berlipat ganda kepada semuanya. Tugas Akhir ini masih sangat jauh dari kata sempurna, kritik dan saran yang dapat menyempurnakan penyusunan Tugas Akhir ini sangat diperlukan. Semoga Tugas Akhir ini bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi.

Penulis,

(Syahrizal Aditya Kuncoro)

Implementasi *Obstacle Avoidance* AUV Berbasis *Stereo Depth* dan APF Pada KRBAI

Syahrizal Aditya Kuncoro

ABSTRAK

Autonomous Underwater Vehicle (AUV) pada Kontes Robot Bawah Air Indonesia (KRBAI) dituntut menyelesaikan lintasan secara mandiri tanpa intervensi operator. Tantangannya adalah memadukan persepsi lingkungan dan perencanaan gerak menjadi satu sistem navigasi yang bekerja secara waktu nyata pada perangkat dengan sumber daya terbatas. Penelitian ini merancang sistem *obstacle avoidance* yang menggabungkan deteksi objek You Only Look Once (YOLO), estimasi jarak berbasis *stereo depth*, dan perencanaan gerak Artificial Potential Field (APF) pada *middleware* Robot Operating System 2 (ROS2). Deteksi YOLO mengenali gawang, suar, dan bak dengan mAP@50 sebesar 0,990. Estimasi jarak dikoreksi terhadap refraksi air, dan rasio faktor skala air terhadap darat sebesar 1,362 mendekati indeks bias air sehingga penyimpangan terbukti berasal dari pembiasan cahaya. Setelah koreksi, galat rata-rata estimasi jarak di bawah 5% pada rentang 0,4–2,8 m. Pengujian arah gaya tolak semu terhadap rintangan tunggal, celah gawang, dan rintangan mendadak menghasilkan kesesuaian arah 100% pada dua belas percobaan. Pada pengujian misi menyerupai arena KRBAI, sistem tanpa APF menumbuk tiang gawang pada tiga dari sepuluh percobaan, sedangkan sistem dengan APF menyelesaikan seluruh percobaan tanpa tumbukan dengan durasi setara. Hasil ini menunjukkan integrasi persepsi visual, estimasi kedalaman, dan perencanaan APF berbasis ROS2 mampu mewujudkan navigasi otonom AUV yang andal pada lintasan terstruktur.

Kata kunci: AUV, *obstacle avoidance*, YOLO, *stereo depth*, Artificial Potential Field.

(Halaman ini sengaja dikosongkan)

Implementation of Stereo Depth and APF – Based AUV Obstacle Avoidance in KRBAI

Syahrizal Aditya Kuncoro

ABSTRACT

Autonomous Underwater Vehicles (AUVs) in the Indonesian Underwater Robot Contest (KRBAI) must complete a course autonomously without operator intervention. The challenge is integrating perception and motion planning into one navigation system running in real time on resource-constrained hardware. This research implements an obstacle avoidance system combining You Only Look Once (YOLO) object detection, stereo depth based distance estimation, and Artificial Potential Field (APF) motion planning on Robot Operating System 2 (ROS2) middleware. YOLO detects the gate, buoy, and bin with a mAP@50 of 0.990. Distance estimation is corrected for water refraction, and the ratio of the underwater to the in-air scale factor, 1.362, approaches the refractive index of water, confirming refraction as the source of the deviation. After correction, the mean distance error is below 5% across the 0.4–2.8 m range. Directional tests against a single obstacle, a gate opening, and a sudden obstacle yield 100% accuracy over twelve trials. In mission testing resembling the KRBAI arena, the system without APF collided with gate posts in three of ten trials, whereas the system with APF completed all ten without collision at an equivalent time. Integrating visual perception, depth estimation, and ROS2 based APF planning therefore achieves reliable AUV navigation.

Keywords: AUV, obstacle avoidance, YOLO, stereo depth, Artificial Potential Field.

(Halaman ini sengaja dikosongkan)

DAFTAR ISI

LEMBAR PENGESAHAN.....	iii
PERNYATAAN BEBAS PLAGIAT	v
KATA PENGANTAR	vii
ABSTRAK.....	ix
ABSTRACT	xi
DAFTAR ISI.....	xiii
DAFTAR TABEL	xix
DAFTAR GAMBAR.....	xxi
DAFTAR NOTASI.....	xxv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Pertanyaan Penelitian.....	4
1.3 Batasan Penelitian	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	6
BAB 2 TINJAUAN PUSTAKA	9
2.1 Kajian Penelitian Terdahulu	9
2.2 Autonomous Underwater Vehicle.....	12
2.3 Navigasi dan <i>Obstacle Avoidance</i>	14
2.4 Kendali <i>Proportional Integral Derivative</i> (PID)	16
2.5 Persepsi Visual Robot Bawah Air.....	18
2.6 <i>Robot Operating System 2</i> (ROS2).....	20
2.7 Pengolahan Citra Digital.....	22

2.8	You Only Look Once (YOLO).....	23
2.9	<i>Stereo Depth</i>	25
2.10	Semi-Global Block Matching (SGBM)	29
2.11	Perencanaan Lintasan Lokal	30
2.12	Artificial Potential Field	32
2.10.1	Fungsi Medan Potensial Tarik Menarik	32
2.10.2	Fungsi Medan Potensial Tolak-menolak	33
2.10.3	Fungsi Medan Potensial Resultan.....	33
2.10.4	Masalah <i>Local Minimum</i>	35
2.13	Sensor dan Akuator.....	37
2.11.1	Sensor <i>Pressure</i>	37
2.11.2	Sensor IMU	38
2.11.3	<i>Magnetic Switch</i>	39
2.11.4	Kamera DWE <i>exploreHD</i>	40
2.11.5	Motor Thruster BLDC	41
2.14	Kontroller.....	42
2.13.1	NVIDIA Jetson Orin NX.....	42
2.13.2	ESP32	43
2.13.3	Electronic Speed Controller.....	44
2.15	Baterai <i>Li-ion</i>	45
2.16	Gazebo	46
2.17	Komunikasi	47
BAB 3 METODE PENELITIAN		49
3.1	Alur dan Tahapan Penelitian.....	49
3.2	Konsep dan Arsitektur Sistem AUV.....	51
3.2.1	Konsep Penelitian	51

3.2.2	Diagram Blok Sistem	52
3.2.3	Arsitektur Sistem Berbasis ROS2.....	55
3.3	Sistem Persepsi dan Pemodelan Lingkungan.....	58
3.3.1	Akuisisi Citra Stereo (Kamera DWE exploreHD).....	58
3.3.2	Kalibrasi Stereo dan Kalibrasi Bawah Air.....	60
3.3.3	Rektifikasi dan Komputasi <i>Depth Map</i> (SGBM)	65
3.3.4	Deteksi Objek Berbasis YOLO	69
3.3.5	Penggabungan Deteksi Objek dengan kedalaman	72
3.3.6	Representasi Lingkungan Lokal Berbasis Zona Kedalaman Lingkungan AUV	74
3.3.7	Alur Kerja Node Persepsi	76
3.4	Perencanaan Lintasan dan <i>Obstacle Avoidance</i>	79
3.4.1	Perencanaan Lintasan Lokal.....	79
3.4.2	<i>Obstacle Avoidance</i> Menggunakan Artificial Potential Field (APF) .	80
3.5	Sistem Kendali dan Implementasi	85
3.5.1	Sistem Kendali AUV	85
3.5.2	Komunikasi Jetson Orin NX dan ESP32	87
3.5.3	Alur Mesin Keadaan <i>Mission Manager</i>	88
3.6	Perancangan Sistem	91
3.6.1	Perancangan Sistem Mekanik.....	91
3.6.2	Perancangan Sistem Elektrik.....	94
3.6.3	Flowchart Sistem Keseluruhan.....	95
BAB 4 HASIL DAN PEMBAHASAN		99
4.1	Hasil Perancangan Hardware.....	99
4.1.1	Hasil Perancangan Mekanik	99
4.1.2	Hasil Perancangan Sistem Elektrik.....	99

4.1.3	Hasil Perancangan ROS 2 Humble.....	100
4.1.4	Spesifikasi dan Kemampuan AUV.....	103
4.2	Pengujian Sensor dan Aktuator.....	104
4.2.1	Pengujian Sensor MS5837 <i>Pressure</i>	105
4.2.2	Pengujian Sensor HWT905	106
4.2.3	Pengujian Motor <i>Thruster</i>	110
4.2.4	Pengujian Gerak Dasar dan Kecepatan AUV.....	115
4.2.5	Pengujian Komunikasi Jetson Orin NX dan ESP32	118
4.3	Pengujian Kamera Stereo Menggunakan 2 Kamera DWE exploreHD	121
4.3.1	Pengetesan Kamera DWE exploreHD.....	121
4.3.2	Proses Kalibrasi Kamera DWE exploreHD.....	122
4.3.3	Hasil Depth Map SGBM	124
4.4	Hasil Pembahasan YOLO	126
4.4.2	Pra-pemrosesan Data	127
4.4.3	Pelatihan Model YOLO.....	129
4.5	Hasil Estimasi Jarak Objek Berbasis Stereo Vision.....	139
4.5.3	Pengujian Jarak Objek Gawang.....	140
4.5.4	Pengujian Jarak Objek Suar.....	142
4.5.5	Pengujian Jarak Objek Bak	143
4.6	Pengujian Sistem Obstacle Avoidance pada Simulasi.....	143
4.6.1	Konfigurasi Lingkungan Simulasi.....	144
4.6.2	Hasil Pengujian Skenario A – Dengan <i>Waypoint</i> AVOID_SUAR ...	147
4.6.3	Hasil Pengujian Skenario B – Tanpa <i>Waypoint</i> AVOID_SUAR	148
4.7	Pengujian Sistem <i>Obstacle Avoidance</i> pada Kolam Nyata	150
4.7.1	Konfigurasi Pengujian Kolam	150

4.7.2	Skenario Pengujian	153
4.7.3	Pengujian Respons Perintah Geser terhadap Satu Objek Diam.....	154
4.7.4	Pengujian Respons Perintah Geser terhadap Dua Objek Diam	156
4.7.5	Pengujian Respons Perintah Geser terhadap Rintangan yang Muncul Mendadak	158
4.7.6	Hasil Pengujian Misi dengan APF Nonaktif	160
4.7.7	Hasil Pengujian Misi dengan APF Aktif.....	162
4.7.8	Perbandingan Kinerja APF Aktif dan Nonaktif	164
BAB 5	KESIMPULAN.....	167
5.1	Kesimpulan	167
5.2	Saran	168
DAFTAR PUSTAKA		171
LAMPIRAN		179

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 2. 1	Perubahan dan Perkembangan dari penelitian sebelumnya	10
Tabel 2. 2	Pengaruh Peningkatan Konstanta PID terhadap Tanggapan Sistem	17
Tabel 2. 3	Spesifikasi Sensor Pressure MS5837	38
Tabel 2. 4	Spesifikasi Sensor IMU Witmotion HWT905	39
Tabel 2. 5	Spesifikasi Magnetic Switch MC-38	40
Tabel 2. 6	Spesifikasi Motor Thruster T200	42
Tabel 2. 7	Spesifikasi Jetson Orin NX 16GB	43
Tabel 2. 8	Spesifikasi ESP32	44
Tabel 2. 9	Spesifikasi Electronic Speed Controller (ESC)	45
Tabel 2. 10	Spesifikasi Baterai Li-ion Sony VCT6	46
Tabel 3. 1	Daftar Node ROS2 dan Fungsinya pada Sistem AUV KRBAI	57
Tabel 3. 2	Parameter Kalibrasi Stereo dan Koreksi Jarak Bawah Air	64
Tabel 3. 3	Parameter Algoritma Semi-Global Block Matching (SGBM)	66
Tabel 3. 4	Parameter Algoritma Artificial Potential Field (APF)	84
Tabel 3. 5	Parameter Pengendali PID pada Setiap Sumbu Kendali	87
Tabel 4. 1	Daftar Node ROS2 yang Aktif pada Sistem AUV KRBAI	102
Tabel 4. 2	Spesifikasi dan Kemampuan Autonomous Underwater Vehicle	103
Tabel 4. 3	Hasil Pengujian Sensor Pressure MS5837	105
Tabel 4. 4	Pengujian Sumbu X	108
Tabel 4. 5	Pengujian Sumbu Y	109
Tabel 4. 6	Pengujian Sumbu Z	109
Tabel 4. 7	Hasil Analisis Sinyal PWM Menggunakan Osiloskop	113
Tabel 4. 8	Hasil Pengujian Kecepatan Gerak AUV	116
Tabel 4. 9	Hasil Pengukuran Laju Data Umpan Balik Jetson–ESP32	119
Tabel 4. 10	Hasil kalibrasi kamera stereo DWE exploreHD	123
Tabel 4. 11	Dataset Gawang, Suar, dan Bola	127
Tabel 4. 12	Pembagian Jumlah Dataset	129
Tabel 4. 13	Hyperparameter Pelatihan Model YOLO	129
Tabel 4. 14	Pembacaan Nilai Berdasarkan Confussion Matrix	131
Tabel 4. 15	Rekapitulasi Hasil Evaluasi Model YOLOv8n	133

Tabel 4. 16	Data Pembacaan Jarak dari Kamera Stereo (Sebelum Kalibrasi) ...	139
Tabel 4. 17	Hasil Pengujian Deteksi Gawang	140
Tabel 4. 18	Hasil Pengujian Deteksi Suar	142
Tabel 4. 19	Hasil Pengujian Deteksi Bak	143
Tabel 4. 20	Hasil Pengujian Skenario A (dengan AVOID_SUAR)	148
Tabel 4. 21	Hasil Pengujian Skenario B (tanpa AVOID_SUAR)	149
Tabel 4. 22	Parameter Final Sistem pada Pengujian Kolam Nyata	153
Tabel 4. 23	Hasil Pengujian Respons Perintah Geser terhadap Satu Objek Diam	155
Tabel 4. 24	Hasil Pengujian Respons Perintah Geser terhadap Dua Objek Diam	157
Tabel 4. 25	Hasil Pengujian Arah Respons Perintah Geser terhadap Rintangan yang Muncul Mendadak.....	159
Tabel 4. 26	Hasil Pengujian Misi dengan APF Nonaktif	161
Tabel 4. 27	Hasil Pengujian Misi dengan APF Aktif	163
Tabel 4. 28	Rangkuman perbandingan APF aktif dan nonaktif.....	165

DAFTAR GAMBAR

Gambar 2. 1 Autonomous Underwater Vehicle HYDROships	14
Gambar 2. 2 Contoh skenario AUV menghadapi rintangan pada lingkungan bawah air.....	16
Gambar 2. 3 Contoh visualisasi hasil persepsi lingkungan AUV terhadap rintangan dan target	20
Gambar 2. 4 Logo ROS2	20
Gambar 2. 5 Menampilkan nilai dari setiap pixel pada gambar	22
Gambar 2. 6 Arsitektur Umum Jaringan YOLO.....	23
Gambar 2. 7 Ilustrasi prinsip kerja stereo vision dalam estimasi jarak objek atau <i>stereo depth</i>	28
Gambar 2. 8 Perbandingan lintasan global dan lintasan lokal pada robot otonom	31
Gambar 2. 9 Model gaya total AUV	35
Gambar 2. 10 Masalah local minimum dengan tiga titik segaris pada metode APF	36
Gambar 2. 11 Masalah local minimum dengan tiga titik segaris.....	36
Gambar 2. 12 Daerah local minimum dengan banyak rintangan.....	36
Gambar 2. 13 Sensor Pressure Rovmaker MS5837.....	38
Gambar 2. 14 Sensor IMU Witmotion HWT905	39
Gambar 2. 15 Magnetic Switch MC-38.....	40
Gambar 2. 16 Kamera DWE exploreHD	41
Gambar 2. 17 Motor Thruster T200.....	41
Gambar 2. 18 Jetson Orin NX 16 GB.....	43
Gambar 2. 19 ESP32.....	44
Gambar 2. 20 Electronic Speed Controller (ESC)	45
Gambar 2. 21 Baterai Li-ion Sony VCT6 18650.....	46
Gambar 2. 22 Logo Gazebo	47
Gambar 3. 1 Flowchart Tahapan Penelitian.....	50
Gambar 3. 2 Blok Diagram Sistem Autonomous Underwater Vehicle	52
Gambar 3. 3 Diagram node ROS2 pada AUV	55

Gambar 3. 4 Pemasangan Dua Kamera DWE exploreHD sebagai Stereo Pair pada AUV	59
Gambar 3. 5 Contoh Pasangan Citra Kamera Kiri dan Kanan DWE exploreHD	60
Gambar 3. 6 Proses Kalibrasi Stereo Menggunakan Papan Catur Tampak Kiri dan Kanan (Checkerboard).....	61
Gambar 3. 7 Proses Kalibrasi Koreksi Jarak Bawah Air Menggunakan Objek Referensi (bak) dengan Pembacaan Jarak Melalui Deteksi YOLO.....	62
Gambar 3. 8 Grafik Hasil Kalibrasi Koreksi Jarak Bawah Air.....	63
Gambar 3. 9 Verifikasi Rektifikasi Stereo pada Pasangan Kamera (garis hijau menunjukkan garis epipolar horizontal).....	65
Gambar 3. 10 Hasil Komputasi Depth Map Menggunakan Algoritma SGBM	67
Gambar 3. 11 Pembagian Zona Deteksi Rintangan pada Depth Map	68
Gambar 3. 12 Contoh Citra Dataset dan Proses Labeling Tiga Kelas Objek	71
Gambar 3. 13 Hasil Deteksi YOLO terhadap Tiga Objek Target Misi	72
Gambar 3. 14 Hasil Penggabungan Deteksi YOLO dengan Informasi Kedalaman	74
Gambar 3. 15 Ilustrasi Representasi Lingkungan Lokal Berbasis Zona Kedalaman AUV pada Kerangka Artificial Potential Field (APF)	75
Gambar 3. 16 Alur Kerja Node Persepsi pada Sistem AUV	77
Gambar 3. 17 Ilustrasi perbedaan perencanaan lintasan global dan perencanaan lintasan lokal pada sistem navigasi AUV	80
Gambar 3. 18 Visualisasi medan gaya Artificial Potential Field (APF).....	82
Gambar 3. 19 Diagram alir algoritma Artificial Potential Field (APF).....	83
Gambar 3. 20 Diagram komunikasi antara Jetson Orin NX dan ESP32 pada sistem AUV	88
Gambar 3. 21 Diagram Alir Mesin Keadaan Mission Manager	89
Gambar 3. 22 Desain AUV tampak Isometri.....	91
Gambar 3. 23 Desain AUV tampak atas.....	92
Gambar 3. 24 Desain AUV tampak belakang.....	92
Gambar 3. 25 Desain AUV tampak depan.....	93
Gambar 3. 26 Desain AUV tampak samping	93
Gambar 3. 27 Diagram Blok pada komponen elektrik	94

Gambar 3. 28 Flowchart Sistem Obstacle Avoidance AUV.....	96
Gambar 4. 1 Tampak depan Robot	99
Gambar 4. 2 Papan PCB AUV.....	100
Gambar 4. 3 Visualisasi Graf Node ROS2 menggunakan rqt_graph.....	101
Gambar 4. 4 Pengujian Sensor Pressure MS5837	105
Gambar 4. 5 Pengujian Nilai Sensor HWT905 dengan Busur.....	107
Gambar 4. 6 Software WitMotion	108
Gambar 4. 7 Spesifikasi Motor Thruster BLDC ROVMAKER T200.....	110
Gambar 4. 8 Sinyal PWM Motor Thruster 1100 μ s.....	111
Gambar 4. 9 Sinyal PWM Motor Thruster 1500 μ s.....	112
Gambar 4. 10 Sinyal PWM Motor Thruster 1900 μ s	112
Gambar 4. 11 Pengujian Motor Thruster Menggunakan Kontroler ESP32.....	113
Gambar 4. 12 Grafik Hasil Pembacaan Tachometer pada Thruster	114
Gambar 4. 13 Proses Pengujian Gerak Dasar AUV pada Kolam Renang.....	116
Gambar 4. 14 Pengukuran Laju Data Umpan Balik pada Topik /serial/feedback	118
Gambar 4. 15 Pengujian Keutuhan Penerusan Nilai PWM dari Jetson ke ESP32	119
Gambar 4. 16 Pengujian Pelaporan Status Kill Switch: (kiri) magnet terpasang, (kanan) magnet terlepas.....	120
Gambar 4. 17 Tampilan Citra Kiri dan Kanan.....	122
Gambar 4. 18 Proses Pengambilan data Papan Catur	122
Gambar 4. 19 Tampilan <i>Stereo Depth</i> pada Bawah Air menghadap Gawang 1	124
Gambar 4. 20 Anotasi Pelabelan Objek Gawang, Suar, dan Bak.....	128
Gambar 4. 21 Grafik Confusion Matrix.....	130
Gambar 4. 22 Precision-Recall Curve.....	135
Gambar 4. 23 Precision-Confidence Curve	136
Gambar 4. 24 Recall - Confidence Curve.....	137
Gambar 4. 25 F-Score Curve	138
Gambar 4. 26 Hasil Pengujian Deteksi Gawang pada Berbagai Jarak (1.2m, 1.6m, 2.0m, 2.4m)	140

Gambar 4. 27 Hasil Pengujian Deteksi Suar pada Berbagai Jarak (0.4m, 1.2m, 2.0m, 2.8m)	142
Gambar 4. 28 Hasil Pengujian Deteksi Suar pada Berbagai Jarak (0.4m, 1.6m, 2.8m, 0.8m)	143
Gambar 4. 29 Tampilan World Gazebo Arena KRBAI.....	144
Gambar 4. 30 Tampilan APF Depth View dari Kamera Stereo AUV	145
Gambar 4. 31 Tampilan Aplikasi KRBAI APF Monitor	146
Gambar 4. 32 Jalur Navigasi AUV pada Skenario A (kiri) dan Skenario B (kanan)	147
Gambar 4. 33 Skema Tata Letak Lintasan Uji dan State Misi pada Setiap Segme	151
Gambar 4. 34 Lintasan Uji yang Dibangun pada Kolam Tirta Sagoro (10,45 m × 5 m)	152
Gambar 4. 35 Pengujian Respons Perintah Geser terhadap Satu Objek Diam	154
Gambar 4. 36 Pengujian Respons Perintah Geser terhadap Dua Objek Diam .	156
Gambar 4. 37 Pengujian Respons Perintah Geser terhadap Rintangan yang Muncul Mendadak	159
Gambar 4. 38 Posisi Wahana saat Menabrak Tiang Gawang pada Pengujian APF Nonaktif.....	162
Gambar 4. 39 Kondisi Wahana saat Misi Selesai pada Pengujian APF Aktif	164

DAFTAR NOTASI

AUV	= Autonomous Underwater Vehicle, wahana bawah air otonom
KRBAI	= Kontes Robot Bawah Air Indonesia
ROS2	= <i>Robot Operating System 2</i> , middleware robotika
APF	= Artificial Potential Field, metode perencanaan lintasan berbasis medan potensial
YOLO	= You Only Look Once, metode deteksi objek berbasis deep learning
SGBM	= Semi-Global Block Matching, algoritma pencocokan stereo
PID	= Proportional–Integral–Derivative, metode kendali umpan balik
ESC	= Electronic Speed Controller
IMU	= Inertial Measurement Unit
PWM	= Pulse Width Modulation
ROI	= Region of Interest, wilayah citra yang diamati
EMA	= Exponential Moving Average, metode penghalusan data
mAP	= mean Average Precision, metrik ketelitian deteksi objek
MAE	= Mean Absolute Error, galat rata-rata absolut
Depth Map	= Representasi kedalaman setiap piksel citra
2.5D	= Representasi lingkungan dua dimensi dengan informasi kedalaman
q	= Vektor posisi AUV saat ini
q_{goal}	= Vektor posisi titik tujuan
q_{obs}	= Vektor posisi rintangan
$\rho(q, q_{goal})$	= Jarak Euclidean antara AUV dan titik tujuan
$\rho(q, q_{obs})$	= Jarak Euclidean antara AUV dan rintangan
X	= Vektor posisi AUV saat ini, $X = (x, y)^T$
X_g	= Vektor posisi AUV saat ini, $X = (x_g, y_g)^T$
x, y	= Koordinat posisi AUV pada bidang horizontal
x_g, y_g	= Koordinat posisi tujuan
U_{att}	= Fungsi medan potensial tarik (<i>attractive potential field</i>)
U_{rep}	= Fungsi medan potensial tolak (<i>repulsive potential field</i>)
$U(X)$	= Fungsi medan potensial total (<i>resultan</i>)
F_{att}	= Gaya Tarik Semu menuju target
F_{rep}	= Gaya tolak Semu menjauhi rintangan
$F(X)$	= Gaya resultan Semu yang bekerja pada AUV
k_{att}	= Konstanta penguat medan potensial tarik
k_{rep}	= Konstanta penguat medan potensial tolak
$d(X, X_g)$	= Jarak <i>Euclidean</i> antara AUV dan titik tujuan
$d(X, X_o)$	= Jarak <i>Euclidean</i> antara AUV dan rintangan
X_o	= Posisi rintangan
D_0	= Jarak pengaruh maksimum medan potensial tolak
d_{min}	= Batas bawah jarak pada perhitungan gaya tolak semu
α	= Konstanta penghalusan Exponential Moving Average
∇	= Operator gradien

Z	= Kedalaman atau jarak objek terhadap kamera
f	= Panjang fokus kamera <i>stereo</i>
b	= Jarak antar kamera <i>stereo</i> (<i>baseline</i>)
X_L	= Posisi objek pada citra kamera kiri
X_R	= Posisi objek pada citra kamera kanan
d	= Nilai <i>disparitas</i> citra <i>stereo</i>
K	= Matriks intrinsik kamera (citra mentah)
P	= Matriks proyeksi kamera (citra terektifikasi)
Z_{ukur}	= Jarak hasil estimasi stereo sebelum koreksi
$Z_{koreksi}$	= Jarak hasil estimasi stereo setelah koreksi bawah air
a	= Faktor skala pada fungsi koreksi jarak bawah air
b	= Konstanta offset pada fungsi koreksi jarak bawah air
n	= Indeks bias medium
cx_norm	= Posisi horizontal objek pada citra yang ternormalisasi
$e(t)$	= Galat antara nilai acuan dan nilai aktual
$u(t)$	= Sinyal kendali keluaran pengendali PID
K_p	= Konstanta proporsional pengendali PID
K_i	= Konstanta integral pengendali PID
K_d	= Konstanta derivatif pengendali PID
v	= Kecepatan gerak translasi
s	= Jarak tempuh
t	= Waktu tempuh
ω	= Kecepatan gerak sudut
$\Delta\theta$	= Perubahan sudut yang ditempuh

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Lautan yang menutupi lebih dari 70% permukaan Bumi menyimpan potensi sumber daya biologis dan mineral yang sangat besar. Namun, aktivitas eksplorasi laut dalam menghadapi berbagai tantangan signifikan akibat kondisi lingkungan yang ekstrem, seperti suhu yang sangat rendah, tekanan tinggi, serta pengaruh gaya hidrodinamika yang kompleks (Lu et al., 2025). Kondisi tersebut membatasi keterlibatan manusia secara langsung, sehingga dibutuhkan teknologi yang mampu beroperasi secara mandiri dan andal di lingkungan bawah air.

Robot bawah air berkembang sebagai salah satu solusi potensial untuk mendukung operasi jangka panjang, khususnya pada wilayah laut dalam. Dengan penerapan material khusus dan desain struktural yang sesuai, robot bawah air mampu bertahan terhadap tekanan tinggi, suhu rendah, serta kondisi ekstrem lainnya. Selain itu, integrasi berbagai sensor memungkinkan robot bawah air untuk melakukan pengumpulan data penting yang berkaitan dengan hidrologi, geologi laut, serta biologi laut secara efektif (Wang et al., 2025).

Salah satu jenis robot bawah air yang banyak dikembangkan adalah Autonomous Underwater Vehicle (AUV) (Fahmi et al., 2025), yaitu robot bawah air yang mampu beroperasi secara mandiri tanpa kendali langsung dari operator manusia. AUV dirancang untuk menjalankan misi tertentu berdasarkan perencanaan yang telah ditetapkan sebelumnya serta kemampuan pengambilan keputusan secara otonom. Dibandingkan dengan *Remotely Operated Vehicle* (ROV) yang bergantung pada kendali operator dan kabel, AUV memiliki *fleksibilitas* lebih tinggi untuk menjelajahi area yang luas dan kompleks. Oleh karena itu, AUV banyak dimanfaatkan dalam kegiatan pemetaan dasar laut, inspeksi infrastruktur bawah air, serta eksplorasi lingkungan laut dalam.

Untuk dapat beroperasi secara otonom, AUV harus memiliki kemampuan navigasi mandiri (Cong et al., 2021) yang dilengkapi dengan mekanisme *obstacle*

avoidance. *Obstacle avoidance* merupakan kemampuan sistem untuk mendeteksi keberadaan rintangan di lingkungan sekitar serta menentukan tindakan penghindaran yang aman tanpa mengganggu jalannya misi utama (Firmansyah et al., 2023). Kemampuan ini menjadi krusial pada lingkungan bawah air yang bersifat terbatas, dinamis, dan memiliki risiko tabrakan yang tinggi, baik dengan rintangan statis maupun dinamis.

Salah satu pendekatan yang banyak digunakan dalam navigasi lokal dan *obstacle avoidance* adalah perencanaan lintasan berbasis Artificial Potential Field (APF) (Nurhalim et al., 2022). Metode APF memodelkan target sebagai gaya tarik dan rintangan sebagai gaya tolak, sehingga arah gerak robot ditentukan berdasarkan resultan gaya yang bekerja pada sistem. Pendekatan ini bersifat reaktif, sederhana, dan sesuai untuk aplikasi *real-time*, sehingga banyak diterapkan pada sistem robot bergerak yang beroperasi pada lingkungan terbatas dan dinamis, termasuk robot bawah air.

Agar proses *obstacle avoidance* dapat berjalan secara efektif, Autonomous Underwater Vehicle (AUV) memerlukan informasi lingkungan yang akurat, khususnya terkait posisi dan jarak rintangan di sekitarnya. Salah satu pendekatan yang banyak digunakan untuk memperoleh informasi jarak adalah penggunaan kamera *stereo*. Kamera stereo bekerja berdasarkan prinsip *stereo vision*, yaitu memanfaatkan perbedaan sudut pandang dua kamera yang ditempatkan sejajar untuk menghitung jarak objek melalui nilai *disparity* (Ginting et al., 2019). Informasi kedalaman yang dihasilkan memungkinkan sistem untuk mengestimasi jarak rintangan secara lebih akurat dibandingkan dengan kamera *monocular*, sehingga sesuai untuk mendukung navigasi dan *obstacle avoidance* pada AUV.

Selain informasi kedalaman, persepsi visual juga diperlukan untuk mengidentifikasi objek atau rintangan pada citra. Pada lingkungan bawah air, pendekatan konvensional berbasis segmentasi warna kerap mengalami penurunan keandalan akibat variasi pencahayaan, perubahan sudut pandang, dan atenuasi warna oleh medium air. Oleh karena itu, penelitian ini menggunakan metode deteksi objek berbasis *deep learning* yaitu You Only Look Once (YOLO), yang mampu mengenali objek secara tangguh terhadap variasi kondisi visual tersebut.

Model YOLO dilatih menggunakan citra dari lingkungan pengujian sehingga sesuai dengan karakteristik objek lintasan KRBAI, dan dijalankan dengan akselerasi inference pada platform *embedded* agar tetap memenuhi kebutuhan operasi *real-time*. Hasil deteksi YOLO selanjutnya dikombinasikan dengan informasi kedalaman dari kamera stereo untuk menentukan posisi dan jarak rintangan secara presisi, sehingga sistem navigasi AUV dapat melakukan *obstacle avoidance* secara *real-time* dan andal. Untuk mendukung integrasi berbagai modul persepsi, perencanaan, dan kendali dalam sistem AUV, penelitian ini menggunakan *Robot Operating System 2* (ROS2) sebagai *middleware*. ROS2 (Suparno & Jalil, 2023) menyediakan kerangka kerja terdistribusi yang memungkinkan komunikasi antar modul sistem secara modular dan *real-time*. Dengan arsitektur berbasis *node* dan mekanisme komunikasi yang fleksibel, ROS2 memudahkan pengembangan serta integrasi sistem navigasi otonom yang kompleks. Oleh karena itu, ROS2 dipilih untuk mendukung implementasi sistem *obstacle avoidance* pada AUV agar dapat beroperasi secara responsif dan terstruktur.

Meskipun berbagai penelitian telah dilakukan dalam bidang deteksi objek dan estimasi jarak untuk robot bawah air, sebagian besar pendekatan tersebut masih berfokus pada masing-masing komponen secara terpisah, tanpa integrasi ke dalam sistem navigasi yang utuh. Padahal, pada skenario operasional nyata seperti Kontes Robot Indonesia (KRI) kategori Kontes Robot Bawah Air Indonesia (KRBAI), AUV dituntut untuk mampu memanfaatkan informasi persepsi lingkungan secara langsung dalam proses pengambilan keputusan navigasi (Pusat Prestasi Nasional, Kementerian Pendidikan, Kebudayaan, Riset, 2024). Integrasi deteksi objek berbasis *deep learning* dan informasi kedalaman stereo ke dalam suatu representasi lingkungan lokal berbasis zona kedalaman yang dapat digunakan oleh *local planner* untuk melakukan *obstacle avoidance* secara *real-time* masih menjadi tantangan utama, khususnya pada sistem AUV yang harus beroperasi secara mandiri dan responsif terhadap perubahan lingkungan.

Sejalan dengan tuntutan tersebut, Kontes Robot Bawah Air Indonesia (KRBAI) mensyaratkan AUV untuk mampu menyelesaikan lintasan secara mandiri dengan menghadapi berbagai rintangan tanpa intervensi operator. Oleh karena itu,

penelitian ini berfokus pada perancangan dan implementasi sistem *obstacle avoidance* pada AUV berbasis deteksi objek You Only Look Once (YOLO), informasi kedalaman kamera stereo, dan perencanaan gerak menggunakan Artificial Potential Field (APF) yang diintegrasikan dalam platform Robot *Operating System 2* (ROS2). Sistem yang dikembangkan diharapkan mampu meningkatkan kemampuan navigasi mandiri AUV serta memenuhi kebutuhan operasional pada lintasan kompetisi KRBAI.

1.2 Pertanyaan Penelitian

Pertanyaan pada penelitian ini adalah :

1. Bagaimana mengimplementasikan metode You Only Look Once (YOLO) yang dikombinasikan dengan informasi kedalaman dari kamera stereo untuk mendeteksi serta mengestimasi posisi dan jarak rintangan di sekitar AUV?
2. Bagaimana penerapan *local planner* berbasis Artificial Potential Field (APF) dalam menentukan arah dan kecepatan gerak AUV untuk melakukan *obstacle avoidance* secara *real-time*?
3. Bagaimana kinerja sistem *obstacle avoidance* yang dikembangkan dalam menghadapi lintasan uji yang menyerupai arena Kontes Robot Bawah Air Indonesia (KRBAI) berdasarkan tingkat keberhasilan navigasi dan jarak aman terhadap rintangan?

1.3 Batasan Penelitian

Batasan pada penelitian ini adalah :

1. Sistem dikembangkan pada Autonomous Underwater Vehicle (AUV) yang beroperasi pada lingkungan air tenang dan terkontrol, menyerupai arena Kontes Robot Bawah Air Indonesia (KRBAI).
2. Navigasi yang dikembangkan bersifat lokal (*local navigation*), sehingga sistem tidak menggunakan peta global lingkungan maupun metode pemetaan simultan seperti *Simultaneous Localization and Mapping* (SLAM).
3. Pengendali *Proportional Integral Derivative* (PID) pada penelitian ini berfungsi sebagai pengendali pendukung untuk menjaga kedalaman

dan arah hadap wahana, dan bukan merupakan objek penelitian. Penyetelan nilai parameter PID dilakukan secara empiris melalui percobaan langsung pada wahana. Pemodelan matematis wahana dalam bentuk fungsi alih beserta identifikasi parameter hidrodinamika tidak dilakukan, demikian pula perbandingan terhadap metode penyetelan klasik seperti *Ziegler-Nichols* maupun *Cohen-Coon*.

4. Deteksi objek dan rintangan dilakukan menggunakan metode deteksi objek berbasis *deep learning* You Only Look Once (YOLO), dengan objek yang dibatasi pada tiga kelas elemen lintasan arena KRBAI, yaitu gawang, suar, dan bak
5. Estimasi jarak rintangan diperoleh dari informasi kedalaman kamera stereo, tanpa memanfaatkan sensor jarak tambahan seperti sonar atau LiDAR.
6. Representasi lingkungan lokal berbasis zona kedalaman lingkungan direpresentasikan dalam bentuk dua dimensi dengan informasi kedalaman terbatas (2.5D), di mana perencanaan lintasan dilakukan pada bidang horizontal, sedangkan kendali kedalaman (*depth*) ditangani secara terpisah menggunakan sistem kendali berbasis PID.
7. Metode perencanaan lintasan lokal yang digunakan adalah Artificial Potential Field (APF), tanpa dilakukan perbandingan dengan metode perencanaan jalur global atau algoritma perencanaan lintasan lainnya.
8. Sistem dikembangkan dan diintegrasikan menggunakan *Robot Operating System 2* (ROS2) sebagai *middleware* untuk komunikasi antar modul sistem.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Merancang sistem *obstacle avoidance* pada Autonomous Underwater Vehicle (AUV) berbasis persepsi visual dan informasi kedalaman untuk mendukung navigasi otonom.
2. Mengimplementasikan metode YOLO yang dikombinasikan dengan informasi kedalaman dari kamera stereo untuk mendeteksi serta

mengestimasi posisi dan jarak rintangan di sekitar AUV secara *real-time*.

3. Membangun representasi lingkungan lokal AUV berbasis zona kedalaman (2.5D) berdasarkan hasil persepsi visual dan estimasi kedalaman sebagai dasar perencanaan lintasan.
4. Menerapkan algoritma Artificial Potential Field (APF) sebagai *local planner* untuk menentukan arah dan kecepatan gerak AUV dalam melakukan *obstacle avoidance*.
5. Mengevaluasi kinerja sistem *obstacle avoidance* berdasarkan tingkat keberhasilan navigasi, jarak aman terhadap rintangan, serta kemampuan AUV dalam melewati lintasan uji yang menyerupai arena Kontes Robot Bawah Air Indonesia (KRBAI).

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Penelitian ini diharapkan dapat menjadi referensi dalam pengembangan sistem navigasi mandiri pada Autonomous Underwater Vehicle (AUV), khususnya pada penerapan *obstacle avoidance* berbasis persepsi visual dan informasi kedalaman. Selain itu, penelitian ini memberikan kontribusi kajian ilmiah terkait pemanfaatan YOLO, kamera stereo, representasi lingkungan berbasis zona, dan algoritma Artificial Potential Field (APF) dalam sistem navigasi AUV berbasis *Robot Operating System 2* (ROS2).
2. Penelitian ini dapat digunakan sebagai acuan dalam pengembangan AUV yang mampu menghindari rintangan secara mandiri pada lintasan terstruktur seperti Kontes Robot Bawah Air Indonesia (KRBAI). Sistem yang dikembangkan diharapkan dapat meningkatkan keselamatan dan keandalan operasi AUV dengan meminimalkan risiko tabrakan terhadap rintangan di lingkungan bawah air.
3. Penelitian ini memberikan kontribusi berupa implementasi sistem *obstacle avoidance* secara *end-to-end* yang mengintegrasikan persepsi visual, pemodelan lingkungan (representasi lingkungan berbasis zonal), dan perencanaan lintasan lokal secara *real-time*. Sistem yang dihasilkan

dapat dikembangkan lebih lanjut untuk aplikasi robotika bawah air lainnya, seperti inspeksi infrastruktur bawah air, pemetaan lingkungan perairan dangkal, dan eksplorasi bawah air.

(Halaman ini sengaja dikosongkan)

BAB 2

TINJAUAN PUSTAKA

2.1 Kajian Penelitian Terdahulu

Penelitian yang dilakukan (C. Chen et al., 2025), berfokus pada pengembangan sistem persepsi visual bawah air secara *real-time* untuk mendukung pelacakan objek oleh Autonomous Underwater Vehicle (AUV). Permasalahan utama yang diangkat adalah keterbatasan sumber daya komputasi pada perangkat *embedded*, sehingga algoritma visi berbasis *deep learning* sulit dijalankan secara *real-time*. Untuk mengatasi hal tersebut, penulis mengusulkan sistem visi bawah air yang dioptimalkan menggunakan model *deep learning* ringan dan teknik akselerasi komputasi. Sistem ini berhasil mencapai keseimbangan antara akurasi pelacakan dan kecepatan pemrosesan pada perangkat *edge*. Namun, penelitian ini tidak membahas perencanaan lintasan maupun algoritma *obstacle avoidance*, sehingga hanya berperan pada aspek persepsi visual AUV.

Penelitian yang dilakukan (Yu & Ning, 2023), mengusulkan metode *Improved Artificial Potential Field* (IAPF) yang dikombinasikan dengan *consistency protocol* untuk mendukung penghindaran rintangan secara terkoordinasi pada sistem *multi-AUV*. Pengembangan dilakukan dengan memodifikasi fungsi medan potensial klasik agar mampu menghindari permasalahan *local minimum* dan konflik lintasan antar AUV. Pendekatan ini efektif untuk skenario *multi-robot* yang kompleks, namun memiliki tingkat kompleksitas komputasi yang relatif tinggi dan tidak secara khusus mempertimbangkan keterbatasan sistem *embedded* atau kebutuhan implementasi *real-time* pada AUV tunggal.

Penelitian yang dilakukan (Liu & Wang, 2021), mengintegrasikan *stereo vision* dengan *Improved Artificial Potential Field* untuk mendukung penghindaran rintangan pada formasi kendaraan udara *fixed-wing*. Informasi kedalaman dari *stereo vision* digunakan untuk meningkatkan akurasi estimasi posisi rintangan, sementara fungsi APF dimodifikasi untuk memperhitungkan kecepatan relatif dan dinamika formasi. Meskipun pendekatan ini menunjukkan performa yang baik, penelitian dilakukan pada platform kendaraan udara dan bukan AUV, sehingga

karakteristik lingkungan dan dinamika yang dibahas berbeda dengan sistem bawah air.

Penelitian yang dilakukan (Xing et al., 2023), mengusulkan metode perencanaan lintasan AUV berbasis *Improved Artificial Potential Field* yang dipadukan dengan *Deep Reinforcement Learning* (DRL). Pengembangan dilakukan dengan menambahkan konsep *traction force* untuk mengatasi permasalahan *local minimum* serta memanfaatkan data *multisensor* dalam pembentukan *state space*.

Pendekatan ini mampu meningkatkan performa navigasi AUV di lingkungan kompleks, namun membutuhkan komputasi yang besar dan proses pelatihan yang tidak sederhana, sehingga kurang sesuai untuk sistem dengan keterbatasan sumber daya atau skenario kompetisi yang menuntut keandalan dan kesederhanaan sistem.

Penelitian yang dilakukan (R. Chen & Chen, 2023), mengembangkan metode deteksi objek bawah air berbasis You Only Look Once (YOLO) versi 5 yang ditingkatkan dan diimplementasikan pada wahana bawah air otonom jenis *Autonomous Underwater Helicopter*. Permasalahan yang diangkat adalah degradasi kualitas citra bawah air akibat atenuasi cahaya dan hamburan partikel yang menyebabkan metode deteksi konvensional tidak andal. Peningkatan dilakukan melalui modifikasi arsitektur jaringan untuk memperbaiki akurasi deteksi objek pada kondisi visual bawah air yang kompleks, dan hasil pengujian menunjukkan model mampu mendeteksi objek target secara akurat pada perangkat yang terpasang langsung di wahana. Namun, penelitian ini berfokus pada aspek deteksi objek saja dan tidak mengintegrasikan hasil deteksi dengan estimasi jarak maupun sistem perencanaan lintasan, sehingga belum membentuk sistem navigasi *obstacle avoidance* yang utuh.

Tabel 2. 1 Perubahan dan Perkembangan dari penelitian sebelumnya

No.	Penulis & Tahun	Judul Jurnal	Fokus Utama Penelitian	Perbedaan / Kontribusi Penelitian Ini
1	(C. Chen et al., 2025)	<i>Real-Time Underwater Vision Sensing System for AUV Tracking</i>	Persepsi visual bawah air <i>real-time</i> dan pelacakan objek berbasis deep learning	Penelitian ini tidak hanya berfokus pada persepsi visual, tetapi mengintegrasikan deteksi objek YOLO dan informasi kedalaman stereo depth secara langsung ke

No.	Penulis & Tahun	Judul Jurnal	Fokus Utama Penelitian	Perbedaan / Kontribusi Penelitian Ini
				dalam perencanaan lintasan dan <i>obstacle avoidance</i> berbasis APF dalam satu sistem navigasi utuh.
2	(Yu & Ning, 2023)	<i>Coordinated Obstacle Avoidance of Multi-AUV Based on Improved Artificial Potential Field Method and Consistency Protocol</i>	<i>Improved</i> APF untuk penghindaran rintangan terkoordinasi pada <i>multi-AUV</i>	Penelitian ini memilih APF klasik tanpa DRL untuk mengurangi kompleksitas perencanaan, dengan kompleksitas komputasi dialokasikan pada persepsi berbasis YOLO yang terakselerasi <i>TensorRT</i> .
3	(Liu & Wang, 2021)	<i>Research on Obstacle Avoidance Technology of Fixed Wing Formation Based on Improved Artificial Potential Field Method with Stereo Vision</i>	Integrasi <i>stereo vision</i> dan IAPF pada kendaraan udara <i>fixed-wing</i>	Penelitian ini diterapkan pada lingkungan bawah air, dengan pemodelan kedalaman berbasis <i>stereo depth</i> untuk AUV
4	(Xing et al., 2023)	<i>Improved Artificial Potential Field Algorithm Assisted by Multisource Data for AUV Path Planning</i>	IAPF + DRL untuk perencanaan lintasan AUV	Penelitian ini memilih APF klasik tanpa DRL untuk mengurangi kompleksitas dan meningkatkan keterterapan pada sistem <i>embedded</i> dan kompetisi KRBAI
5	(R. Chen & Chen, 2023)	<i>Improved Convolutional Neural Network YOLOv5 for Underwater Target Detection Based on</i>	Deteksi objek bawah air berbasis YOLOv5 yang ditingkatkan pada wahana	Penelitian ini tidak berhenti pada deteksi objek, melainkan menggabungkan deteksi YOLO dengan estimasi jarak <i>stereo depth</i> dan mengintegrasikannya ke

No.	Penulis & Tahun	Judul Jurnal	Fokus Utama Penelitian	Perbedaan / Kontribusi Penelitian Ini
		<i>Autonomous Underwater Helicopter</i>	bawah air otonom	perencanaan lintasan APF dalam satu sistem navigasi <i>real-time</i> untuk KRBAI

(Sumber : Dokumentasi Pribadi)

Berdasarkan kajian penelitian terdahulu, dapat disimpulkan bahwa pengembangan metode Artificial Potential Field (APF) dalam beberapa tahun terakhir cenderung mengarah pada peningkatan kompleksitas melalui pendekatan *improved* maupun *hybrid*, seperti penggabungan dengan metode optimasi atau *Deep Reinforcement Learning*. Meskipun pendekatan tersebut mampu meningkatkan performa navigasi dan mengatasi berbagai keterbatasan APF klasik, kebutuhan implementasi *real-time*, keterbatasan sumber daya komputasi pada sistem *embedded*, serta tuntutan keandalan sistem pada skenario kompetisi menjadi pertimbangan utama dalam penelitian ini. Oleh karena itu, penelitian ini menerapkan pembagian kompleksitas yang proporsional pada setiap lapisan sistem pada sisi persepsi digunakan metode deteksi objek berbasis *deep learning* You Only Look Once (YOLO) yang dikombinasikan dengan estimasi jarak *stereo depth* untuk memperoleh informasi lingkungan yang andal terhadap variasi kondisi visual bawah air, sedangkan pada sisi perencanaan lintasan digunakan APF klasik yang bersifat reaktif dan ringan secara komputasi. Kombinasi ini mampu mencapai keseimbangan antara akurasi persepsi, kesederhanaan algoritma perencanaan, dan keterterapan sistem secara *real-time* pada Autonomous Underwater Vehicle (AUV) berbasis *embedded*.

2.2 Autonomous Underwater Vehicle

Autonomous Underwater Vehicle (AUV) merupakan wahana bawah air yang dirancang untuk beroperasi secara mandiri dengan kemampuan menyelam dan bermanuver menggunakan sistem propulsi internal. AUV umumnya ditenagai oleh sumber energi seperti baterai atau sel bahan bakar, serta dikendalikan oleh sistem komputasi yang terpasang di dalam wahana (*onboard computer*) untuk menjalankan fungsi navigasi, pengendalian, dan pengambilan keputusan secara otonom (Yang et al., 2021). Perkembangan pesat dalam bidang ilmu material, perangkat keras komputasi, dan teori kendali dalam beberapa tahun terakhir telah

meningkatkan keandalan serta efisiensi AUV, sehingga menjadikannya solusi yang lebih ekonomis dan fleksibel dibandingkan kapal selam berawak maupun *Remotely Operated Vehicle* (ROV) dalam pelaksanaan misi bawah air.

Berbeda dengan ROV yang bergantung pada kendali operator melalui kabel (*tether*), AUV dirancang untuk menjalankan misi secara mandiri tanpa intervensi manusia secara langsung. Kemampuan ini memungkinkan AUV untuk beroperasi pada area yang luas, kompleks, dan sulit dijangkau, seperti pemetaan dasar laut, inspeksi struktur bawah air, eksplorasi sumber daya laut, serta kompetisi robot bawah air. Namun, sifat otonom tersebut menuntut AUV memiliki sistem persepsi lingkungan, navigasi, dan pengambilan keputusan yang andal agar dapat bergerak secara aman di lingkungan bawah air yang dinamis dan tidak terstruktur.

Lingkungan bawah air memiliki tantangan yang signifikan bagi sistem navigasi AUV, seperti keterbatasan visibilitas, gangguan cahaya, arus air, serta ketiadaan sistem penentuan posisi global seperti GPS. Kondisi ini menyebabkan AUV harus mengandalkan sensor internal dan sistem persepsi berbasis kamera atau sensor jarak untuk memahami kondisi sekitarnya. Keterbatasan tersebut menuntut sistem navigasi yang adaptif dan mampu mengambil keputusan secara *real-time* berdasarkan informasi lingkungan yang tersedia. Oleh karena itu, kemampuan navigasi mandiri dan penghindaran rintangan (*obstacle avoidance*) menjadi aspek krusial dalam perancangan sistem AUV, khususnya untuk memastikan keselamatan wahana dan keberhasilan misi (Jalal & Nasir, 2021).

Dalam konteks penelitian ini, AUV dirancang untuk mampu melakukan navigasi lokal secara mandiri dengan memanfaatkan informasi lingkungan yang diperoleh dari sistem persepsi visual. Data persepsi tersebut selanjutnya digunakan dalam proses perencanaan lintasan dan pengambilan keputusan gerak secara *real-time*. Pendekatan ini memungkinkan AUV untuk bereaksi secara adaptif terhadap keberadaan rintangan di sekitarnya, sehingga mendukung pergerakan yang aman dan stabil selama menjalankan misi. Lebih jauh lagi, integrasi antara pemrosesan data visual dan algoritma navigasi yang responsif akan memastikan wahana mampu mengkalkulasi jalur alternatif secara cepat. Keandalan sistem ini diharapkan dapat meminimalisir risiko kegagalan misi akibat benturan fisik, sekaligus

mengoptimalkan efisiensi pergerakan wahana saat bermanuver di lingkungan yang kompleks.



Gambar 2. 1 Autonomous Underwater Vehicle HYDROships

(Sumber : Dokumentasi Pribadi)

Sebagai ilustrasi, Gambar 2.1 menunjukkan salah satu contoh AUV yang dikembangkan oleh Politeknik Perkapalan Negeri Surabaya, yaitu tim HYDROships, yang berhasil meraih posisi *Runners-Up* pada kompetisi SAUVC 2025. Keberhasilan tersebut menunjukkan bahwa pengembangan AUV dengan sistem navigasi dan pengendalian yang baik memiliki peran penting dalam menyelesaikan misi bawah air secara mandiri dan kompetitif. Contoh ini memperlihatkan relevansi pengembangan AUV dalam konteks penelitian dan kompetisi robotika bawah air, serta menjadi salah satu motivasi dalam pengembangan sistem AUV pada penelitian ini.

2.3 Navigasi dan *Obstacle Avoidance*

Robot bergerak secara mandiri memerlukan kemampuan navigasi untuk dapat beroperasi di lingkungannya. Kemampuan ini mencakup proses penginderaan lingkungan sekitar, pengolahan informasi hasil sensor untuk memperkirakan posisi robot, serta perencanaan lintasan secara *real-time* yang menghubungkan posisi awal menuju target. Selama proses tersebut, robot harus mampu menyesuaikan kecepatan dan arah geraknya sekaligus menghindari keberadaan rintangan agar dapat mencapai tujuan dengan aman (Saleh et al., 2025).

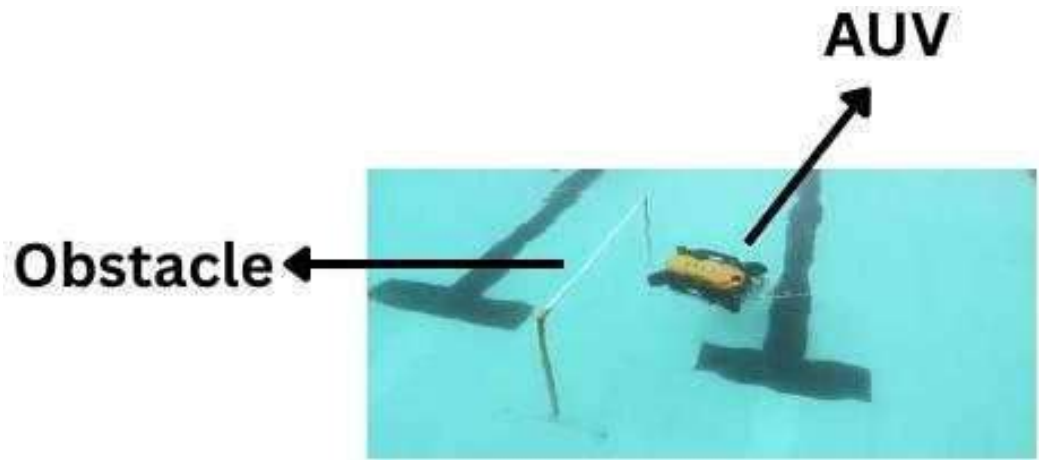
Obstacle avoidance merupakan kemampuan robot untuk mendeteksi, mengidentifikasi, serta memperkirakan posisi rintangan fisik di sekitarnya, kemudian mengambil tindakan yang sesuai guna menghindari potensi tabrakan. Kemampuan ini berperan penting dalam menjaga keselamatan fisik robot sekaligus memastikan keberhasilan penyelesaian tugas. Dalam pengembangan sistem robotika, *obstacle avoidance* menjadi komponen fundamental yang memungkinkan robot berinteraksi secara efektif dengan lingkungan yang kompleks dan dinamis (Widodo & Sumaedi, 2024).

Navigasi dan *obstacle avoidance* merupakan dua aspek yang saling berkaitan dalam sistem pergerakan robot. Navigasi berfokus pada perencanaan dan pencapaian tujuan akhir, sedangkan *obstacle avoidance* memastikan bahwa lintasan yang ditempuh tetap aman dari rintangan selama perjalanan menuju tujuan tersebut. Berdasarkan cakupan perencanaannya, navigasi robot umumnya dibedakan menjadi navigasi global dan navigasi lokal. Navigasi global memanfaatkan peta lingkungan untuk merencanakan lintasan secara menyeluruh, sementara navigasi lokal menitikberatkan pada pengambilan keputusan berdasarkan kondisi lingkungan di sekitar robot secara *real-time*.

Pada Autonomous Underwater Vehicle (AUV), permasalahan navigasi dan *obstacle avoidance* menjadi lebih kompleks dibandingkan robot darat. Hal ini disebabkan oleh keterbatasan sistem penentuan posisi, kondisi visual yang tidak stabil, serta pengaruh faktor lingkungan bawah air seperti arus dan visibilitas yang rendah. Oleh karena itu, berbagai pendekatan *obstacle avoidance* telah dikembangkan, mulai dari metode reaktif hingga metode perencanaan lintasan lokal, dengan pemilihan pendekatan yang disesuaikan dengan karakteristik lingkungan dan kebutuhan sistem navigasi.

Berdasarkan pertimbangan tersebut, penelitian ini difokuskan pada pengembangan sistem *obstacle avoidance* berbasis navigasi lokal untuk AUV yang beroperasi pada lingkungan terstruktur, sehingga AUV mampu merespons rintangan secara adaptif dan *real-time* selama menjalankan misi. Penerapan pendekatan ini diharapkan dapat meningkatkan persentase keberhasilan manuver penghindaran tanpa mengorbankan stabilitas dinamis wahana. Selain itu, hasil

penelitian ini diharapkan mampu memberikan kontribusi aplikatif bagi pengembangan teknologi kendali robotika bawah air di masa mendatang.



Gambar 2. 2 Contoh skenario AUV menghadapi rintangan pada lingkungan bawah air

(Sumber : Dokumentasi Pribadi)

Contoh kondisi AUV saat menghadapi rintangan di lingkungan bawah air ditunjukkan pada Gambar 2.2 menunjukkan contoh kondisi operasional Autonomous Underwater Vehicle (AUV) pada lingkungan bawah air yang dilengkapi dengan rintangan (*obstacle*). Pada kondisi tersebut, AUV dituntut untuk mampu mendeteksi keberadaan rintangan di sekitarnya serta menentukan arah gerak yang aman untuk menghindari potensi tabrakan. Ilustrasi ini menggambarkan pentingnya kemampuan navigasi dan *obstacle avoidance* pada AUV, khususnya ketika beroperasi di lingkungan terbatas dan terstruktur seperti kolam uji atau arena kompetisi.

2.4 Kendali *Proportional Integral Derivative* (PID)

Proportional Integral Derivative (PID) merupakan metode kendali umpan balik tertutup (*closed-loop*) yang paling banyak digunakan pada sistem kendali industri maupun robotika karena strukturnya sederhana, tidak memerlukan model matematis lengkap dari sistem yang dikendalikan, serta ringan secara komputasi. Prinsip kerja PID adalah menghitung galat (*error*), yaitu selisih antara nilai acuan (*setpoint*) dengan nilai aktual yang terukur oleh sensor, kemudian mengolah galat tersebut menjadi sinyal kendali bagi aktuator. Secara matematis, keluaran pengendali PID dinyatakan pada Persamaan 2.1.

$$u(t) = K_p \cdot e(t) + K_i \cdot \int e(t)dt + K_d \cdot de(t)/dt \quad (2.1)$$

dengan $u(t)$ adalah sinyal kendali, $e(t)$ adalah galat pada waktu t , K_p adalah konstanta proporsional, K_i adalah konstanta integral, dan K_d adalah konstanta derivatif. Ketiga suku tersebut memiliki peran berbeda. Suku proporsional menghasilkan koreksi yang sebanding dengan besar galat saat ini, suku integral menjumlahkan galat sepanjang waktu sehingga mampu menghilangkan galat tunak (*steady-state error*), dan suku derivatif memperhitungkan laju perubahan galat sehingga meredam gerakan dan mencegah sistem melewati nilai acuan secara berlebihan (*overshoot*). Pengaruh perubahan ketiga konstanta terhadap tanggapan sistem dirangkum pada Tabel 2.2.

Tabel 2. 2 Pengaruh Peningkatan Konstanta PID terhadap Tanggapan Sistem

Konstanta	Waktu Naik	Overshoot	Waktu Tunak	Galat Tunak
K_p	Menurun	Meningkat	Sedikit berubah	Menurun
K_i	Menurun	Meningkat	Meningkat	Hilang
K_d	Sedikit berubah	Menurun	Menurun	Tidak berubah

(Sumber : Dokumentasi Pribadi)

Penentuan nilai ketiga konstanta dapat dilakukan secara analitis apabila fungsi alih (*transfer function*) sistem diketahui, maupun secara empiris melalui penyetelan langsung pada sistem nyata dengan berpedoman pada Tabel 2.2. Pada wahana bawah air, pendekatan empiris lebih lazim digunakan karena penurunan fungsi alih menuntut identifikasi parameter hidrodinamika seperti massa tambahan (*added mass*) dan koefisien hambatan, sedangkan dinamika wahana bersifat taklinear sehingga satu fungsi alih linear hanya berlaku di sekitar satu titik kerja tertentu.

Pada implementasi praktis, keluaran pengendali PID umumnya dibatasi pada nilai maksimum tertentu melalui mekanisme pembatasan keluaran (*output clamping*). Pembatasan ini mencegah penumpukan nilai pada suku integral ketika keluaran melampaui kemampuan aktuator, suatu kondisi yang dikenal sebagai

integral *windup*, sekaligus membatasi kecepatan gerak sistem agar tetap halus dan tidak menimbulkan osilasi.

2.5 Persepsi Visual Robot Bawah Air

Persepsi lingkungan merupakan aspek penting dalam pengoperasian robot bawah air, khususnya untuk mendukung navigasi dan penghindaran rintangan. Berbagai sensor telah dikembangkan untuk memperoleh informasi lingkungan, salah satunya adalah sistem radar. Meskipun radar mampu menghasilkan pencitraan lingkungan, penerapannya pada kendaraan otonom berukuran kecil menghadapi keterbatasan akibat ukuran perangkat yang relatif besar. Beberapa penelitian telah menguji penggunaan radar pada kendaraan otonom di lingkungan laut, seperti radar otomotif yang diaplikasikan pada kondisi perairan. Namun, jangkauan sistem yang terbatas, yaitu sekitar 100 meter, menyebabkan radar hanya efektif untuk penghindaran tabrakan jarak dekat. Selain itu, penggunaan radar navigasi dengan jangkauan hingga beberapa kilometer membutuhkan antena dan volume perangkat yang besar, sehingga kurang sesuai untuk diterapkan pada AUV berukuran kompak (Zalewski & Hozyń, 2024).

Sebagai alternatif, sistem penglihatan berbasis kamera menjadi solusi yang lebih umum digunakan karena bersifat relatif murah, ringan, dan mudah diintegrasikan. Kamera cahaya tampak maupun inframerah memungkinkan perekaman lingkungan sekitar AUV, baik untuk objek yang berada pada jarak dekat maupun jarak jauh. Dengan ukuran perangkat yang kecil dan biaya yang rendah, sistem penglihatan banyak digunakan pada berbagai kendaraan tanpa awak, sehingga memungkinkan pendeteksian rintangan serta identifikasi objek di lingkungan sekitar.

Salah satu pendekatan yang banyak diterapkan dalam persepsi visual adalah penggunaan sistem kamera ganda atau *stereo vision*. Sistem ini memungkinkan ekstraksi informasi tiga dimensi dengan memanfaatkan perbedaan sudut pandang dua kamera yang dipasang sejajar secara horizontal. Informasi kedalaman relatif yang dihasilkan dapat digunakan untuk memperkirakan posisi kendaraan terhadap objek-objek di sekitarnya. Meskipun demikian, penerapan sistem *stereo vision* memerlukan jarak tertentu antar kamera, yang berdampak pada peningkatan ruang fisik yang dibutuhkan oleh sistem sensor.

Pendekatan lain dalam persepsi visual adalah penggunaan kamera tunggal (*monocular vision*). Keunggulan utama metode ini terletak pada bobot dan dimensi perangkat yang kecil, sehingga mudah diterapkan pada berbagai jenis kendaraan otonom. Namun, keterbatasan utama kamera tunggal adalah kesulitan dalam menentukan jarak objek secara akurat. Beberapa metode telah dikembangkan untuk mengatasi permasalahan tersebut, seperti pendekatan berbasis pengukuran sudut *horizontal*. Akan tetapi, metode ini masih memerlukan intervensi manusia untuk menentukan titik-titik karakteristik pada citra. Selain itu, kamera pada AUV juga banyak dimanfaatkan untuk estimasi pose relatif, proses *docking* berbasis *landmark*, serta deteksi rintangan dan pelacakan objek di lingkungan maritim.

Pengembangan sistem persepsi visual yang andal menuntut penerapan algoritma yang mampu mengekstraksi informasi objek secara optimal dari citra masukan. Pendekatan konvensional seperti segmentasi citra berbasis warna atau ambang batas memiliki keterbatasan pada lingkungan bawah air, karena kualitas citra sangat dipengaruhi oleh atenuasi cahaya, hamburan partikel, dan perubahan karakteristik warna terhadap kedalaman. Kondisi tersebut menyebabkan parameter segmentasi yang telah dikalibrasi pada satu kondisi menjadi tidak konsisten pada kondisi pencahayaan atau jarak yang berbeda. Oleh karena itu, pendekatan deteksi objek berbasis *deep learning* semakin banyak digunakan pada persepsi visual bawah air karena mampu mengenali objek berdasarkan fitur yang dipelajari dari data, sehingga lebih tangguh terhadap variasi kondisi visual. Dalam penelitian ini, deteksi objek dilakukan menggunakan metode You Only Look Once (YOLO) yang dikombinasikan dengan informasi kedalaman dari kamera *stereo*, sebagaimana dijelaskan lebih lanjut pada subbab berikutnya. Selain pemilihan algoritma deteksi, keandalan sistem persepsi visual bawah air juga dipengaruhi oleh karakteristik optis medium air itu sendiri. Cahaya mengalami atenuasi yang tidak seragam terhadap panjang gelombang, dimana komponen warna merah teredam paling cepat sehingga citra bawah air cenderung didominasi warna hijau kebiruan seiring bertambahnya kedalaman. Partikel terlarut di dalam air juga menimbulkan hamburan balik (*backscatter*) yang menurunkan kontras citra dan mengaburkan tekstur permukaan objek. Penurunan tekstur tersebut berpengaruh langsung terhadap proses pencocokan stereo, karena algoritma pencocokan bergantung pada keberadaan pola

tekstur untuk menemukan korespondensi *piksel* antara citra kiri dan citra kanan. Oleh sebab itu, sistem persepsi bawah air perlu dirancang agar tetap menghasilkan estimasi jarak yang stabil meskipun sebagian *piksel* pada peta kedalaman yang dihasilkan tidak valid.



Gambar 2. 3 Contoh visualisasi hasil persepsi lingkungan AUV terhadap rintangan dan target

(Sumber : Dokumentasi Pribadi)

Gambar 2.3 menunjukkan contoh kondisi operasional Autonomous Underwater Vehicle (AUV) pada lingkungan bawah air yang dilengkapi dengan rintangan (*obstacle*). Pada citra tersebut, sistem persepsi visual digunakan untuk mengenali keberadaan objek di lintasan, seperti rintangan (*obstacle*) dan area target, sebagai dasar pengambilan keputusan navigasi. Informasi visual yang diperoleh selanjutnya dapat dikombinasikan dengan data kedalaman untuk memperkirakan posisi relatif objek terhadap AUV, sehingga mendukung proses navigasi dan penghindaran rintangan secara aman dan *real-time*.

2.6 Robot Operating System 2 (ROS2)



Gambar 2. 4 Logo ROS2

(Sumber : <https://foxglove.dev/blog/the-building-blocks-of-ros2>)

Robot Operating System 2 (ROS2) merupakan pengembangan generasi kedua dari *Robot Operating System (ROS)* yang dirancang untuk mengatasi berbagai keterbatasan pada ROS generasi sebelumnya, khususnya terkait dukungan sistem terdistribusi, kemampuan *real-time*, keandalan komunikasi, serta aspek keamanan. ROS telah lama dimanfaatkan sebagai *middleware* dalam pengembangan sistem robotika karena menyediakan kumpulan pustaka, *tools*, dan konvensi standar yang mempermudah integrasi antara sensor, *aktuator*, serta algoritma tingkat tinggi, seperti persepsi, navigasi, dan kendali. Melalui pendekatan modular berbasis *node*, ROS memungkinkan pengembang untuk membangun sistem robot yang kompleks secara bertahap dan terstruktur. Namun demikian, ROS 1 memiliki sejumlah keterbatasan mendasar, antara lain ketergantungan pada ROS Master sebagai pusat komunikasi, keterbatasan *skalabilitas* untuk sistem multi-robot, serta tidak tersedianya mekanisme *Quality of Service (QoS)* dan dukungan *real-time* yang memadai, sehingga kurang optimal untuk aplikasi robot otonom yang bersifat kritis (Al-Batati et al., 2024).

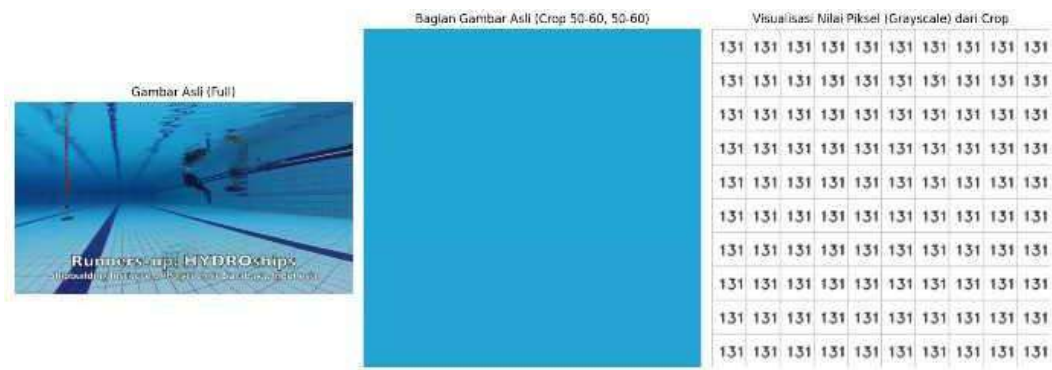
Untuk mengatasi permasalahan tersebut, ROS2 dikembangkan dengan arsitektur terdistribusi yang menghilangkan kebutuhan ROS Master dan memanfaatkan *middleware Data Distribution Service (DDS)*. Penggunaan DDS memungkinkan komunikasi *publish-subscribe* dengan dukungan QoS yang dapat dikonfigurasi, seperti reliabilitas, latensi, dan prioritas pesan, sehingga meningkatkan keandalan dan determinisme sistem dalam menangani data sensor maupun perintah kendali, terutama pada sistem robot yang beroperasi di lingkungan dinamis dan memiliki keterbatasan komunikasi, seperti lingkungan bawah air.

ROS2 juga mendukung aplikasi *real-time* dan sistem tertanam dengan menyediakan mekanisme eksekusi paralel melalui *executor* yang lebih fleksibel dibandingkan ROS 1. Kemampuan ini sangat penting bagi sistem robot otonom, termasuk Autonomous Underwater Vehicle (AUV), yang memerlukan pemrosesan persepsi, perencanaan lintasan, dan pengendalian aktuator secara simultan dan responsif terhadap perubahan lingkungan.

Selain itu, ROS2 dilengkapi dengan fitur keamanan melalui *Secure ROS 2 (SROS2)* yang mendukung autentikasi, enkripsi komunikasi, dan pengaturan hak akses antar *node*. Dalam konteks navigasi dan *obstacle avoidance*, ROS2

menyediakan lingkungan pengembangan yang modular dan terintegrasi, sehingga memungkinkan implementasi sistem persepsi, representasi lingkungan lokal berbasis zona kedalaman, perencanaan lintasan, dan kendali secara efisien. Oleh karena itu, ROS2 dipilih sebagai *middleware* utama dalam pengembangan sistem navigasi mandiri AUV pada penelitian ini.

2.7 Pengolahan Citra Digital



Gambar 2. 5 Menampilkan nilai dari setiap pixel pada gambar

(Sumber : Dokumentasi Pribadi)

Pengolahan citra digital merupakan bidang ilmu yang mempelajari teknik dan metode untuk memproses citra dalam bentuk digital agar dapat meningkatkan kualitas citra, mengekstraksi informasi penting, serta mendukung pengambilan keputusan pada sistem berbasis komputer. Citra digital sendiri didefinisikan sebagai representasi visual dari objek atau lingkungan nyata yang tersusun atas elemen-elemen kecil yang disebut *piksel*, di mana setiap *piksel* memiliki nilai numerik yang merepresentasikan intensitas kecerahan atau warna tertentu, hal ini dapat dilihat pada Gambar 2.5. Representasi ini memungkinkan citra diproses secara matematis dan komputasional oleh sistem komputer (Dijaya & Setiawan, 2023).

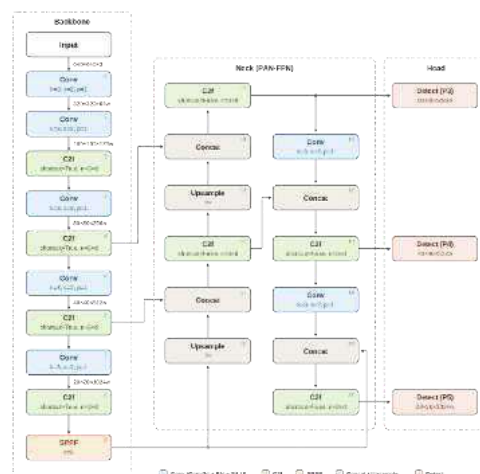
Tujuan utama dari pengolahan citra digital meliputi peningkatan kualitas citra, ekstraksi fitur, serta pengenalan objek. Dalam perkembangannya, pendekatan pengenalan objek pada citra terbagi menjadi dua kelompok besar, yaitu pendekatan konvensional dan pendekatan berbasis pembelajaran (*learning-based*). Pendekatan konvensional mengandalkan fitur yang dirancang secara manual (*hand-crafted features*), seperti ambang batas intensitas (*thresholding*), karakteristik warna, tepi, maupun bentuk objek. Pendekatan ini memiliki komputasi yang ringan, namun sangat bergantung pada parameter yang dikalibrasi untuk kondisi tertentu, sehingga

keandalannya menurun ketika kondisi pencahayaan, sudut pandang, atau jarak objek berubah.

Keterbatasan tersebut menjadi semakin signifikan pada citra bawah air, dimana atenuasi cahaya oleh medium air menyebabkan pergeseran karakteristik warna terhadap jarak dan kedalaman, hamburan partikel menurunkan kontras citra, serta kondisi pencahayaan yang tidak seragam di seluruh area operasi. Akibatnya, parameter deteksi konvensional yang telah disesuaikan pada satu kondisi kerap tidak konsisten pada kondisi lainnya, sehingga tidak andal untuk sistem yang menuntut deteksi objek secara berkelanjutan seperti navigasi otonom.

Untuk mengatasi keterbatasan tersebut, pendekatan berbasis pembelajaran mendalam (*deep learning*) dikembangkan dengan konsep mempelajari fitur objek secara otomatis dari data pelatihan, tanpa memerlukan perancangan fitur secara manual. Model *deep learning* mampu mengenali objek berdasarkan kombinasi fitur yang kompleks mencakup bentuk, tekstur, dan konteks spasial sehingga lebih tangguh terhadap variasi kondisi visual. Dalam konteks robotika dan kendaraan otonom, termasuk Autonomous Underwater Vehicle (AUV), pendekatan ini semakin banyak digunakan sebagai dasar sistem persepsi visual untuk mendeteksi rintangan dan mengenali objek secara *real-time*. Salah satu metode deteksi objek berbasis *deep learning* yang banyak diterapkan adalah You Only Look Once (YOLO), yang dijelaskan lebih lanjut pada subbab berikutnya.

2.8 You Only Look Once (YOLO)



Gambar 2. 6 Arsitektur Umum Jaringan YOLO

(Sumber : Dokumentasi Pribadi)

You Only Look Once (YOLO) merupakan metode deteksi objek berbasis *deep learning* yang dirancang dengan pendekatan satu tahap (*one-stage detector*). Berbeda dengan metode deteksi dua tahap seperti R-CNN yang terlebih dahulu menghasilkan kandidat wilayah objek (*region proposal*) sebelum melakukan klasifikasi, YOLO memproses seluruh citra dalam satu kali proses inferensi untuk langsung memprediksi posisi dan kelas seluruh objek secara bersamaan. Pendekatan ini menjadikan YOLO memiliki kecepatan inferensi yang jauh lebih tinggi dibandingkan metode dua tahap, sehingga banyak digunakan pada aplikasi yang menuntut deteksi objek secara *real-time*, termasuk pada sistem robotika dan kendaraan otonom (R. Chen & Chen, 2023).

Prinsip kerja YOLO adalah membagi citra masukan menjadi kisi-kisi (*grid*), dimana setiap sel kisi bertanggung jawab memprediksi objek yang titik pusatnya berada di dalam sel tersebut. Setiap prediksi terdiri dari tiga komponen utama, yaitu koordinat kotak pembatas (*bounding box*) yang menyatakan posisi dan dimensi objek dalam citra, kelas objek hasil klasifikasi, serta nilai keyakinan (*confidence score*) yang menyatakan tingkat kepercayaan model terhadap keberadaan dan ketepatan deteksi tersebut. Nilai *confidence* berada pada rentang 0 hingga 1, dimana nilai yang semakin tinggi menunjukkan keyakinan model yang semakin besar. Untuk menghilangkan deteksi ganda pada objek yang sama, YOLO menerapkan *Non-Maximum Suppression* (NMS) yang hanya mempertahankan *bounding box* dengan nilai *confidence* tertinggi di antara deteksi-deteksi yang saling tumpang tindih.

Secara *arsitektural*, jaringan YOLO tersusun atas tiga bagian utama. Bagian *backbone* berfungsi mengekstraksi fitur dari citra masukan melalui serangkaian lapisan konvolusi, menghasilkan representasi fitur pada berbagai tingkat kedalaman. Bagian *neck* bertugas menggabungkan fitur dari berbagai skala agar model mampu mendeteksi objek berukuran besar maupun kecil secara seimbang. Bagian *head* menghasilkan keluaran akhir berupa prediksi *bounding box*, kelas, dan *confidence* untuk setiap objek yang terdeteksi. Sejak pertama kali diperkenalkan, YOLO terus mengalami pengembangan hingga berbagai versi dengan peningkatan pada akurasi, kecepatan, dan efisiensi parameter. Pada penelitian ini digunakan YOLO versi 8 (YOLOv8) varian *nano* (YOLOv8n), yaitu varian dengan jumlah

parameter terkecil yang dirancang khusus untuk perangkat dengan sumber daya komputasi terbatas, sehingga sesuai untuk dijalankan pada komputer *embedded* Jetson Orin NX yang digunakan sebagai unit pemrosesan utama AUV.

Kinerja model YOLO sangat ditentukan oleh kualitas data pelatihan yang digunakan. Model perlu dilatih menggunakan *dataset* citra yang merepresentasikan kondisi operasional sebenarnya, mencakup variasi jarak, sudut pandang, dan kondisi pencahayaan objek target. Setiap citra pada *dataset* diberi label berupa *bounding box* dan kelas objek melalui proses anotasi (*labeling*), yang selanjutnya digunakan model untuk mempelajari karakteristik visual setiap kelas objek. Pada penelitian ini, *dataset* dikumpulkan langsung dari lingkungan pengujian menggunakan kamera yang sama dengan yang terpasang pada AUV, sehingga karakteristik citra pelatihan sesuai dengan kondisi inferensi pada saat operasi.

Untuk memenuhi kebutuhan inferensi *real-time* pada platform *embedded*, model YOLO hasil pelatihan dioptimalkan menggunakan TensorRT, yaitu pustaka optimasi inferensi *deep learning* yang memanfaatkan akselerasi GPU pada perangkat NVIDIA. TensorRT melakukan serangkaian optimasi terhadap model, antara lain penggabungan lapisan (*layer fusion*) untuk mengurangi jumlah operasi komputasi, serta kuantisasi presisi numerik dari *floating point* 32-bit menjadi presisi yang lebih rendah untuk mempercepat komputasi tanpa penurunan akurasi yang signifikan. Dengan optimasi tersebut, model YOLOv8n mampu melakukan deteksi objek secara *real-time* pada Jetson Orin NX sekaligus menyisakan sumber daya komputasi untuk proses lain seperti komputasi *stereo depth* dan perencanaan lintasan yang berjalan secara bersamaan.

2.9 Stereo Depth

Sistem estimasi jarak pada penelitian ini memanfaatkan kamera stereo sebagai sensor utama untuk memperoleh informasi kedalaman lingkungan di sekitar Autonomous Underwater Vehicle (AUV). Kamera stereo bekerja berdasarkan prinsip *stereo vision*, yaitu dengan memanfaatkan perbedaan sudut pandang dua kamera yang ditempatkan sejajar dan terpisah oleh jarak tertentu untuk mengestimasi jarak suatu objek. Perbedaan posisi objek yang terekam pada citra kamera kiri dan kamera kanan digunakan sebagai dasar dalam menghitung kedalaman atau jarak objek terhadap sistem kamera.

Secara umum, prinsip *stereo vision* memanfaatkan hubungan geometris antara posisi objek pada citra kiri dan kanan, panjang fokus kamera, serta jarak antar kamera (*baseline*). Hubungan tersebut dapat dijelaskan melalui model geometri kamera *stereo*, di mana posisi objek pada bidang citra kiri dinyatakan sebagai X_L dan pada bidang citra kanan sebagai X_R . Perbedaan posisi objek pada kedua citra tersebut menghasilkan nilai disparitas (Adi Rahmad Ramadhan et al., 2025).

Nilai disparitas yang besar menunjukkan bahwa objek berada dekat dengan kamera, sedangkan nilai disparitas yang kecil mengindikasikan objek berada pada jarak yang lebih jauh. Oleh karena itu, proses estimasi jarak sangat bergantung pada akurasi perhitungan disparitas, yang dipengaruhi oleh kualitas kalibrasi kamera, pencocokan fitur antar citra, serta kondisi lingkungan seperti pencahayaan dan kejernihan air.

Dalam implementasinya, sistem kamera stereo terlebih dahulu melalui proses kalibrasi untuk memperoleh parameter intrinsik dan ekstrinsik kamera, seperti panjang fokus, titik pusat citra, serta jarak antar kamera. Parameter-parameter ini digunakan untuk memastikan bahwa hubungan geometris antara kamera kiri dan kanan dapat dimodelkan secara akurat. Selanjutnya, dilakukan proses *stereo matching* untuk menentukan pasangan *piksel* yang merepresentasikan objek yang sama pada kedua citra, sehingga nilai disparitas dapat dihitung secara tepat.

Proses *stereo matching* pada penelitian ini dilakukan menggunakan algoritma *Semi-Global Block Matching* (SGBM). Sebelum pencocokan dilakukan, kedua citra terlebih dahulu melalui proses rektifikasi, yaitu transformasi geometris berdasarkan parameter hasil kalibrasi yang menyelaraskan kedua citra sehingga titik korespondensi antar citra berada pada baris *piksel* yang sama. Rektifikasi menyederhanakan pencarian korespondensi dari dua dimensi menjadi satu dimensi sepanjang baris, sehingga mempercepat komputasi secara signifikan. Algoritma SGBM kemudian mencari korespondensi setiap blok *piksel* dengan mempertimbangkan kehalusan disparitas antar *piksel* bertetangga, menghasilkan *disparity map* yang lebih halus dan konsisten dibandingkan metode pencocokan blok sederhana, dengan beban komputasi yang masih memadai untuk dijalankan secara *real-time* pada perangkat *embedded*.

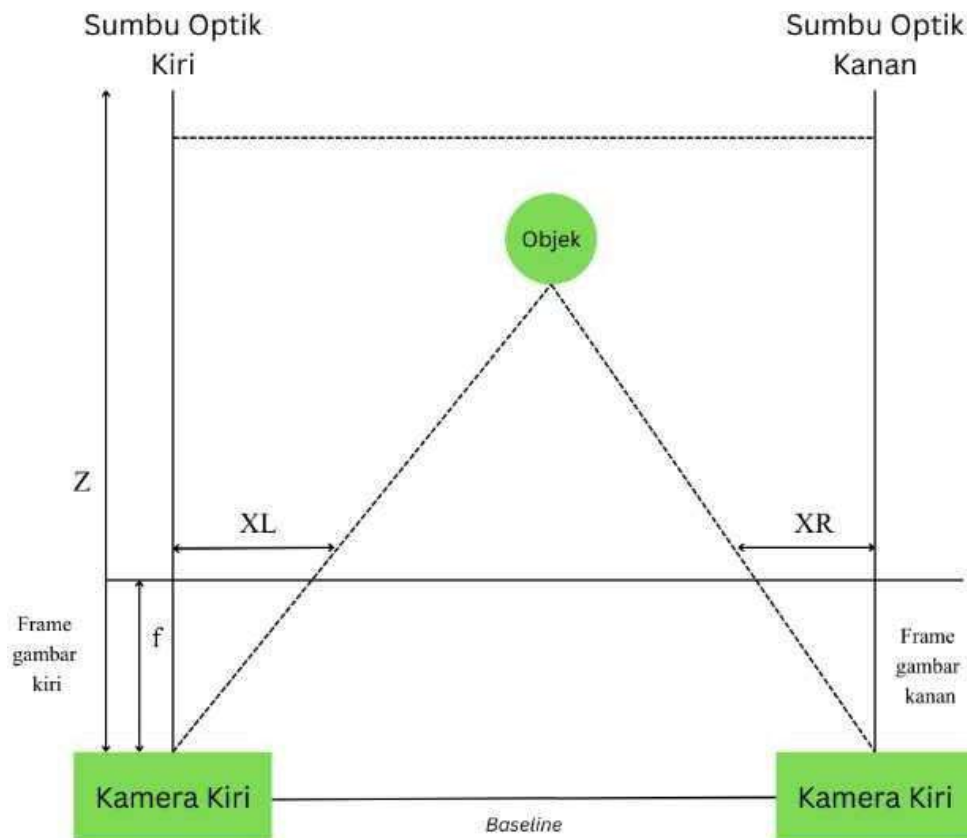
Penerapan kamera stereo pada lingkungan bawah air memiliki tantangan tambahan berupa efek refraksi cahaya. Cahaya yang merambat dari medium air menuju lensa kamera mengalami pembelokan pada antarmuka air dan *housing* kamera akibat perbedaan indeks bias kedua medium, sesuai dengan hukum *Snellius*. Efek ini menyebabkan panjang fokus efektif sistem kamera di dalam air berbeda dengan hasil kalibrasi di udara, sehingga estimasi jarak yang menggunakan parameter kalibrasi udara akan menghasilkan penyimpangan sistematis. Oleh karena itu, sistem kamera stereo yang dioperasikan di bawah air memerlukan kalibrasi tambahan atau koreksi skala yang dilakukan langsung pada kondisi terendam, agar estimasi jarak yang dihasilkan tetap akurat pada kondisi operasional sebenarnya.

Informasi kedalaman yang dihasilkan dari sistem *stereo vision* kemudian digunakan sebagai masukan utama dalam pembentukan representasi lingkungan lokal berbasis zona kedalaman dan sistem *obstacle avoidance* AUV. Dengan memanfaatkan estimasi jarak secara langsung dari data visual, AUV dapat mengenali keberadaan rintangan serta menentukan jarak relatif terhadap objek di sekitarnya secara *real-time*. Pendekatan ini memungkinkan sistem navigasi AUV untuk mengambil keputusan pergerakan yang lebih aman dan adaptif terhadap kondisi lingkungan bawah air yang dinamis.

Selain efek refraksi, ketelitian estimasi jarak pada sistem stereo juga ditentukan oleh pemilihan jarak antar kamera atau *baseline*. *Baseline* yang lebih lebar menghasilkan nilai disparitas yang lebih besar pada jarak yang sama sehingga ketelitian pengukuran meningkat, namun daerah tumpang tindih bidang pandang kedua kamera menjadi lebih sempit dan jarak minimum yang dapat diukur bertambah jauh. Sebaliknya, *baseline* yang terlalu sempit menghasilkan nilai disparitas yang kecil sehingga kesalahan sebesar satu *piksel* pada proses pencocokan berdampak besar terhadap jarak yang dihitung. Karena jarak berbanding terbalik terhadap disparitas, kesalahan estimasi jarak meningkat secara kuadratik terhadap jarak objek, sehingga sistem stereo memiliki ketelitian yang baik pada jarak dekat namun menurun dengan cepat pada jarak jauh. Karakteristik tersebut menyebabkan setiap sistem stereo memiliki rentang jarak kerja efektif yang terbatas, yang penentuannya harus disesuaikan dengan kebutuhan misi. Pada

wahana bawah air, rentang kerja tersebut umumnya dibatasi pada jarak dekat karena jarak pandang di dalam air juga terbatas oleh atenuasi cahaya dan hamburan partikel.

Hubungan antara kedalaman objek dan parameter kamera stereo secara umum dapat dinyatakan sebagai berikut:



Gambar 2. 7 Ilustrasi prinsip kerja *stereo vision* dalam estimasi jarak objek atau *stereo depth*

(Sumber : Dokumentasi Pribadi)

$$\frac{Z}{f} = \frac{X}{X_L} \quad (2.2)$$

$$\frac{Z}{f} = \frac{X - b}{X_R} \quad (2.3)$$

Sehingga diperoleh :

$$X_L = \frac{X}{Z} \times f \quad (2.4)$$

$$X_R = \frac{X - b}{Z} \times f \quad (2.5)$$

$$Disparitas = X_L - X_R \quad (2.6)$$

dengan :

- Z menyatakan kedalaman atau jarak objek terhadap kamera,
- f adalah panjang fokus kamera,
- X adalah posisi objek pada koordinat dunia,
- b merupakan jarak antar dua kamera (*baseline*),
- X_L dan X_R adalah posisi objek pada bidang citra kiri dan kanan

Dari hubungan tersebut, nilai *disparitas* (d) dapat didefinisikan sebagai selisih posisi objek pada citra kiri dan kanan, yaitu:

$$Disparitas = X_L - X_R \quad (2.7)$$

Nilai *disparitas* memiliki hubungan berbanding terbalik dengan kedalaman objek. Semakin besar nilai *disparitas* yang dihasilkan, maka objek berada semakin dekat dengan kamera. Sebaliknya, nilai *disparitas* yang kecil menunjukkan bahwa objek berada pada jarak yang lebih jauh. Prinsip ini menjadi dasar dalam perhitungan kedalaman pada sistem kamera stereo.

Keluaran utama dari sistem kamera stereo adalah *depth map*, yaitu peta kedalaman yang merepresentasikan jarak setiap *piksel* terhadap kamera dalam satuan tertentu. Informasi kedalaman ini dapat dimanfaatkan untuk mendeteksi keberadaan rintangan berdasarkan jaraknya terhadap AUV. Dengan menentukan ambang batas kedalaman tertentu, sistem dapat mengklasifikasikan area yang dianggap sebagai rintangan dan area yang aman untuk dilalui.

Untuk meningkatkan kestabilan estimasi jarak, nilai kedalaman yang digunakan dapat diambil dari nilai median pada area objek yang terdeteksi. Pendekatan ini bertujuan untuk mengurangi pengaruh *noise* dan kesalahan pengukuran pada *piksel* individu. Informasi jarak dan posisi rintangan yang diperoleh selanjutnya digunakan sebagai masukan dalam pembentukan representasi lingkungan lokal berbasis zona kedalaman dan dimanfaatkan oleh algoritma Artificial Potential Field (APF) untuk menentukan arah dan kecepatan gerak AUV secara *real-time*.

2.10 Semi-Global Block Matching (SGBM)

Semi-Global Block Matching (SGBM) merupakan algoritma *stereo matching* yang berfungsi mencari korespondensi antara *piksel* pada citra kiri dan citra kanan dari sepasang kamera stereo, sehingga diperoleh peta *disparitas*

(*disparity map*) yang selanjutnya dapat dikonversi menjadi informasi jarak. Disparitas didefinisikan sebagai selisih posisi horizontal suatu titik objek yang sama ketika diamati pada citra kiri dan citra kanan. Semakin dekat suatu objek terhadap kamera, semakin besar nilai disparitasnya, dan sebaliknya.

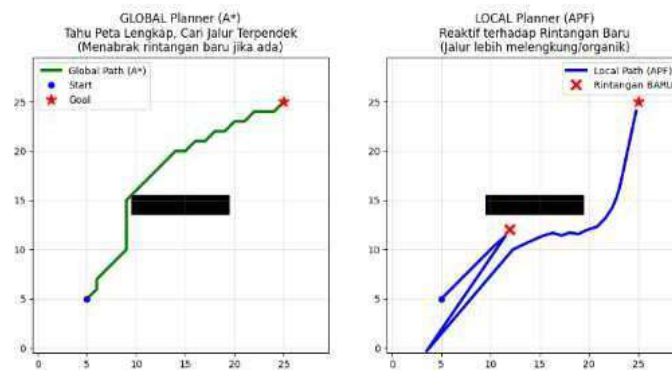
Algoritma *stereo matching* secara umum dikelompokkan menjadi dua pendekatan, yaitu metode lokal (*local method*) dan metode global (*global method*). Metode lokal, seperti *Block Matching* (BM), menentukan disparitas setiap *piksel* hanya berdasarkan informasi dari *piksel-piksel* tetangga di sekitarnya. Pendekatan ini bersifat sederhana dan cepat, namun menghasilkan peta disparitas yang kasar serta rentan terhadap kesalahan pada area bertekstur rendah dan area oklusi. Sebaliknya, metode global menentukan disparitas dengan mempertimbangkan seluruh informasi citra melalui optimasi fungsi energi secara menyeluruh. Metode global menghasilkan akurasi yang tinggi, tetapi memiliki kompleksitas komputasi yang besar sehingga sulit diterapkan pada sistem yang menuntut pemrosesan secara *real-time*.

SGBM hadir sebagai pendekatan semi-global yang menjembatani kedua metode tersebut. Algoritma ini mengaproksimasi kendala kehalusan (*smoothness constraint*) dua dimensi milik metode global dengan cara melakukan agregasi biaya sepanjang sejumlah lintasan satu dimensi dari berbagai arah. Dengan pendekatan ini, SGBM memperoleh akurasi yang mendekati metode global namun dengan beban komputasi yang jauh lebih ringan, sehingga sesuai untuk diterapkan pada sistem tertanam (*embedded system*) yang beroperasi secara *real-time* (Zhang et al., 2024).

2.11 Perencanaan Lintasan Lokal

Path planning merupakan metode atau algoritma yang digunakan untuk menentukan lintasan pergerakan suatu robot bergerak dari posisi awal menuju titik tujuan. Tujuan utama dari *path planning* adalah menghasilkan jalur yang aman dan efisien, serta dalam beberapa kasus meminimalkan jarak tempuh atau biaya tertentu sehingga lintasan yang dihasilkan bersifat optimal (Adryady et al., 2022).

Berdasarkan cakupan dan sumber informasi yang digunakan, perancangan lintasan dapat diklasifikasikan menjadi perancangan lintasan global dan perancangan lintasan lokal.



Gambar 2. 8 Perbandingan lintasan global dan lintasan lokal pada robot otonom
(Sumber : Dokumentasi Pribadi)

Gambar 2.8 menunjukkan perbedaan karakteristik antara perencanaan lintasan global dan perencanaan lintasan lokal. Perancangan lintasan global memanfaatkan peta lingkungan untuk merencanakan jalur secara menyeluruh sebelum robot bergerak, sedangkan perancangan lintasan lokal menitikberatkan pada pengambilan keputusan secara langsung berdasarkan kondisi lingkungan di sekitar robot yang diperoleh secara *real-time* melalui sensor.

Perancangan lintasan lokal bersifat reaktif terhadap perubahan lingkungan dan sangat bergantung pada data sensor yang diperoleh secara kontinu. Dalam sistem *obstacle avoidance*, perancangan lintasan lokal berperan sebagai komponen utama yang menentukan arah serta kecepatan gerak robot agar dapat menghindari rintangan yang muncul selama pergerakan menuju tujuan.

Berbagai pendekatan *path planning* lokal telah dikembangkan, mulai dari metode reaktif sederhana hingga metode berbasis gaya atau fungsi potensial, yang masing-masing memiliki kelebihan dan keterbatasan dalam hal kompleksitas komputasi, stabilitas, serta respons terhadap lingkungan. Pada AUV, perancangan lintasan lokal menghadapi tantangan tambahan akibat dinamika lingkungan bawah air, seperti arus, visibilitas terbatas, serta keterbatasan sistem navigasi yang tersedia.

Oleh karena itu, penelitian ini memfokuskan pada penggunaan perancangan lintasan lokal sebagai pendekatan utama dalam sistem *obstacle avoidance* AUV, dengan memanfaatkan informasi lingkungan yang diperoleh secara langsung untuk menghasilkan respons navigasi yang adaptif dan *real-time*.

2.12 Artificial Potential Field

Metode Artificial Potential Field (APF) adalah algoritma perencanaan jalur yang memodelkan lingkungan robot sebagai medan gaya virtual. Robot dianggap sebagai sebuah titik muatan yang bergerak di bawah pengaruh resultan gaya dari dua medan potensial utama: gaya tarik (*attractive force*) dari titik target dan gaya tolak (*repulsive force*) dari rintangan. Pergerakan robot ditentukan oleh arah resultan gaya tersebut, yang mengarahkan robot menuju titik dengan energi potensial terendah (Fan et al., 2020). Pendekatan ini dikenal karena kesederhanaan konsep dan efisiensi komputasi yang tinggi, sehingga banyak digunakan dalam sistem navigasi robot secara *real-time*. Namun, metode APF juga memiliki keterbatasan, seperti kemungkinan terjebaknya robot pada titik minimum lokal, yang dapat menghambat pencapaian tujuan akhir.

2.10.1 Fungsi Medan Potensial Tarik Menarik

Posisi AUV saat ini dinyatakan dalam koordinat $X = (x, y)^T$, sedangkan posisi tujuan direpresentasikan sebagai $X_g = (x_g, y_g)^T$. Selama proses pergerakan menuju tujuan, AUV dipengaruhi oleh gaya tarik (*attractive force*) yang arahnya menuju titik tujuan tersebut. Besarnya gaya tarik bergantung pada jarak antara posisi AUV dan titik tujuan, di mana gaya tarik akan meningkat seiring bertambahnya jarak. Sebaliknya, gaya tarik akan bernilai nol ketika AUV telah mencapai titik tujuan. Medan potensial tarik U_{att} , yang dihasilkan oleh keberadaan titik tujuan, dinyatakan sebagai:

$$U_{att}(X) = \frac{1}{2} k_{att} \cdot d^2(X, X_g), \quad (2.8)$$

Pada persamaan tersebut, $d(X, X_g) = \|X_g - X\|$ menyatakan jarak *Euclidean* antara posisi AUV dan titik tujuan, sedangkan k_{att} merupakan konstanta yang mengatur besarnya medan potensial tarik.

Gaya tarik $F_{att}(X)$ yang bekerja pada AUV dalam medan potensial tarik diperoleh dari gradien negatif fungsi potensial tarik $U_{att}(X)$, yang dirumuskan sebagai berikut:

$$F_{att}(X) = -\nabla U_{att}(X) = k_{att} \cdot d(X, X_g) \quad (2.9)$$

Arah gaya tarik sejalan dengan garis yang menghubungkan posisi AUV dan titik tujuan, serta selalu berorientasi menuju titik tujuan tersebut.

2.10.2 Fungsi Medan Potensial Tolak-menolak

Serupa dengan fungsi medan potensial tarik, fungsi medan potensial tolak juga bergantung pada jarak antara AUV dan rintangan. Ketika AUV berada dalam wilayah pengaruh suatu rintangan, wahana tersebut akan mengalami gaya tolak. Besarnya gaya tolak meningkat seiring dengan berkurangnya jarak antara AUV dan rintangan, sehingga gaya maksimum terjadi saat AUV berada sangat dekat dengan rintangan. Posisi rintangan dinyatakan dalam bentuk $X_o = (x_o, y_o)^T$, sedangkan medan potensial tolak $U_{rep}(X)$ ditentukan berdasarkan jarak relatif antara posisi AUV dan rintangan, yang dirumuskan sebagai:

$$U_{rep}(X) = \begin{cases} \frac{1}{2} k_{rep} \left(\frac{1}{d(X, X_o)} - \frac{1}{d_0} \right)^2, & d(X, X_o) \leq d_0, \\ 0, & d(X, X_o) > d_0, \end{cases} \quad (2.10)$$

Pada persamaan tersebut, $d(X, X_o) = \|X_o - X\|$ menyatakan jarak *Euclidean* antara posisi AUV dan rintangan, sedangkan k_{rep} merupakan konstanta yang mengatur intensitas medan potensial tolak. Parameter d_0 merepresentasikan batas jangkauan pengaruh medan potensial tolak, di mana gaya tolak hanya bekerja ketika AUV berada dalam jarak tersebut.

Gaya tolak $F_{rep}(X)$ yang dialami AUV dalam medan potensial tolak diperoleh dari gradien negatif fungsi potensial tolak $U_{rep}(X)$, yang dinyatakan sebagai:

$$F_{rep}(X) = -\nabla U_{rep}(X) = \begin{cases} k_{rep} \left(\frac{1}{d(X, X_o)} - \frac{1}{d_0} \right) \frac{\nabla d(X, X_o)}{d^2(X, X_o)}, & d(X, X_o) \leq d_0, \\ 0, & d(X, X_o) > d_0, \end{cases} \quad (2.11)$$

Arah gaya tolak sejalan dengan garis yang menghubungkan posisi rintangan dan posisi AUV, serta selalu berorientasi menjauhi rintangan tersebut.

2.10.3 Fungsi Medan Potensial Resultan

Selama bergerak menuju titik tujuan, AUV dipengaruhi oleh aksi gabungan dari medan potensial tarik dan medan potensial tolak secara bersamaan. Fungsi

medan potensial resultan $U(X)$ yang bekerja pada AUV dinyatakan sebagai penjumlahan kedua medan potensial tersebut:

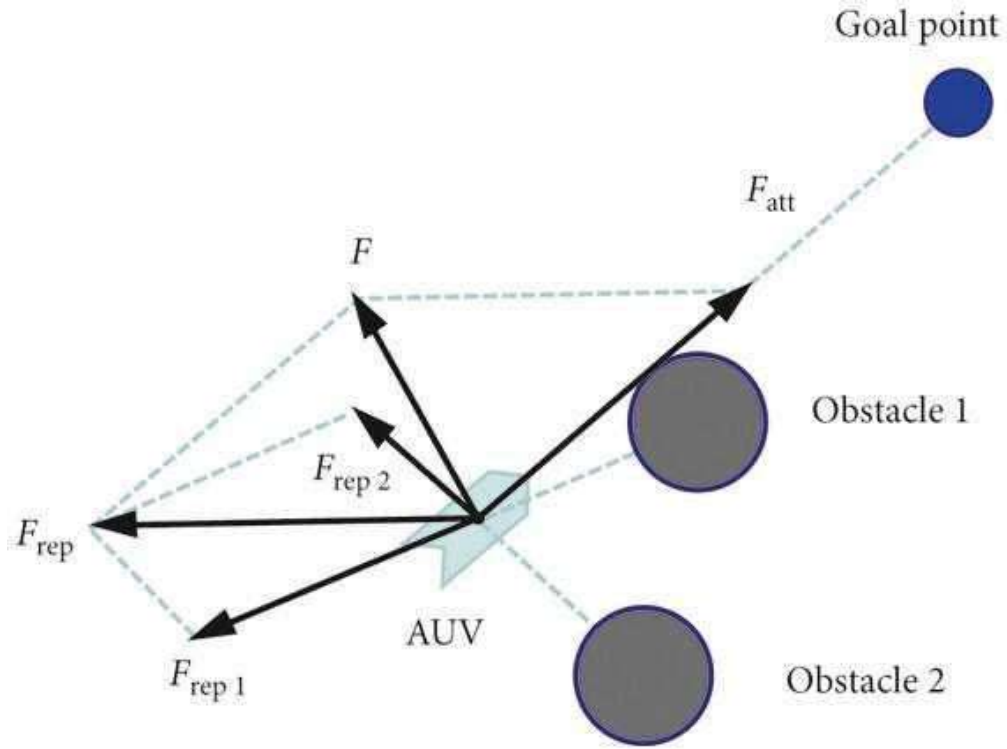
$$U(X) = U_{att}(X) + U_{rep}(X) \quad (2.12)$$

Gaya resultan $F(X)$ yang bekerja pada AUV diperoleh dari gradien negatif fungsi medan potensial resultan, yang dirumuskan sebagai:

$$F(X) = -\nabla U(X) = -\nabla U_{att}(X) - \nabla U_{rep}(X) = F_{att}(X) + F_{rep}(X) \quad (2.13)$$

Pembentukan model gaya resultan pada ruang dua dimensi ditunjukkan pada Gambar 2.9. Apabila terdapat lebih dari satu rintangan dalam wilayah pengaruh, setiap rintangan menghasilkan gaya tolaknya masing-masing (F_{rep1} , F_{rep2} , dan seterusnya) dan gaya tolak total merupakan penjumlahan vektor dari seluruh gaya tolak tersebut. Gaya resultan F selanjutnya diperoleh melalui prinsip superposisi antara gaya tarik yang dihasilkan titik tujuan dan gaya tolak total dari seluruh rintangan (Fan et al., 2020). Arah dan besar gaya resultan inilah yang menentukan arah serta kecepatan gerak AUV pada setiap siklus komputasi, sehingga AUV dapat bergerak menuju tujuan sekaligus menghindari rintangan secara simultan. Secara teknis, vektor gaya resultan ini tidak serta-merta diaplikasikan langsung sebagai gaya dorong fisik, melainkan dikonversi terlebih dahulu menjadi parameter perintah aktuator, seperti sudut haluan (*heading angle*) dan kecepatan referensi (*surge velocity*). Proses transformasi ini sangat esensial karena AUV terikat oleh batasan dinamis dan inersia yang membuatnya tidak bisa mengubah arah secara instan. Di samping itu, meskipun pendekatan superposisi gaya ini unggul dalam komputasi *real-time*, sistem navigasi tetap perlu mempertimbangkan potensi terjadinya *local minima*. Kondisi ini muncul ketika gaya tarik dan gaya tolak saling meniadakan, menghasilkan gaya resultan semu nol sebelum AUV mencapai tujuan akhir, sehingga sering kali memerlukan algoritma penyesuaian tambahan agar wahana dapat terlepas dari titik jebakan tersebut. Besarnya pengaruh gaya tolak terhadap lintasan juga bergantung pada pemilihan konstanta penguat gaya tolak, dimana nilai yang terlalu besar menyebabkan wahana bereaksi berlebihan sehingga lintasan menjadi berosilasi, sedangkan nilai yang terlalu kecil membuat jarak bebas

terhadap rintangan tidak memadai. Oleh sebab itu, penentuan konstanta penguat beserta radius pengaruh umumnya dilakukan secara empiris dengan menyesuaikan dimensi wahana dan lebar ruang gerak yang tersedia pada lingkungan operasi.



Gambar 2. 9 Model gaya total AUV

(Sumber : (Fan et al., 2020))

2.10.4 Masalah *Local Minimum*

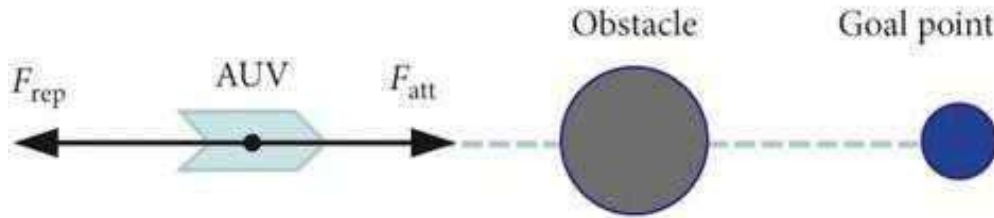
Masalah *local minima* merupakan hambatan utama dalam penggunaan metode Artificial Potential Field (APF). Kondisi ini terjadi ketika resultan gaya yang bekerja pada robot bernilai nol, namun robot belum mencapai titik target.

Secara matematis, *local minimum* terjadi pada titik koordinat di mana gaya atraksi (F_{att}) dan gaya repulsi (F_{rep}) memiliki besar yang hampir sama namun berada dalam arah yang berlawanan (kolinier). Hal ini dirumuskan sebagai berikut:

$$F(X) = F_{att}(X) + F_{rep}(X) \approx 0 \quad (2.14)$$

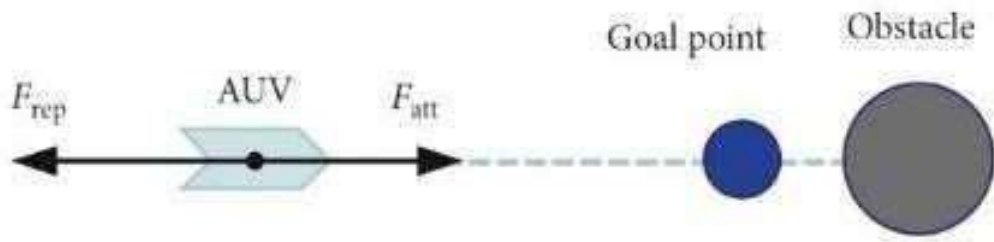
Berdasarkan penelitian (Fan et al., 2020), terdapat tiga skenario umum penyebab *local minimum*:

1. Rintangan *Kolinier* (Antara): Rintangan berada tepat di antara robot dan target, menyebabkan gaya dorong ke belakang meniadakan gaya tarik ke depan.



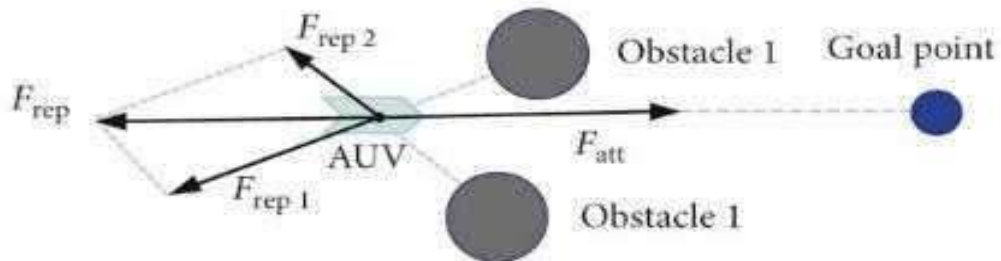
Gambar 2. 10 Masalah *local minimum* dengan tiga titik segaris pada metode APF
(Sumber : (Fan et al., 2020))

2. Rintangan *Kolinier* (Belakang): Target berada di antara robot dan rintangan, di mana gaya tarik ke target diimbangi oleh gaya tolak rintangan yang sangat kuat.



Gambar 2. 11 Masalah *local minimum* dengan tiga titik segaris
(Sumber : (Fan et al., 2020))

3. Hambatan Majemuk atau Cekung: Robot terjebak dalam area yang dikelilingi banyak rintangan (seperti bentuk-U), sehingga gaya-gaya repulsi dari berbagai sisi mengunci posisi robot.



Gambar 2. 12 Daerah *local minimum* dengan banyak rintangan
(Sumber : (Fan et al., 2020))

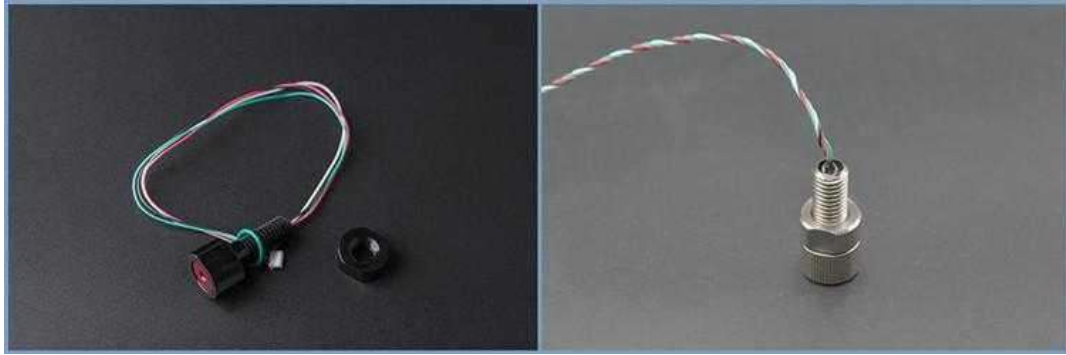
Dalam konteks penelitian ini, risiko *local minimum* diminimalkan oleh karakteristik lingkungan operasional yang terstruktur. Arena KRBAI memiliki konfigurasi rintangan yang telah diketahui, yaitu rintangan tunggal berupa suar yang tidak membentuk formasi cekung maupun kolinier majemuk, sehingga skenario penyebab *local minimum* sebagaimana dijelaskan sebelumnya tidak terbentuk pada lintasan misi. Selain itu, target navigasi AUV diperbarui secara bertahap mengikuti urutan misi, sehingga konfigurasi gaya tarik dan gaya tolak berubah secara dinamis dan tidak menghasilkan titik kesetimbangan yang menetap. Dengan pertimbangan tersebut, metode APF klasik tetap dapat diterapkan secara andal tanpa memerlukan modifikasi khusus penanganan *local minimum*.

2.13 Sensor dan Aktuator

Sensor dan aktuator merupakan komponen utama dalam sistem Autonomous Underwater Vehicle (AUV) yang berperan sebagai penghubung antara lingkungan dan sistem kendali. Sensor digunakan untuk memperoleh informasi kondisi lingkungan dan keadaan AUV, seperti orientasi, kedalaman, dan kondisi sekitar, yang selanjutnya diproses oleh sistem kendali. Sementara itu, aktuator berfungsi untuk mengeksekusi perintah kendali dalam bentuk gerakan atau aksi fisik, seperti penggerakan *thruster* dan mekanisme misi. Integrasi sensor dan aktuator yang baik memungkinkan AUV merespons lingkungan secara akurat dan menjalankan navigasi serta *obstacle avoidance* secara efektif.

2.11.1 Sensor Pressure

Sensor tekanan (*pressure*) MS5837 digunakan untuk mengukur kedalaman AUV secara presisi. Sensor ini bekerja dengan mengukur tekanan hidrostatik air dan mengonversinya menjadi data kedalaman melalui antarmuka I2C yang telah dikalibrasi secara pabrik (Rovmaker, 2023a). Pada sistem ini, MS5837 memberikan umpan balik (*feedback*) kedalaman secara *real-time* kepada sistem kendali PID di Jetson Orin NX agar AUV dapat mempertahankan posisi vertikalnya secara stabil saat bermanuver menghindari rintangan. Keandalan data kedalaman yang dihasilkan oleh sensor ini sangat krusial untuk mencegah terjadinya osilasi vertikal yang dapat mengganggu kualitas persepsi visual dari sistem kamera. Bentuk fisik dari sensor tekanan MS5837 ditunjukkan pada Gambar 2.13, sedangkan rincian spesifikasi teknisnya disajikan pada Tabel 2.3.



Gambar 2. 13 Sensor *Pressure* Rovmaker MS5837

(Sumber : (Rovmaker, 2023a))

Tabel 2. 3 Spesifikasi Sensor *Pressure* MS5837

Spesifikasi	Keterangan
Pressure Sensor Type	MS5837-30BA
Working Voltage	1.5-3.6V
Refresh Rate	0.5ms
Communication	Inter-Integrated Circuit (I2C)
Measuring range	0-30bar
Working Temperature	(-20)-85°C
Accuracy	Resolution 0.2mbar
Material	Ceramics

(Sumber : Dokumen Pribadi)

2.11.2 Sensor IMU

Inertial Measurement Unit (IMU) HWT905 adalah sensor orientasi yang digunakan untuk memantau perubahan sikap (sudut *roll*, *pitch*, dan *yaw*) AUV secara akurat dan stabil (Dai & Jing, 2021). Pada penelitian ini, magnetometer bawaan dinonaktifkan untuk menghindari interferensi medan magnet dari putaran delapan motor thruster, sehingga orientasi hanya dihitung menggunakan algoritma sensor *fusion* 6-axis (*gyroscope* dan *accelerometer*). Data sudut ini dikirimkan melalui komunikasi UART ke unit utama untuk menjaga kestabilan arah hadap (*heading*) AUV. Keakuratan pembacaan orientasi ini sangat krusial mengingat AUV beroperasi di lingkungan bawah air yang dinamis dan rentan terhadap gangguan *hidrodinamika*. Oleh karena itu, sinyal dari *gyroscope* dan *accelerometer* diproses secara terpadu untuk meredam noise maupun getaran mekanis dari rangka wahana. Informasi orientasi yang presisi tersebut selanjutnya digunakan sebagai umpan balik (*feedback*) utama pada sistem kendali *Proporsional-Integral-Derivatif* (PID). Dengan adanya umpan balik yang stabil, sistem dapat mengeksekusi

manuver pergerakan hasil kalkulasi algoritma Artificial Potential Field (APF) tanpa menyebabkan AUV kehilangan keseimbangan. Visualisasi fisik dari sensor IMU HWT905 ditunjukkan pada Gambar 2.14, dengan spesifikasi teknis lengkap yang diuraikan pada Tabel 2.4.



Gambar 2. 14 Sensor IMU Witmotion HWT905

(Sumber : (WitMotion, 2023))

Tabel 2. 4 Spesifikasi Sensor IMU Witmotion HWT905

Spesifikasi	Keterangan
Operating Voltage	5-36V
Operating Current	<40mA
Data Output	3-axis (Acceleration, Gyro, Angle, Magnetic field) Quaternion, Chip time
Sensing Range	Acceleration : $\pm 16g$ Gyro : $\pm 2000^\circ/s$ Angle : X Z $\pm 180^\circ$, Y $\pm 90^\circ$
Angle Accuracy	X, Y axis static 0.05° dynamic 0.1° Z axis : 1° (after calibrated)
Magnetic Field	1mg (milligauss)
Return Rate	0.2-200Hz (adjustable)
Baud Rate	4800-921600bps
Protocol	TTL: UART, RS485: ModBusRTU, RS232: Witmotion-Protocol
Dimensions	55 x 36.8 x 24mm

(Sumber : Dokumentasi Pribadi)

2.11.3 Magnetic Switch

Magnetic switch MC-38 diaplikasikan sebagai saklar nirsentuh berbasis medan magnet yang mendeteksi kondisi buka atau tutup (Amini et al., 2021). Komponen ini dihubungkan langsung ke mikrokontroler ESP32 untuk berfungsi ganda: sebagai pemicu (trigger) untuk mengaktifkan mode navigasi otonom, serta sebagai mekanisme keamanan darurat (kill switch). Penggunaan saklar magnetik ini memungkinkan operator mengontrol AUV dari luar tanpa perlu melubangi

badan wahana, sehingga integritas kedap air (waterproofing) tetap terjaga. Bentuk fisik *magnetic switch* MC-38 yang digunakan pada sistem ini dapat dilihat pada Gambar 2.15, beserta spesifikasi teknisnya pada Tabel 2.5.



Gambar 2. 15 *Magnetic Switch* MC-38

(Sumber : (Amini et al., 2021))

Tabel 2. 5 Spesifikasi *Magnetic Switch* MC-38

Spesifikasi	Keterangan
Rared current	100mA
Rated Voltage	200V DC
Operating Distance	15mm-25mm
Resistance	$\pm 4\Omega$
Dimension	28x15x0.9cm

(Sumber : Dokumentasi Pribadi)

2.11.4 Kamera DWE *exploreHD*

Kamera DWE *exploreHD* merupakan kamera USB yang dirancang khusus untuk lingkungan bawah air dengan *housing* kedap air terintegrasi, meminimalisir distorsi optik ganda (Exploration, 2023). Pada penelitian ini, dua unit kamera tersebut dipasang sejajar di bagian depan wahana untuk membentuk sistem *stereo vision*. Citra yang ditangkap oleh lensa *fisheye* berbidang pandang lebar ini kemudian diproses oleh Jetson Orin NX untuk komputasi peta kedalaman (*depth map*) menggunakan algoritma SGBM dan deteksi objek target berbasis YOLO. Penggunaan lensa sudut lebar ini memberikan keuntungan strategis karena memungkinkan sistem persepsi menangkap area observasi yang lebih komprehensif dalam satu bingkai (*frame*) tunggal. Selain itu, kualitas transmisi citra yang dihasilkan tetap stabil terhadap efek atenuasi cahaya di dalam air, sehingga sangat mendukung keandalan proses ekstraksi fitur secara *real-time*. Desain fisik kamera

DWE exploreHD yang telah terintegrasi dengan housing kedap airnya ditunjukkan pada Gambar 2.16.



Gambar 2. 16 Kamera DWE exploreHD

(Sumber : (Exploration, 2023))

2.11.5 Motor Thruster BLDC

Pada Thruster T200 dari ROVMAKER adalah aktuator utama pendorong AUV yang menggunakan motor brushless DC (BLDC) berdesain fully flooded (membiarkan air masuk sebagai pendingin) yang cocok untuk ruang terbatas (Rovmaker, 2023b). Wahana ini menggunakan delapan unit thruster, dengan konfigurasi empat thruster vertikal untuk kendali kedalaman dan empat thruster horizontal untuk translasi serta rotasi. Masing-masing motor dikendalikan melalui sinyal PWM dari Electronic Speed Controller (ESC). Visualisasi fisik motor thruster T200 ditunjukkan pada Gambar 2.17, dan spesifikasi teknis pendorong ini dirangkum pada Tabel 2.6.



Gambar 2. 17 Motor *Thruster* T200

(Sumber : (Rovmaker, 2023b))

Tabel 2. 6 Spesifikasi Motor Thruster T200

Parameter	Spesifikasi	Keterangan
Motor	Brushless DC 3-phase	Motor tanpa sikat, efisien & tahan lama
Tegangan Operasi	6 – 20 V DC	Rentang umum tegangan kerja
Gaya Dorong Maksimum	± 5.1 kgf @ 16 V	Gaya dorong puncak (± 51 N)
Diameter Propeller	± 76 mm	Diameter baling-baling
Diameter Tubuh	± 100 mm	Dimensi badan utama
Pajang Thruster	± 113 mm	Total panjang unit
Kecepatan Maksimum	± 3500 RPM (tergantung ESC)	Kecepatan putaran propeller
Konsumsi Arus	$\pm 6 - 15$ A (tergantung beban)	Rentang arus tipikal
Berat (tanpa kabel)	$\pm 162 - 180$ g	Berat unit saja

(Sumber : Dokumentasi Pribadi)

2.14 Kontroller

Kontroller merupakan bagian inti dalam sistem Autonomous Underwater Vehicle (AUV) yang berfungsi untuk mengolah data dari sensor, mengambil keputusan, serta menghasilkan sinyal kendali bagi aktuator. Kontroller berperan sebagai penghubung antara sistem persepsi, perencanaan, dan aktuasi sehingga seluruh subsistem dapat bekerja secara terkoordinasi. Dalam sistem AUV, kontroller umumnya dibagi menjadi pengendali tingkat tinggi dan tingkat rendah untuk meningkatkan efisiensi pemrosesan, keandalan sistem, serta fleksibilitas pengoperasian, baik pada mode otomatis maupun manual selama proses pengujian.

2.13.1 NVIDIA Jetson Orin NX

NVIDIA Jetson Orin NX bertindak sebagai otak komputasi utama (*high-level controller*) pada AUV. *System-on-Module* (SoM) berperforma tinggi ini menjalankan sistem operasi Ubuntu 22.04 dan *middleware ROS2 Humble* secara lebih efisien dibandingkan pemrosesan murni berbasis CPU atau GPU konvensional (NVIDIA & Kit, 2023). Jetson memikul seluruh beban pemrosesan berat secara *real-time*, yang meliputi akuisisi kamera, komputasi *stereo matching* SGBM, inferensi YOLO yang diakselerasi *TensorRT* (Rey et al., 2025), pemodelan lingkungan, hingga eksekusi algoritma Artificial Potential Field (APF) dan PID navigasi. Bentuk fisik modul komputasi NVIDIA Jetson Orin NX 16 GB

ditunjukkan pada Gambar 2.18, sementara spesifikasi teknisnya disajikan pada Tabel 2.7.



Gambar 2. 18 Jetson Orin NX 16 GB

(Sumber : (NVIDIA & Kit, 2023))

Tabel 2. 7 Spesifikasi Jetson Orin NX 16GB

Spesifikasi	Keterangan
CPU	8-core Arm® Cortex®-A78AE v8.2 64-bit CPU 2MB L2 + 4MB L3, up to 2 GHz
Memory size	16 GB
Memory Types	16GB 128-bit LPDDR5
Memory Speed	102.4GB/s
Graphics Processing Unit	1024-core NVIDIA Ampere architecture GPU with 32 Tensor Cores
DC Input Voltage Supported	5-20 VDC
USB Port	3x USB 3.2 Gen2 (10 Gbps) 3x USB 2.0
Dimension	92,6 x 99,8 x 28 (mm) (LxWxH)
Weight	Net 300gr

(Sumber : Dokumentasi Pribadi)

2.13.2 ESP32

ESP32 merupakan mikrokontroler yang difungsikan sebagai pengendali tingkat rendah (*low-level controller*) dengan konsumsi daya yang relatif rendah (Kurnia AR, 2025). Modul ini bertugas menjembatani perintah dari Jetson Orin NX dengan aktuator. ESP32 menerima hasil kalkulasi kinematika (berupa nilai PWM) via komunikasi serial, lalu mendistribusikannya ke delapan ESC. Di samping itu, pemisahan tugas ini memungkinkan ESP32 untuk memantau status *magnetic switch* (sebagai *kill switch*) secara independen, sehingga fungsi keselamatan tetap bekerja

secara instan tanpa bergantung pada beban komputasi di Jetson. Modul mikrokontroler ESP32 yang difungsikan sebagai *low-level controller* ini ditunjukkan pada Gambar 2.19, beserta detail spesifikasinya pada Tabel 2.8.



Gambar 2. 19 ESP32

(Sumber : Dokumentasi Pribadi)

Tabel 2. 8 Spesifikasi ESP32

Spesifikasi	Keterangan
Produsen	Espressif Systems
Arsitektur CPU	Tensilica Xtensa LX6 32-bit
Jumlah Core	Dual-core
Frekuensi Clock	Hingga 240 Mhz
Memori ROM	448 KB
Memori SRAM	520 KB
RTC Fast Memory	8 KB
RTC Slow Memory	8 KB
Konektivitas Nirkabel	Wi-Fi 802.11 b/g/n (2.4 GHz), Bluetooth v4.2 BR/EDR
GPIO	Hingga 30 pin
Antarmuka	UART, SPI, I ² C, I ² S, CAN, PWM
ADC	12-bit ADC
DAC	2 kanal DAC
Tegangan Operasi	3.0 – 3.6 V

(Sumber : Dokumentasi Pribadi)

2.13.3 Electronic Speed Controller

Electronic Speed Controller (ESC) digunakan untuk menerjemahkan sinyal PWM dari ESP32 menjadi daya penggerak tiga fase untuk mengatur kecepatan motor BLDC pada *thruster* secara responsif (Sai Prasad Reddy et al., 2025). ESC yang digunakan mendukung fitur *bidirectional control*, sehingga *thruster* dapat berputar secara cepat baik ke arah maju (*forward*) maupun mundur (*reverse*) yang sangat esensial untuk kelincahan manuver *obstacle avoidance*. Bentuk fisik *Electronic Speed Controller* (ESC) yang digunakan untuk mengontrol *thruster* ditunjukkan pada Gambar 2.20, dengan rincian spesifikasi pada Tabel 2.9.



Gambar 2. 20 *Electronic Speed Controller (ESC)*

(Sumber : (RovMaker, 2023))

Tabel 2. 9 Spesifikasi *Electronic Speed Controller (ESC)*

Spesifikasi	Keterangan
Tegangan Input	DC 6V – 26V
Arus Maksimum	30A
Jenis Motor	Brushless DC (BLDC)
Sinyal Input	PWM 1–2 ms (standar servo)
Firmware	SimonK
Mode Kontrol	Bidirectional
Proteksi	Overcurrent, Overheat, Low Voltage
Aplikasi	ROV, AUV, Robot Air, Drone
Ukuran	Kompak ($\pm 30 \times 20$ mm, tergantung seri)
Tahan Air	Ya

(Sumber : Dokumentasi Pribadi)

2.15 Baterai *Li-ion*

Sistem kelistrikan AUV ditenagai oleh susunan sel baterai *Lithium-Ion* (Li-ion) tipe SONY VTC6 18650, yang lebih ramah lingkungan dan stabil dibandingkan baterai konvensional (Nursusanto et al., 2022). Baterai dikonfigurasi menjadi dua sistem terpisah untuk menghindari fluktuasi tegangan: Konfigurasi 4-sel (16,8 V) dialokasikan secara khusus untuk menyuplai beban tinggi pada kedelapan *thruster* dan ESC, sementara konfigurasi 3-sel (12,6 V) digunakan untuk menghidupkan komponen elektronik presisi dan modul komputasi seperti Jetson Orin NX dan ESP32 melalui regulator tegangan (*buck converter*). Bentuk fisik sel baterai SONY VTC6 18650 ditunjukkan pada Gambar 2.21, sedangkan rincian distribusi daya ke masing-masing komponen serta spesifikasi teknis baterai disajikan pada tabel spesifikasi terkait dengan rincian spesifikasi pada Tabel 2.10.



Gambar 2. 21 Baterai Li-ion Sony VCT6 18650

(Sumber : Dokumentasi Pribadi)

Tabel 2. 10 Spesifikasi Baterai *Li-ion* Sony VCT6

Spesifikasi	Keterangan
Tipe Baterai	Lithium-Ion (Li-ion)
Form Factor	18650
Tegangan Nominal	3,7 V
Tegangan Maksimum	4,2 V
Tegangan Cut-off	2,5 V
Kapasitas Nominal	3000 mAh
Energi Nominal	$\pm 11,1$ Wh
Arus Discharge Maksimum	15 A (continuous)

(Sumber : Dokumentasi Pribadi)

2.16 Gazebo

Gazebo merupakan perangkat lunak simulasi robotika berbasis fisika yang terintegrasi secara langsung dengan ROS2 untuk memodelkan dinamika fisik wahana dan lingkungannya secara realistis. Pada penelitian ini, Gazebo dimanfaatkan sebagai platform pengujian awal (preliminary testing) guna memvalidasi algoritma navigasi dan obstacle avoidance sebelum diimplementasikan secara langsung pada perangkat keras AUV yang sesungguhnya. Lingkungan virtual dirancang secara spesifik menyerupai lintasan arena KRBAI, di mana data dari sensor virtual dipublikasikan ke ROS2 untuk mengevaluasi kinerja integrasi persepsi visual dan Artificial Potential Field (APF) secara aman, efisien, dan tanpa risiko kerusakan perangkat fisik. Logo perangkat lunak simulasi Gazebo yang digunakan ditunjukkan pada Gambar 2.22.



Gambar 2. 22 Logo Gazebo

(Sumber : https://gazebosim.org/docs/latest/ros2_integration/)

2.17 Komunikasi

Sistem komunikasi pada AUV dirancang menggunakan arsitektur terdistribusi untuk menjembatani unit pemrosesan tingkat tinggi (NVIDIA Jetson Orin NX) dengan unit kendali tingkat rendah (ESP32). Pertukaran data antar kedua unit dilakukan menggunakan antarmuka serial Universal Asynchronous Receiver Transmitter (UART) melalui modul USB to TTL, yang dipilih karena keandalannya serta latensinya yang sangat rendah untuk pengiriman perintah real-time. Secara operasional, Jetson Orin NX memproses seluruh beban komputasi navigasi ROS2 hingga menghasilkan nilai PWM, yang kemudian dikemas oleh node antarmuka (`serial_bridge_node`) menjadi paket data serial. Di sisi lain, ESP32 menerima paket tersebut untuk mengeksekusi pergerakan aktuator, sembari memantau status kill switch dan mengirimkan data umpan balik (feedback) sistem kembali ke unit utama secara terus-menerus.

(Halaman ini sengaja dikosongkan)

BAB 3

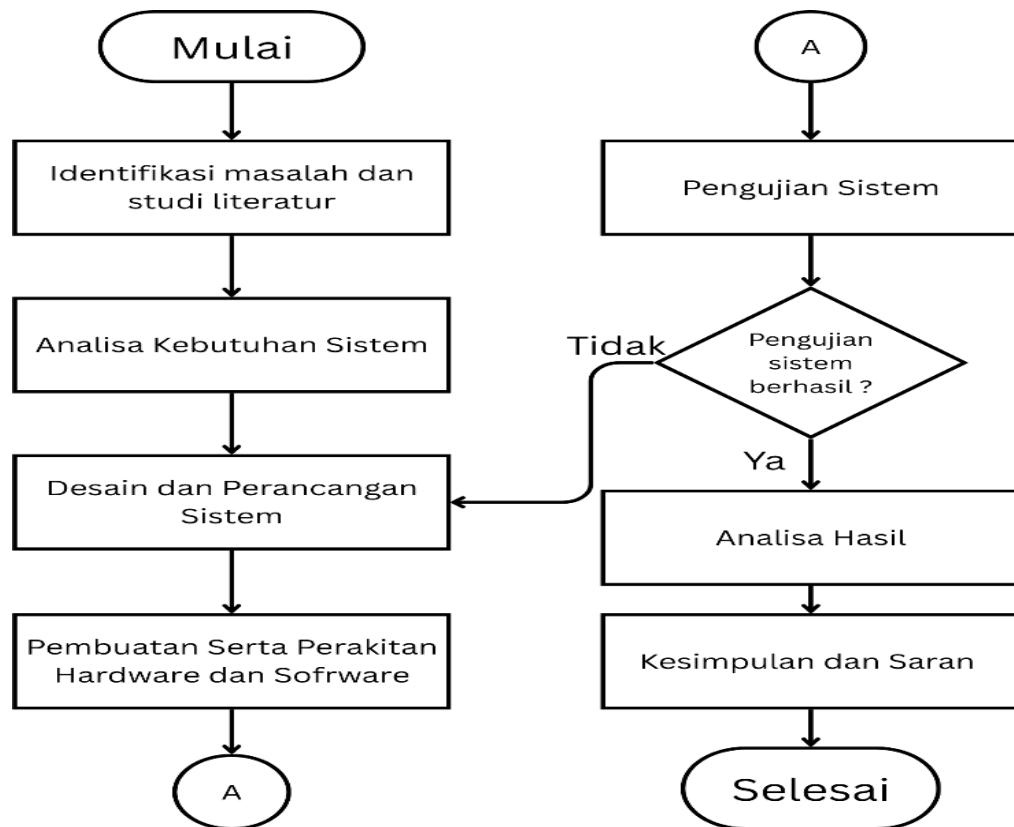
METODE PENELITIAN

Bab ini menjelaskan metode penelitian yang digunakan dalam perancangan dan pengembangan sistem Autonomous Underwater Vehicle (AUV) dengan kemampuan navigasi mandiri dan *obstacle avoidance*. Pembahasan meliputi alur dan tahapan penelitian yang disusun secara sistematis, mulai dari identifikasi masalah, studi literatur, perancangan sistem, hingga tahap implementasi dan pengujian. Metode penelitian ini dirancang untuk memastikan proses pengembangan sistem berjalan secara terstruktur, terukur, dan dapat direplikasi.

Selain itu, Bab III juga membahas perancangan dan integrasi sistem AUV yang meliputi aspek mekanik, elektrik, dan perangkat lunak. Pada bagian ini dijelaskan konsep sistem, arsitektur berbasis Robot Operating System 2 (ROS2), pemodelan lingkungan (representasi lingkungan lokal berbasis zona kedalaman), serta *perencanaan lintasan* dan *obstacle avoidance* menggunakan algoritma Artificial Potential Field (APF). Tahap pengujian dan evaluasi dilakukan untuk menilai kinerja sistem dalam merespons rintangan secara adaptif dan *real-time* sesuai dengan skenario pengujian yang dirancang.

3.1 Alur dan Tahapan Penelitian

Penelitian ini dilaksanakan melalui tahapan yang disusun secara sistematis untuk memastikan proses pengembangan sistem AUV berjalan terstruktur dan terukur. Alur penelitian secara keseluruhan ditunjukkan pada Gambar 3.1. Tahapan penelitian dimulai dari identifikasi masalah, studi literatur, perancangan sistem, implementasi, hingga pengujian dan evaluasi. Setiap tahapan saling berkaitan dan hasil dari satu tahapan menjadi dasar pelaksanaan tahapan berikutnya, sehingga proses pengembangan sistem dapat dilakukan secara terencana dan efisien. Apabila hasil pengujian belum memenuhi kriteria keberhasilan yang ditetapkan, proses dikembalikan ke tahap perancangan untuk dilakukan perbaikan sebelum pengujian diulang kembali. Melalui pendekatan ini, diharapkan sistem AUV yang dikembangkan dapat memenuhi kebutuhan navigasi mandiri dan *obstacle avoidance* pada arena kompetisi KRBAI secara optimal.



Gambar 3. 1 *Flowchart Tahapan Penelitian*

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 3.1, proses penelitian diawali dengan identifikasi masalah, yaitu merumuskan permasalahan utama terkait kebutuhan sistem navigasi mandiri dan *obstacle avoidance* pada AUV di lingkungan bawah air. Selanjutnya dilakukan studi literatur dengan mengkaji jurnal ilmiah, buku referensi, dan penelitian terdahulu yang berkaitan dengan navigasi AUV, persepsi visual berbasis kamera stereo dan deteksi objek *deep learning*, serta algoritma perencanaan lintasan lokal. Hasil studi literatur digunakan sebagai dasar dalam pemilihan metode dan perancangan sistem.

Tahap berikutnya adalah analisis kebutuhan dan perancangan sistem, yang mencakup penentuan spesifikasi perangkat keras dan perangkat lunak serta perancangan arsitektur sistem secara keseluruhan, meliputi sistem persepsi lingkungan, pemodelan lingkungan (representasi lingkungan lokal berbasis zona kedalaman), perencanaan lintasan dan *obstacle avoidance*, serta sistem kendali AUV. Setelah perancangan selesai, dilakukan implementasi berupa realisasi sistem ke dalam bentuk perangkat keras dan perangkat lunak, termasuk perakitan mekanik

dan elektrik, integrasi sensor, aktuator, dan unit pemrosesan, serta implementasi algoritma navigasi berbasis ROS2.

Tahap terakhir adalah pengujian dan evaluasi, yang bertujuan menilai kinerja sistem yang telah dikembangkan. Pengujian dilakukan untuk mengevaluasi kemampuan AUV dalam menghindari rintangan, kestabilan pergerakan, dan respons sistem terhadap perubahan lingkungan, baik pada lingkungan simulasi maupun pengujian langsung di kolam uji. Apabila hasil pengujian belum memenuhi kriteria keberhasilan, proses dikembalikan ke tahap perancangan untuk dilakukan perbaikan. Jika pengujian dinyatakan berhasil, penelitian dilanjutkan ke tahap analisis hasil dan penarikan kesimpulan.

3.2 Konsep dan Arsitektur Sistem AUV

Subbab ini membahas gambaran konseptual dan struktur sistem AUV secara menyeluruh, meliputi perancangan sistem, keterhubungan antar komponen, serta aliran data dari sensor hingga aktuator.

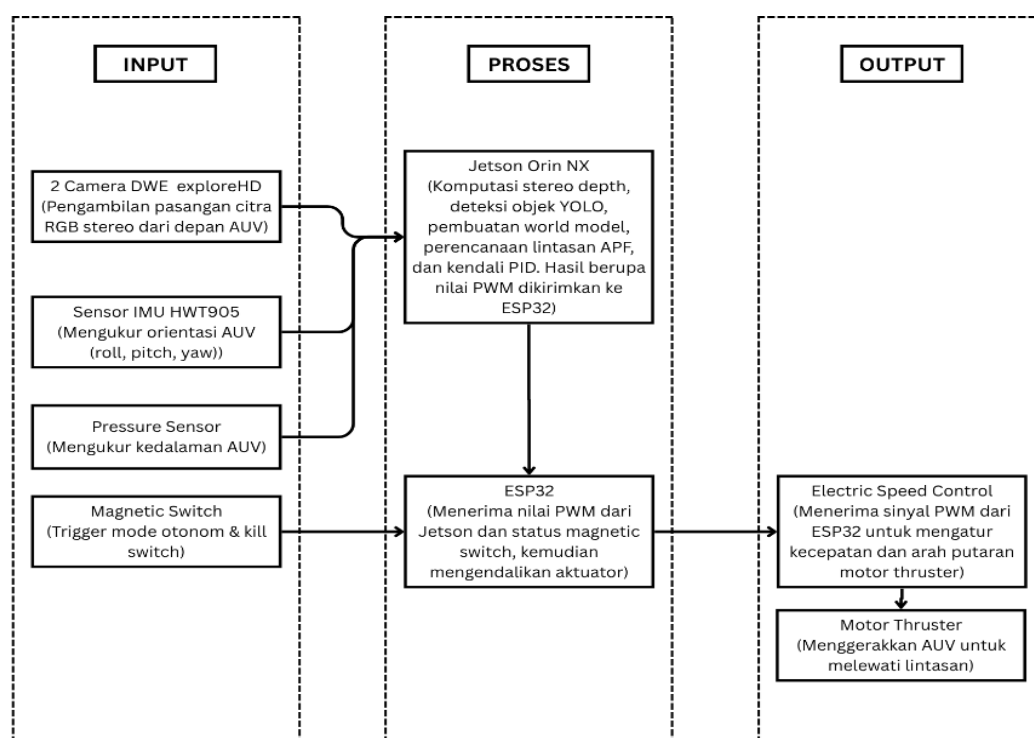
3.2.1 Konsep Penelitian

Konsep penelitian pada tugas akhir ini berfokus pada pengembangan sistem navigasi mandiri dan obstacle avoidance pada Autonomous Underwater Vehicle (AUV) berbasis persepsi visual dan perencanaan lintasan lokal. Sistem dirancang agar AUV mampu mendeteksi keberadaan rintangan di lingkungan bawah air, memperkirakan jarak rintangan secara akurat, serta menentukan arah dan kecepatan gerak yang aman menuju target secara *real-time*.

Pendekatan utama yang digunakan dalam penelitian ini adalah pemanfaatan kamera stereo sebagai sensor persepsi lingkungan. Kamera stereo memungkinkan sistem memperoleh informasi kedalaman melalui prinsip *stereo vision*, sehingga jarak relatif antara AUV dan rintangan dapat diestimasi tanpa bergantung pada sensor akustik tambahan. Untuk identifikasi objek target misi, digunakan metode deteksi objek berbasis *deep learning* You Only Look Once (YOLO) yang mampu mengenali objek secara andal terhadap variasi kondisi visual bawah air. Hasil deteksi YOLO dikombinasikan dengan informasi kedalaman *stereo*, kemudian diolah untuk membangun representasi lingkungan lokal berbasis zona kedalaman yang merepresentasikan posisi rintangan dan target dalam kerangka koordinat AUV.

Berdasarkan representasi lingkungan lokal berbasis zona kedalaman yang terbentuk, sistem selanjutnya menerapkan algoritma Artificial Potential Field (APF) sebagai perencana lintasan lokal. Algoritma APF menghasilkan gaya tarik semu menuju target dan gaya tolak semu dari rintangan, sehingga AUV dapat menentukan arah dan kecepatan gerak secara adaptif untuk menghindari tabrakan selama proses navigasi. Pendekatan ini bersifat reaktif terhadap perubahan lingkungan dan sesuai untuk skenario lintasan terstruktur seperti arena Kontes Robot Bawah Air Indonesia (KRBAI). Seluruh sistem diintegrasikan menggunakan *Robot Operating System 2* (ROS2) sebagai *middleware*, yang memungkinkan pembagian fungsi sistem ke dalam *node-node* modular dan saling terhubung melalui mekanisme komunikasi *publish-subscribe*. Dengan arsitektur tersebut, proses persepsi lingkungan, pemodelan lingkungan, perencanaan lintasan, dan pengendalian AUV dapat berjalan secara paralel dan real-time. Konsep penelitian ini diharapkan mampu menghasilkan sistem navigasi AUV yang stabil, adaptif, dan andal dalam menghadapi rintangan di lingkungan bawah air.

3.2.2 Diagram Blok Sistem



Gambar 3. 2 Blok Diagram Sistem Autonomous Underwater Vehicle

(Sumber : Dokumentasi Pribadi)

Seperti yang ditunjukkan pada Gambar 3.2, diagram blok sistem menggambarkan alur kerja sistem secara keseluruhan dalam penelitian ini. Sistem dibagi menjadi tiga bagian utama, yaitu *input*, proses, dan *output*, yang saling terhubung untuk mendukung navigasi mandiri dan *obstacle avoidance* pada Autonomous Underwater Vehicle (AUV).

Pada bagian input, Dua kamera DWE exploreHD yang disusun sebagai pasangan stereo (*stereo pair*) digunakan untuk mengambil citra dari arah depan AUV. Pasangan citra dari kedua kamera diolah oleh unit pemrosesan menghasilkan peta kedalaman (*depth map*) berdasarkan prinsip *stereo vision*, yang dimanfaatkan untuk mengestimasi jarak objek atau rintangan di depan AUV sehingga mendukung proses *obstacle avoidance* secara *real-time*. Sensor IMU HWT905 digunakan untuk mengukur orientasi dan arah hadap AUV berupa sudut *yaw*, *pitch*, dan *roll* yang diperlukan untuk menjaga kestabilan arah gerak selama misi berlangsung. Sensor tekanan MS5837 digunakan untuk mengukur kedalaman AUV saat berada di dalam air sehingga sistem dapat mempertahankan kedalaman operasi sesuai dengan target yang ditentukan. Selain itu, *magnetic switch* digunakan sebagai pemicu (*trigger*) otomatis yang mengaktifkan mode operasi mandiri AUV sekaligus berfungsi sebagai mekanisme keamanan (*kill switch*) untuk menghentikan seluruh aktuator secara instan apabila diperlukan.

Pada bagian proses, data dari seluruh sensor diolah oleh dua unit pemrosesan utama yang memiliki peran berbeda, yaitu NVIDIA Jetson Orin NX dan mikrokontroler ESP32. Kedua kamera DWE exploreHD, sensor IMU HWT905, dan sensor tekanan MS5837 seluruhnya terhubung langsung ke Jetson Orin NX sebagai unit pemrosesan tingkat tinggi. Jetson bertugas menangani komputasi *stereo depth* dari pasangan citra kamera untuk estimasi jarak rintangan, serta menjalankan model deteksi objek You Only Look Once (YOLO) yang diakselerasi TensorRT untuk mengidentifikasi jenis objek target sekaligus memperoleh posisi lateral objek dalam *frame* kamera yang digunakan untuk meluruskan *heading* AUV menuju target. Informasi jarak rintangan yang diperoleh kemudian digunakan untuk membangun representasi lingkungan lokal berbasis zona kedalaman yang merepresentasikan posisi rintangan secara relatif terhadap AUV. Berdasarkan representasi lingkungan lokal berbasis zona kedalaman

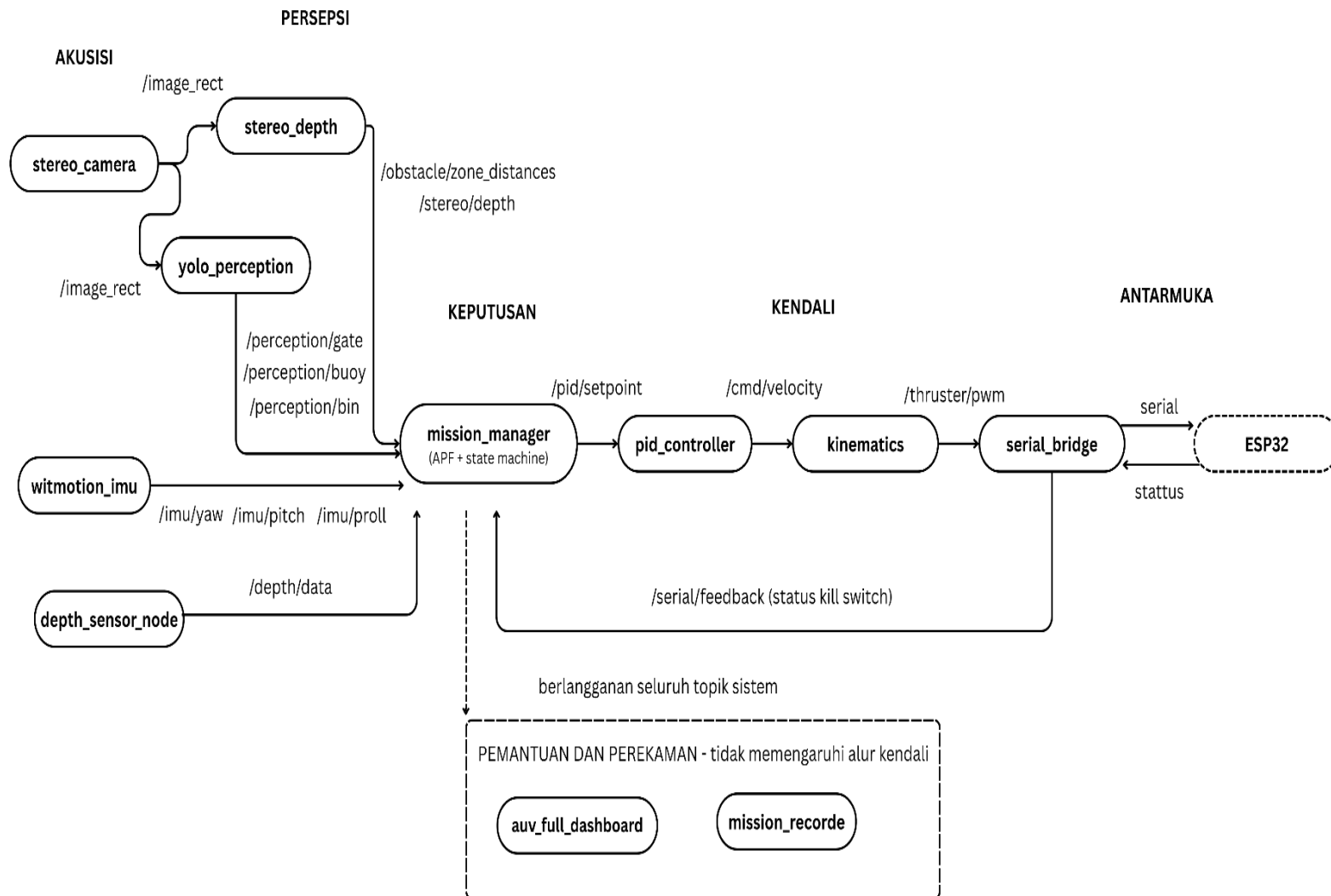
tersebut, algoritma Artificial Potential Field (APF) dijalankan sebagai *local planner* untuk menentukan arah dan kecepatan gerak AUV dalam menghindari rintangan.

Data orientasi dari sensor *Inertial Measurement Unit* (IMU) HWT905 digunakan oleh Jetson untuk mengetahui arah hadap AUV sehingga sistem navigasi dapat menentukan koreksi *heading* yang tepat. Data kedalaman dari sensor tekanan MS5837 digunakan oleh Jetson untuk memantau posisi vertikal AUV dan memberikan perintah kendali kedalaman yang sesuai. Seluruh hasil pemrosesan dari Jetson Orin NX kemudian dikirimkan ke ESP32 melalui komunikasi serial dua arah sebagai perintah gerak tingkat tinggi untuk mengendalikan aktuator.

ESP32 berfungsi sebagai unit kendali tingkat rendah (*low-level controller*) yang menerima perintah gerak dari Jetson Orin NX dan mengeksekusinya langsung ke aktuator. Selain menerima perintah dari Jetson, ESP32 juga terhubung langsung ke *magnetic switch* untuk memantau status pemicu secara independen. Hal ini memastikan bahwa mekanisme aktivasi mode otonom dan *kill switch* tetap berfungsi secara *real-time* tanpa bergantung pada kondisi Jetson. Berdasarkan perintah dari Jetson dan status *magnetic switch*, ESP32 menghasilkan sinyal kendali berupa *Pulse Width Modulation* (PWM) yang dikirimkan ke aktuator.

Pada bagian *output*, sinyal PWM dari ESP32 dikirimkan ke *Electronic Speed Controller* (ESC) yang mengendalikan kecepatan dan arah putaran delapan motor *thruster* T200. Kombinasi putaran kedelapan *thruster* ini memungkinkan AUV untuk bergerak maju, berbelok, menyesuaikan kedalaman, serta mempertahankan kestabilan posisi selama melewati lintasan arena Kontes Robot Bawah Air (KRBAI). Dengan demikian, seluruh rangkaian *input*, proses, dan *output* bekerja secara terintegrasi dalam satu arsitektur sistem yang modular untuk mendukung operasi AUV secara otonom pada lintasan kompetisi KRBAI. Arsitektur terintegrasi ini tidak hanya memastikan pembagian beban komputasi yang efisien antara pemrosesan visual tingkat tinggi dan kendali aktuator tingkat rendah, tetapi juga menciptakan sistem umpan balik (*feedback loop*) yang sangat responsif. Kecepatan transmisi data dari Jetson menuju ESP32 meminimalkan waktu tunda (*latency*) eksekusi, sehingga manuver penghindaran rintangan hasil kalkulasi algoritma Artificial Potential Field (APF) dapat direalisasikan secara instan dan presisi oleh kedelapan *thruster*.

3.2.3 Arsitektur Sistem Berbasis ROS2



Gambar 3. 3 Diagram *node* ROS2 pada AUV

(Sumber : Dokumentasi Pribadi)

Gambar 3.3 menggambarkan arsitektur komunikasi antar *node* dalam sistem ROS2 yang diterapkan pada penelitian ini. Sistem terdiri dari sebelas *node* utama yang masing-masing memiliki fungsi spesifik dan saling berkomunikasi melalui *topic* ROS2 dengan mekanisme *publish-subscribe*. Penjelasan alur komunikasi antar *node* adalah sebagai berikut.

Proses dimulai dari *stereo_camera_node* yang bertugas melakukan akuisisi citra dari kedua kamera DWE exploreHD melalui antarmuka V4L2 secara bersamaan, kemudian menerapkan rektifikasi pada pasangan citra menggunakan parameter hasil kalibrasi *stereo*. Pasangan citra terektifikasi dipublikasikan melalui *topic* */stereo/left/image_rect* dan */stereo/right/image_rect*. Citra terektifikasi ini selanjutnya dikonsumsi oleh dua *node* persepsi yang berjalan secara paralel. Pertama, *stereo_depth_node* menghitung peta kedalaman (*depth map*) dari pasangan citra menggunakan algoritma *Semi-Global Block Matching* (SGBM), menerapkan koreksi jarak bawah air hasil kalibrasi, kemudian membagi *depth map* menjadi zona-zona deteksi rintangan dan mempublikasikan jarak rintangan terdekat setiap zona sebagai masukan gaya tolak semu algoritma APF. Kedua, *yolo_perception_node* menjalankan model deteksi objek YOLO yang diakselerasi TensorRT pada citra kiri untuk mengidentifikasi tiga objek target misi, kemudian mempublikasikan hasil deteksi melalui *topic* */perception/gate*, */perception/buoy*, dan */perception/bin* dalam format larik tujuh elemen [cx, cy, dist, w, cx_norm, cy_norm, conf] yang memuat posisi objek dalam frame, estimasi jarak, dan nilai keyakinan deteksi.

Selanjutnya, *witmotion_imu_node* mengakuisisi data orientasi AUV dari sensor IMU HWT905 dan mempublikasikannya melalui *topic* */imu/yaw*, */imu/pitch*, dan */imu/roll*, sedangkan *depth_sensor_node* mengakuisisi data kedalaman dari sensor MS5837 dan mempublikasikannya melalui *topic* */depth/data*. Data dari kedua *node* persepsi visual bersama data orientasi dan kedalaman seluruhnya dikonsumsi oleh *mission_manager_node* sebagai *node* inti yang mengelola urutan misi, membangun representasi lingkungan lokal berbasis zona kedalaman posisi rintangan relatif terhadap AUV, dan menjalankan algoritma Artificial Potential Field (APF) sebagai *local planner* gaya tarik semu dihitung berdasarkan posisi

lateral objek target dari hasil deteksi YOLO, sedangkan gaya tolak semu dihitung dari zona jarak rintangan hasil *stereo depth*.

Hasil perencanaan berupa setpoint gerak dipublikasikan *mission_manager_node* melalui *topic /pid/setpoint* dan */cmd/velocity*, kemudian diterima oleh *pid_controller_node* yang menghitung sinyal kendali PID pada empat *Degree of Freedom* yaitu *depth*, *yaw*, *roll*, dan *pitch* dengan membandingkan *setpoint* terhadap umpan balik sensor. Keluaran PID pada *topic /pid/output* selanjutnya dikonversi oleh *kinematics_node* menjadi nilai PWM untuk kedelapan *thruster* dan dipublikasikan melalui *topic /thruster/pwm*. Nilai PWM tersebut diterima oleh *serial_bridge_node* yang mengemasnya menjadi paket serial berformat $P\{m1\}, \dots, \{m8\}E$ dan mengirimkannya ke ESP32 melalui antarmuka *USB to TTL*, sekaligus menerima umpan balik status sistem dari ESP32 melalui *topic /serial/feedback*. Seluruh data dari semua node aktif dipantau secara *real-time* oleh *auv_full_dashboard* yang menampilkannya pada antarmuka AUV KRBAI Dashboard dan ada node satu lagi untuk merekam seluruh node kemudian disimpan menjadi *rosvbag* untuk kebutuhan analisis pengujian yaitu *mission_recorder_node*. Tabel 3.1 menyajikan ringkasan seluruh node beserta fungsinya.

Tabel 3. 1 Daftar *Node* ROS2 dan Fungsinya pada Sistem AUV KRBAI

No	Nama <i>Node</i>	Fungsi	<i>Topic</i> Utama
1	<i>stereo_camera_node</i>	Akuisisi 2 kamera DWE exploreHD + rektifikasi citra stereo	<i>/stereo/left/image_rect</i> , <i>/stereo/right/image_rect</i>
2	<i>stereo_depth_node</i>	Komputasi <i>depth map</i> SGBM + koreksi bawah air + zona rintangan	<i>/stereo/depth</i> , <i>/obstacle/nearest_distance</i>
3	<i>yolo_perception_node</i>	Deteksi objek YOLO TensorRT (gawang, suar, bak)	<i>/perception/gate</i> , <i>/perception/buoy</i> , <i>/perception/bin</i>
4	<i>witmotion_imu_node</i>	Akuisisi data orientasi sensor HWT905	<i>/imu/yaw</i> , <i>/imu/pitch</i> , <i>/imu/roll</i>
5	<i>depth_sensor_node</i>	Akuisisi data kedalaman sensor MS5837	<i>/depth/data</i>

No	Nama <i>Node</i>	Fungsi	<i>Topic</i> Utama
6	<i>mission_manager_node</i>	Manajemen misi, representasi lingkungan lokal berbasis zona kedalaman, dan algoritma APF	<i>/pid/setpoint, /cmd/velocity</i>
7	<i>pid_controller_node</i>	Kendali PID 4 DOF (<i>depth, yaw, roll, pitch</i>)	<i>/pid/output, /pid/error</i>
8	<i>kinematics_node</i>	Konversi <i>setpoint</i> ke nilai PWM 8 <i>thruster</i>	<i>/thruster/pwm</i>
9	<i>serial_bridge_node</i>	Komunikasi serial dua arah Jetson–ESP32	<i>/serial/feedback</i>
10	<i>auv_full_dashboard</i>	Monitoring dan perekaman data sistem <i>real-time</i>	<i>Subscriber</i> seluruh topic
11	<i>mission_recorder_node</i>	Merekam seluruh data atau <i>topic</i> disimpan pada <i>rosvbag</i>	<i>Subscriber</i> seluruh topic

(Sumber : Dokumentasi Pribadi)

3.3 Sistem Persepsi dan Pemodelan Lingkungan

Subbab ini menjelaskan sistem persepsi lingkungan AUV dan proses pengolahan data hasil persepsi menjadi representasi lingkungan yang digunakan sebagai masukan algoritma perencanaan lintasan berbasis APF.

3.3.1 Akuisisi Citra Stereo (Kamera DWE exploreHD)

Sistem persepsi visual pada penelitian ini diawali dari tahap akuisisi citra menggunakan dua kamera DWE exploreHD yang dipasang sejajar secara horizontal pada bagian depan AUV dengan jarak antar pusat lensa (*baseline*) yang tetap. Kedua kamera terhubung ke Jetson Orin NX melalui antarmuka USB dan dibaca sebagai perangkat Video4Linux (V4L2) tanpa memerlukan *driver* khusus. Kamera DWE *exploreHD* dipilih karena dirancang khusus untuk aplikasi bawah air dengan *housing* kedap air bawaan, sehingga dapat dipasang langsung pada bagian luar bodi AUV tanpa memerlukan *enclosure* tambahan yang berpotensi menambah distorsi optik.

Kedua kamera menggunakan lensa *fisheye* dengan medan pandang yang lebar, sehingga mampu menangkap area pengamatan yang luas dalam satu *frame*

karakteristik yang menguntungkan untuk mendeteksi objek target pada arena kompetisi sedini mungkin. Konsekuensi dari penggunaan lensa *fisheye* adalah citra mentah yang dihasilkan mengalami distorsi radial signifikan, sehingga koreksi distorsi melalui proses kalibrasi menjadi tahapan wajib sebelum citra dapat digunakan untuk komputasi *stereo depth*, sebagaimana dijelaskan pada subbab 3.3.2.

Berbeda dengan kamera stereo terintegrasi yang memiliki sinkronisasi perangkat keras antar lensa, kedua kamera DWE beroperasi secara independen sehingga pengambilan *frame* kiri dan kanan dilakukan sedekat mungkin secara perangkat lunak untuk meminimalkan selisih waktu antar *frame*. Pendekatan ini memadai untuk kondisi operasional AUV yang bergerak dengan kecepatan rendah sesuai batasan kecepatan maksimum sistem sebesar 0,5 m/s. Seluruh proses akuisisi ditangani oleh *stereo_camera_node* yang membaca kedua kamera secara bersamaan pada resolusi 640×480 *piksel*, menerapkan rektifikasi menggunakan peta hasil kalibrasi, kemudian mempublikasikan pasangan citra terektifikasi melalui *topic /stereo/left/image_rect* dan */stereo/right/image_rect* untuk dikonsumsi oleh *node* persepsi berikutnya.



Gambar 3. 4 Pemasangan Dua Kamera DWE exploreHD sebagai *Stereo Pair* pada AUV

(Sumber: Dokumentasi Pribadi)

Gambar 3.4 menunjukkan konfigurasi pemasangan kedua kamera DWE *exploreHD* pada bagian depan AUV. Kedua kamera dipasang padaudukan (*mounting*) yang *rigid* untuk memastikan *baseline* dan orientasi relatif antar kamera tidak berubah selama operasi, karena pergeseran posisi kamera sekecil apapun akan membatalkan validitas parameter kalibrasi *stereo*.



Gambar 3. 5 Contoh Pasangan Citra Kamera Kiri dan Kanan DWE *exploreHD*

(Sumber: Dokumentasi Pribadi)

Gambar 3.5 memperlihatkan contoh pasangan citra yang diakuisisi secara bersamaan dari kamera kiri dan kamera kanan. Terlihat bahwa objek yang sama berada pada posisi *piksel* yang sedikit berbeda di kedua citra pergeseran posisi inilah yang menjadi dasar perhitungan disparitas untuk estimasi jarak. Terlihat pula efek distorsi lensa *fish-eye* pada bagian tepi citra yang menyebabkan garis lurus tampak melengkung, yang selanjutnya dikoreksi melalui proses rektifikasi.

3.3.2 Kalibrasi Stereo dan Kalibrasi Bawah Air

Sebelum pasangan kamera dapat digunakan untuk komputasi kedalaman, sistem harus melalui dua tahap kalibrasi, yaitu kalibrasi stereo untuk memperoleh parameter geometris kamera, serta kalibrasi bawah air untuk mengoreksi penyimpangan estimasi jarak akibat efek refraksi ketika kamera dioperasikan di dalam air. Tahap kalibrasi stereo berfokus pada ekstraksi parameter intrinsik seperti panjang fokus dan titik pusat optik, serta parameter ekstrinsik yang mencakup matriks rotasi dan translasi antar kedua lensa. Parameter-parameter tersebut sangat esensial untuk melakukan rektifikasi citra, sehingga gambar dari kamera kiri dan kanan sejajar secara horizontal pada garis *epipolar* yang sama. Namun, parameter geometris yang diperoleh di udara terbuka tidak dapat sepenuhnya diterapkan saat AUV beroperasi di bawah permukaan air akibat adanya perbedaan indeks bias

medium. Perbedaan indeks bias antara air, kaca pelindung (*housing*), dan udara di dalam wadah kamera menyebabkan gelombang cahaya mengalami pembelokan yang mendistorsi proyeksi visual. Fenomena ini secara langsung mengakibatkan bias pada pembacaan nilai disparitas, yang berujung pada penyimpangan estimasi jarak rintangan dari posisi aktualnya. Oleh karena itu, tahap kalibrasi bawah air bertindak sebagai mekanisme kompensasi matematis untuk memperbaiki skala kesalahan tersebut, guna memastikan sistem AUV mampu mengukur kedalaman objek dengan tingkat akurasi yang presisi.



Gambar 3. 6 Proses Kalibrasi Stereo Menggunakan Papan Catur Tampak Kiri dan Kanan (Checkerboard)

(Sumber: Dokumentasi Pribadi)

Kalibrasi stereo bertujuan memperoleh dua kelompok parameter. Pertama, parameter intrinsik masing-masing kamera yang meliputi panjang fokus, titik pusat citra (*principal point*), serta koefisien distorsi lensa. Parameter distorsi menjadi sangat penting pada penelitian ini karena kamera DWE exploreHD menggunakan lensa *fish-eye* dengan medan pandang lebar yang menghasilkan distorsi radial signifikan pada citra mentah. Kedua, parameter ekstrinsik yang mendeskripsikan hubungan geometris antar kedua kamera, berupa matriks rotasi dan vektor translasi dari kamera kiri terhadap kamera kanan. Kalibrasi dilakukan menggunakan papan catur (*checkerboard*) dengan dimensi kotak yang diketahui, yang ditangkap secara bersamaan oleh kedua kamera dari berbagai posisi dan sudut pandang. Titik-titik sudut papan catur yang terdeteksi pada pasangan citra digunakan untuk menghitung seluruh parameter kalibrasi melalui proses optimasi yang meminimalkan kesalahan *reproyeksi* (*reprojection error*). Gambar 3.6 menunjukkan proses pengambilan citra kalibrasi, dimana papan catur ditangkap secara bersamaan oleh kamera kiri dan kamera kanan dari berbagai posisi dan orientasi. Kalibrasi stereo dilakukan di

udara. Percobaan kalibrasi dengan menempatkan papan catur langsung di dalam air sempat dilakukan, namun menghasilkan galat reprojeksi intrinsik yang tinggi (di atas 1,2 *piksel*) akibat distorsi tambahan dari pembiasan pada antarmuka air, sehingga parameter hasil kalibrasi udara yang digunakan. Penyimpangan jarak yang timbul saat kamera dioperasikan di dalam air kemudian dikompensasi secara terpisah melalui kalibrasi koreksi jarak, sebagaimana dijelaskan berikut. Titik-titik sudut internal papan catur yang terdeteksi pada kedua citra (ditandai dengan pola garis berwarna) menjadi titik referensi dalam perhitungan parameter kalibrasi. Hasil kalibrasi selanjutnya digunakan untuk menghasilkan peta rektifikasi (*rectification map*) yang disimpan dan dimuat oleh *stereo_camera_node* pada saat sistem dijalankan, sehingga proses kalibrasi tidak perlu diulang selama posisi kedua kamera pada kedudukan tidak berubah.

Kalibrasi yang dilakukan di udara belum cukup untuk pengoperasian di dalam air. Sebagaimana dijelaskan pada landasan teori, cahaya yang merambat dari medium air menuju kamera mengalami pembelokan pada antarmuka air dan *housing* kamera akibat perbedaan indeks bias, sehingga panjang fokus efektif sistem kamera di dalam air berbeda dengan hasil kalibrasi udara. Akibatnya, estimasi jarak yang dihitung menggunakan parameter kalibrasi udara menghasilkan penyimpangan yang bersifat sistematis terhadap jarak sebenarnya. Untuk mengatasi hal tersebut, dilakukan kalibrasi koreksi jarak tambahan yang dilaksanakan langsung pada kondisi kamera terendam di dalam air.

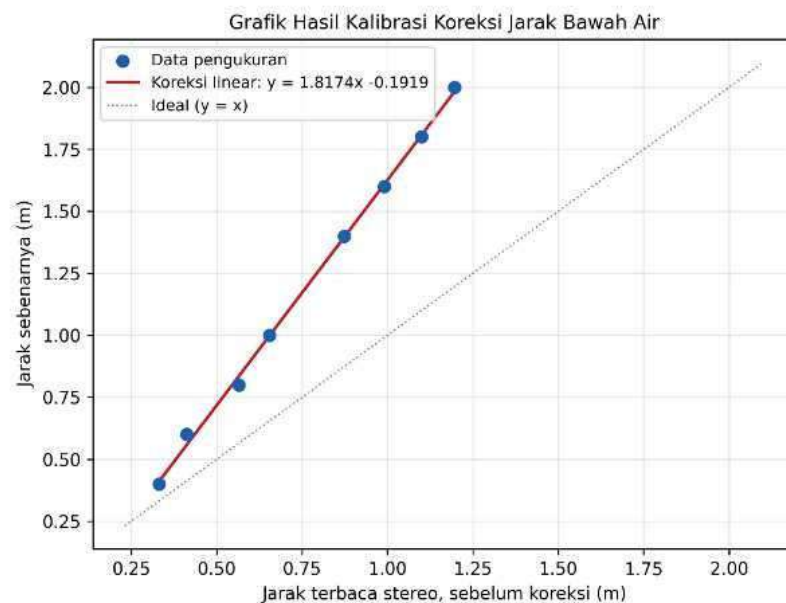


Gambar 3. 7 Proses Kalibrasi Koreksi Jarak Bawah Air Menggunakan Objek Referensi (bak) dengan Pembacaan Jarak Melalui Deteksi YOLO

(Sumber: Dokumentasi Pribadi)

Prosedur kalibrasi bawah air dilakukan dengan menempatkan objek referensi pada serangkaian jarak nyata yang diukur secara manual, kemudian mencatat nilai jarak yang diestimasi oleh sistem *stereo depth* pada setiap titik pengukuran. Gambar 3.7 memperlihatkan proses pengambilan data kalibrasi bawah air. Objek referensi ditempatkan pada jarak yang diukur secara manual menggunakan meteran, sementara sistem membaca estimasi jarak stereo pada area objek. Untuk memastikan pembacaan jarak diambil tepat pada objek dan bukan pada latar, area objek ditentukan melalui kotak deteksi YOLO, lalu nilai kedalaman dihitung dari dalam kotak tersebut (nilai jarak terbaca tampak pada label deteksi). Pengukuran awal menggunakan suar, namun estimasi jaraknya menunjukkan galat yang besar karena bentuk suar yang sempit membuat sebagian kotak deteksi berisi air alih-alih permukaan objek, sehingga disparitas yang dihasilkan kurang stabil. Objek kemudian diganti dengan bak yang memiliki permukaan lebih lebar dan bertekstur, sehingga estimasi jaraknya jauh lebih stabil dan konsisten. Pasangan data jarak terukur dan jarak sebenarnya selanjutnya digunakan untuk menghitung fungsi koreksi linear. Pasangan data jarak terukur dan jarak sebenarnya selanjutnya digunakan untuk menghitung fungsi koreksi linear:

$$Z_{koreksi} = a \cdot Z_{ukur} + b \quad (3.1)$$



Gambar 3. 8 Grafik Hasil Kalibrasi Koreksi Jarak Bawah Air

(Sumber: Dokumentasi Pribadi)

dengan $Z_{koreksi}$ adalah jarak hasil koreksi (meter), Z_{ukur} adalah jarak hasil estimasi *stereo depth* sebelum koreksi (meter), a adalah faktor skala, dan b adalah konstanta offset. Kedua parameter diperoleh melalui regresi linear terhadap data pengukuran. Berdasarkan hasil kalibrasi pada rentang jarak operasional 0,4 hingga 2,0 meter, diperoleh faktor skala $a = 1,8174$ dan offset $b = -0,1919$, sehingga fungsi koreksi yang diterapkan adalah $Z_{koreksi} = 1,8174 \text{ dam } Z_{ukur} - 0,1919$. Seluruh parameter geometris dan parameter koreksi yang digunakan sistem dirangkum pada Tabel 3.2.

Tabel 3. 2 Parameter Kalibrasi Stereo dan Koreksi Jarak Bawah Air

No.	Parameter	Nilai	Keterangan
1	Panjang fokus terekstifikasi (f)	350,99 <i>piksel</i>	Diambil dari matriks proyeksi P[0]
2	<i>Baseline stereo</i> (B)	98,51 mm	Jarak antar pusat kamera kiri dan kanan
3	Faktor skala koreksi (a)	1,8174	Hasil regresi linear data bawah air
4	Konstanta <i>offset</i> (b)	-0,1919	Hasil regresi linear data bawah air
5	Rentang belaku koreksi	0,4 – 2,0 m	Rentang jarak operasional misi
6	Galat maksimum setelah koreksi	4,3 cm	Diuji pada Bab 4

(Sumber : Dokumentasi Pribadi)

Penggunaan panjang fokus dari matriks proyeksi P dan bukan dari matriks intrinsik K perlu ditegaskan, karena nilai kedalaman dihitung dari citra yang telah melalui proses rektifikasi. Matriks K menggambarkan kondisi citra mentah yang masih mengandung distorsi lensa *fisheye*, sedangkan matriks P menggambarkan kondisi citra setelah rektifikasi, yaitu kondisi yang sebenarnya dipakai algoritma SGBM saat mencari kesesuaian *piksel*. Penggunaan nilai dari matriks K akan menghasilkan estimasi jarak yang keliru meskipun algoritmanya benar.

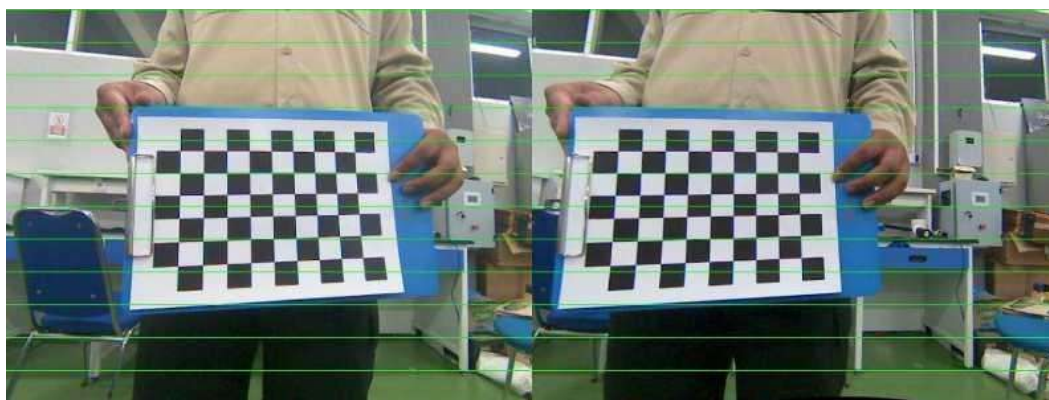
Faktor skala sebesar 1,8174 dapat dijelaskan secara fisis melalui indeks bias air. Regresi yang sama pada data pengukuran di udara menghasilkan faktor skala 1,3342, sehingga rasio antara keduanya adalah $1,8174 \div 1,3342 = 1,362$. Nilai rasio ini sangat mendekati indeks bias air tawar yaitu $n \approx 1,333$, sehingga penyimpangan

estimasi jarak di dalam air terbukti berasal dari pembiasan cahaya pada antarmuka air dan *housing* kamera, bukan dari kesalahan kalibrasi maupun kesalahan algoritma. Kesesuaian ini menjadi dasar fisis yang memperkuat penggunaan fungsi koreksi linear pada penelitian ini.

3.3.3 Rektifikasi dan Komputasi *Depth Map* (SGBM)

Setelah parameter kalibrasi diperoleh, tahap berikutnya adalah pengolahan pasangan citra menjadi peta kedalaman (*depth map*). Proses ini terdiri dari dua tahapan utama, yaitu rektifikasi citra dan komputasi disparitas menggunakan algoritma *Semi-Global Block Matching* (SGBM).

Rektifikasi merupakan transformasi geometris yang menyelaraskan pasangan citra sehingga setiap titik pada citra kiri berada pada baris *pixel* yang sama dengan titik korespondensinya di citra kanan. Proses ini sekaligus mengoreksi distorsi lensa *fish-eye* menggunakan koefisien distorsi hasil kalibrasi. Rektifikasi menyederhanakan pencarian korespondensi dari pencarian dua dimensi menjadi pencarian satu dimensi di sepanjang baris citra, sehingga mengurangi beban komputasi secara signifikan dan memungkinkan pemrosesan berjalan *real-time*.



Gambar 3. 9 Verifikasi Rektifikasi Stereo pada Pasangan Kamera (garis hijau menunjukkan garis epipolar horizontal)

(Sumber: Dokumentasi Pribadi)

Gambar 3.9 menunjukkan hasil rektifikasi pasangan kamera stereo. Garis epipolar horizontal (hijau) ditumpangkan untuk memverifikasi kesejajaran kedua kamera: titik objek yang sama pada citra kiri dan kanan misalnya sudut-sudut pola papan catur berada pada baris *pixel* yang sama. Kesejajaran ini memastikan pencarian korespondensi (disparitas) untuk perhitungan kedalaman cukup dilakukan sepanjang garis horizontal, yang menjadi syarat kerja algoritma stereo

matching. Distorsi fisheye pada citra mentah telah dikoreksi sehingga garis-garis lurus pada objek tampak lurus setelah rektifikasi.

Pasangan citra terekstifikasi selanjutnya diproses oleh algoritma SGBM untuk menghasilkan *disparity map*. Algoritma ini mencari korespondensi setiap blok *piksel* citra kiri terhadap citra kanan di sepanjang baris yang sama dengan mempertimbangkan kehalusan disparitas antar *piksel* bertetangga, sehingga menghasilkan peta disparitas yang lebih konsisten dibandingkan metode pencocokan blok sederhana. Nilai disparitas kemudian dikonversi menjadi jarak menggunakan hubungan triangulasi stereo:

$$Z = \frac{f \times B}{d} \quad (3.2)$$

dengan Z adalah jarak objek terhadap kamera (meter), f adalah panjang fokus hasil kalibrasi (*piksel*), B adalah *baseline* antar kamera (meter), dan d adalah nilai disparitas (*piksel*). Sebagai ilustrasi, panjang fokus hasil kalibrasi sebesar 350,99 *piksel* dan gambar sebesar 0,09851 meter menghasilkan hasil kali $f \times B$ sebesar 34,576. Apabila disparitas suatu objek terukur 40 *piksel*, maka jarak objek adalah $34,576 \div 40 = 0,864$ meter. Nilai jarak hasil perhitungan tersebut selanjutnya dikoreksi menggunakan fungsi kalibrasi bawah air sebagaimana dijelaskan pada subbab 3.3.2, sehingga diperoleh jarak sebenarnya sebesar $1,8174 \times 0,864 = 1,580$ meter. Selisih yang cukup besar antara nilai sebelum dan sesudah koreksi menegaskan bahwa tanpa koreksi bawah air sistem akan mengestimasi jarak objek jauh lebih pendek daripada jarak sebenarnya.

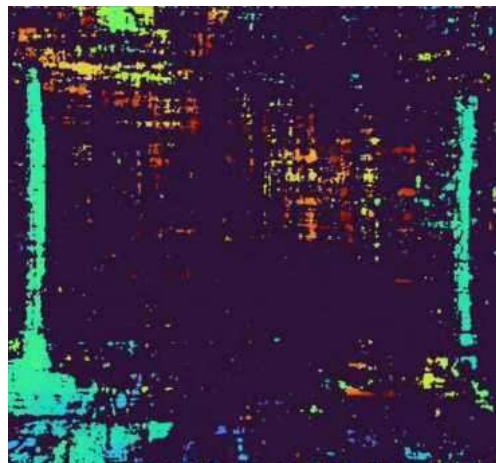
Nilai parameter algoritma SGBM yang digunakan pada penelitian ini dirangkum pada Tabel 3.3.

Tabel 3. 3 Parameter Algoritma *Semi-Global Block Matching* (SGBM)

No.	Parameter	Nilai	Keterangan
1	minDisparity	0	Nilai disparitas terkecil yang dicari
2	numDisparities	128	Rentang pencarian disparitas
3	blockSize	7	Ukuran jendela pencocokan
4	P1	10	Penalti perubahan disparitas kecil

No.	Parameter	Nilai	Keterangan
5	P2	120	Penalti perubahan disparitas besar
6	disp12MaxDiff	2	Batas selisih pemeriksaan kiri-kanan
7	uniquenessRatio	13	Ambang keunikan hasil pencocokan
8	speckleWindowSize	200	Ukuran jendela penyaringan bercak
9	speckleRange	2	Rentang toleransi penyaringan bercak
10	Mode komputasi	CUDA (GPU)	Dijalankan pada GPU Jetson Orin NX

(Sumber : Dokumentasi Pribadi)



Gambar 3. 10 Hasil Komputasi Depth Map Menggunakan Algoritma SGBM

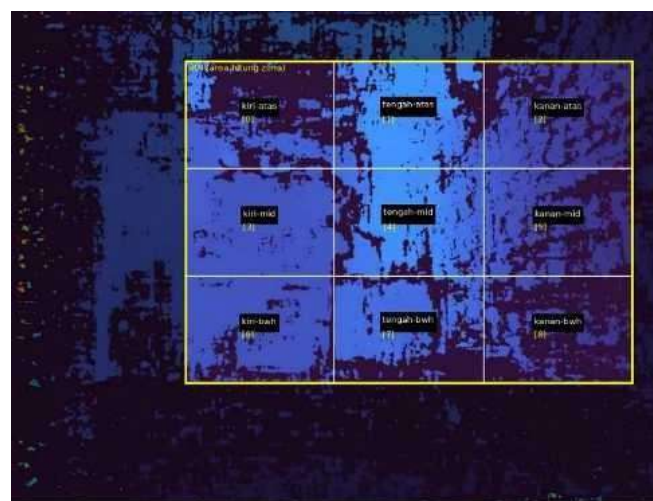
(Sumber: Dokumentasi Pribadi)

Gambar 3.10 menunjukkan hasil komputasi depth map menggunakan algoritma *Semi-Global Block Matching* (SGBM). Pada representasi warna ini, warna terang seperti hijau kebiruan (*cyan*) menunjukkan permukaan objek yang berada lebih dekat dengan kamera, sedangkan area berwarna gelap (ungu tua) merepresentasikan latar belakang atau area yang tidak memiliki nilai kedalaman. Berdasarkan hasil pemetaan, struktur vertikal pada sisi kiri dan sisi kanan tiang gawang berhasil terbaca dengan jelas oleh algoritma. Namun, bagian tengah

gawang tidak dapat terbaca dan menghasilkan area gelap (kosong). Hal ini menunjukkan bahwa algoritma gagal menemukan korespondensi disparitas pada area tengah tersebut, yang kemungkinan besar disebabkan oleh minimnya tekstur (area bertekstur rendah) pada bagian palang tengah gawang atau akibat tantangan visual yang umum terjadi di dalam air.

Panel kiri merupakan citra masukan dari kamera kiri, sedangkan panel kanan merupakan peta kedalaman dengan representasi warna: merah menunjukkan objek dekat dan biru menunjukkan objek jauh. Objek di latar depan (papan kalibrasi dan operator) terpetakan dengan kedalaman dekat, sedangkan latar belakang (kursi, meja, dinding) terpetakan lebih jauh, menunjukkan pemisahan kedalaman yang sesuai dengan geometri scene. Area gelap pada tepi kiri peta kedalaman merupakan daerah yang tidak memiliki korespondensi disparitas antara kedua kamera, sehingga tidak dihitung nilai kedalamannya.

Untuk kebutuhan *obstacle avoidance*, *depth map* tidak digunakan per *piksel* melainkan dibagi menjadi zona-zona deteksi berbentuk kisi. Setiap zona merepresentasikan satu arah relatif terhadap AUV, dan nilai jarak perwakilan setiap zona ditentukan menggunakan pendekatan persentil terhadap *piksel* valid di dalam zona tersebut untuk mengurangi pengaruh *noise* dan *piksel outlier*. Zona dengan jarak di bawah radius pengaruh D_0 diklasifikasikan sebagai zona rintangan aktif dan menjadi masukan gaya tolak semu algoritma APF. Seluruh proses komputasi *depth map* hingga publikasi zona rintangan diimplementasikan pada *stereo_depth_node*.



Gambar 3. 11 Pembagian Zona Deteksi Rintangan pada Depth Map

(Sumber: Dokumentasi Pribadi)

Gambar 3.11 menunjukkan pembagian wilayah deteksi rintangan pada depth map. Sebelum dibagi menjadi zona, citra kedalaman dipotong terlebih dahulu menjadi sebuah Region of Interest (ROI) untuk membuang bagian tepi yang tidak andal. Pemotongan dilakukan sebesar 10% dari atas, 25% dari bawah, 170 *piksel* dari kiri, dan 5% dari kanan. Bagian bawah dibuang karena menangkap lantai atau dasar kolam yang masuk ke dalam pandangan kamera; kolom kiri dibuang karena tidak memiliki korespondensi disparitas akibat nilai $\text{num_disparities} = 128$, sehingga tidak memiliki data kedalaman; sedangkan tepi atas dan kanan dibuang untuk mengurangi pengaruh distorsi fisheye yang paling besar pada tepi lensa. Wilayah ROI hasil pemotongan tersebut tidak simetris terhadap pusat citra, karena sisi kiri dipotong jauh lebih besar (170 *piksel*, setara $\pm 27\%$ lebar citra) dibandingkan sisi kanan (5%). Ketidaksimetrisan ini merupakan konsekuensi langsung dari keterbatasan stereo matching: wilayah tak-ter-match selalu berada pada sisi kiri citra selebar num_disparities . Akibatnya, kolom tengah pada pembagian zona sedikit bergeser ke kanan dari sumbu optik kamera. Hal ini tidak berpengaruh signifikan terhadap kinerja penghindaran rintangan, karena Keputusan navigasi bersifat relatif antar-kolom (kiri–tengah–kanan) di dalam ROI, bukan bergantung pada posisi absolut terhadap pusat citra.

Wilayah ROI kemudian dibagi menjadi grid 3×3 sehingga menghasilkan 9 zona: kiri-atas, tengah-atas, kanan-atas, kiri-mid, tengah-mid, kanan-mid, kiri-bawah, tengah-bawah, dan kanan-bawah (indeks [0]–[8]). Setiap zona diwakili oleh satu nilai jarak, yaitu persentil ke-25 dari *piksel* valid di dalam zona tersebut. Penggunaan persentil ke-25 (bukan nilai minimum) beserta syarat minimum 80 *piksel* valid bertujuan agar pembacaan tahan terhadap gangguan (artefak): zona yang hanya memiliki sedikit *piksel* valid dianggap kosong, dan nilai jaraknya tidak terkunci pada segelintir *piksel* artefak. Nilai kesembilan zona inilah yang menjadi masukan bagi perhitungan gaya tolak semu APF pada kolom kiri, tengah, dan kanan.

3.3.4 Deteksi Objek Berbasis YOLO

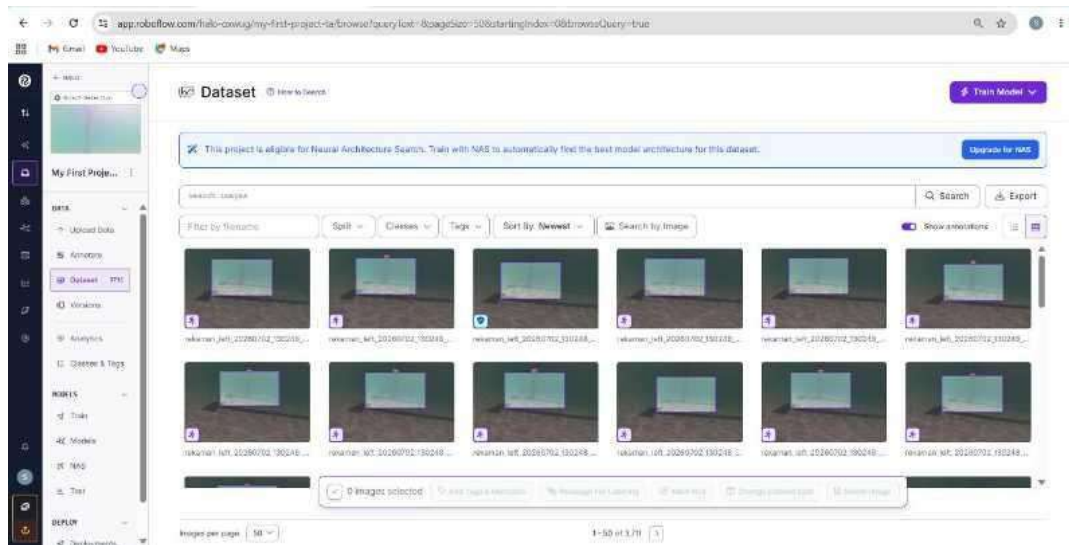
Identifikasi objek target misi dilakukan menggunakan model deteksi objek You Only Look Once (YOLO) yang dilatih khusus untuk mengenali tiga kelas objek arena KRBAI, yaitu gawang (*gate*), suar (*buoy*), dan bak (*bin*). Metode

deteksi berbasis *deep learning* dipilih menggantikan pendekatan segmentasi warna konvensional karena lebih tangguh terhadap variasi pencahayaan bawah air, perubahan sudut pandang, dan variasi jarak objek. Pada tahap pengembangan awal, deteksi objek diuji menggunakan segmentasi warna HSV, namun parameter warna terbukti tidak konsisten terhadap variasi jarak dan kondisi pencahayaan bawah air rentang warna yang telah dikalibrasi pada satu jarak menghasilkan deteksi yang tidak stabil pada jarak lainnya, sehingga pendekatan deteksi dialihkan ke metode YOLO.

Model yang digunakan adalah YOLOv8 varian *nano* (YOLOv8n), yaitu varian dengan jumlah parameter terkecil yang dirancang untuk perangkat dengan sumber daya komputasi terbatas. Pemilihan varian ini didasarkan pada kebutuhan menjalankan inferensi secara *real-time* pada Jetson Orin NX bersamaan dengan proses komputasi *stereo depth* dan perencanaan lintasan yang berjalan paralel pada perangkat yang sama.

Proses pengembangan model deteksi dilakukan melalui empat tahapan. Pertama, pengumpulan *dataset* dilakukan dengan merekam video ketiga objek target langsung dari lingkungan pengujian menggunakan kamera DWE exploreHD yang sama dengan yang terpasang pada AUV, sehingga karakteristik citra pelatihan meliputi distorsi lensa, karakteristik warna bawah air, dan kondisi pencahayaan sesuai dengan kondisi inferensi saat operasi. Rekaman video kemudian diekstraksi menjadi kumpulan citra dengan pengambilan setiap beberapa *frame* untuk menghindari duplikasi citra yang berlebihan. Kedua, setiap citra diberi label (*labeling*) berupa kotak pembatas (*bounding box*) dan kelas objek untuk seluruh objek target yang tampak pada citra. Ketiga, model dilatih (*training*) menggunakan *dataset* berlabel tersebut dengan augmentasi data untuk meningkatkan ketahanan model terhadap variasi kondisi citra. Keempat, model hasil dikonversi ke format TensorRT agar inferensi berjalan lebih cepat pada GPU Jetson Orin NX, sehingga memenuhi kebutuhan *real-time* bersamaan dengan proses komputasi lain yang berjalan paralel. Langkah optimasi ini secara efektif memangkas waktu tunda (*latency*) inferensi tanpa menurunkan tingkat akurasi pengenalan objek secara signifikan. Dengan efisiensi pemrosesan tersebut, sistem persepsi visual AUV

mampu mempertahankan laju bingkai (*frame rate*) yang stabil guna mendukung pengambilan keputusan navigasi yang sangat responsif.



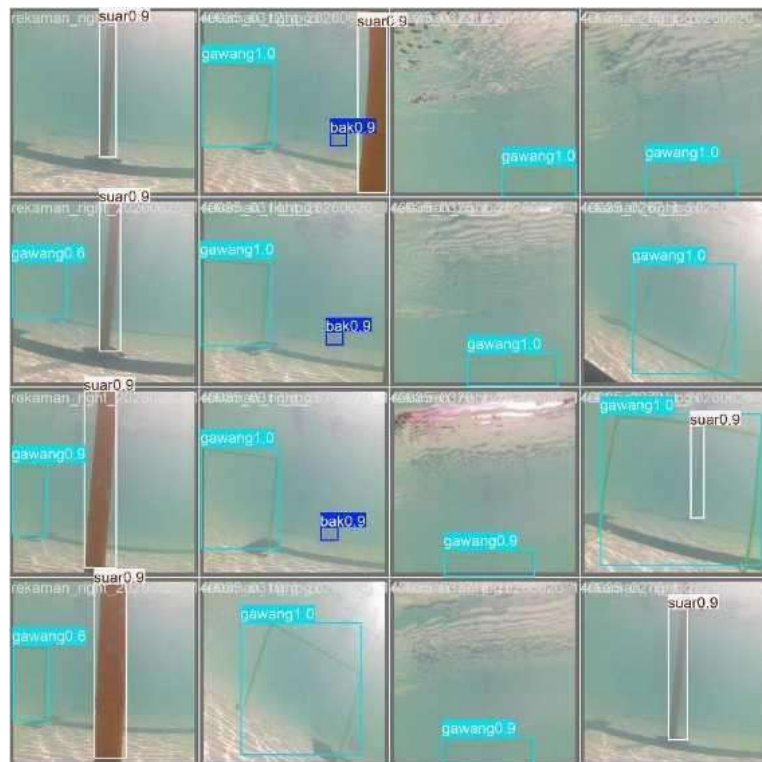
Gambar 3. 12 Contoh Citra Dataset dan Proses *Labeling* Tiga Kelas Objek

(Sumber: Dokumentasi Pribadi)

Gambar 3.12 menunjukkan contoh citra *dataset* beserta proses pemberian label *bounding box* untuk ketiga kelas objek. Setiap objek pada citra ditandai dengan kotak pembatas dan nama kelasnya, yang menjadi data acuan bagi model dalam mempelajari karakteristik visual setiap objek selama proses pelatihan.

Pada saat operasi, inferensi YOLO dijalankan oleh *yolo_perception_node* terhadap citra kamera kiri yang telah terektifikasi. Setiap objek yang terdeteksi menghasilkan *bounding box* beserta nilai keyakinan deteksi (*confidence*) pada rentang 0 hingga 1 yang menyatakan tingkat kepercayaan model terhadap hasil klasifikasi. Deteksi dengan nilai *confidence* di bawah ambang batas yang ditetapkan disaring dan tidak diteruskan ke sistem navigasi, sehingga hanya deteksi yang andal yang digunakan dalam pengambilan keputusan. Apabila terdapat lebih dari satu deteksi untuk kelas yang sama, sistem memilih deteksi dengan nilai *confidence* tertinggi sebagai deteksi yang valid. Selain proses seleksi nilai tertinggi tersebut, algoritma juga menerapkan teknik *Non-Maximum Suppression* (NMS) guna mereduksi tumpang tindih kotak pembatas yang berlebihan pada satu target. Keluaran dari model yang telah tersaring ini kemudian menghasilkan parameter spasial berupa titik pusat koordinat objek di dalam bingkai citra. Titik pusat horizontal dari koordinat tersebut selanjutnya dinormalisasi untuk mengetahui

seberapa jauh simpangan target terhadap sumbu vertikal tengah pandangan AUV. Informasi simpangan lateral ini sangat esensial karena digunakan secara langsung sebagai nilai galat (*error*) dalam sistem kendali *yaw*. Melalui pembacaan nilai galat tersebut, subsistem propulsi dapat memberikan koreksi manuver agar wahana secara presisi berbelok dan meluruskan arah hadapnya menuju target. Pada akhirnya, seluruh data persepsi visual yang telah divalidasi ini didistribusikan ke modul *Mission Manager* untuk direpresentasikan sebagai komponen gaya tarik semu (*attractive force*) dalam metode perancangan lintasan.



Gambar 3. 13 Hasil Deteksi YOLO terhadap Tiga Objek Target Misi

(Sumber: Dokumentasi Pribadi)

Gambar 3.13 menunjukkan contoh hasil deteksi model YOLO terhadap ketiga objek target, dimana setiap objek yang terdeteksi ditandai dengan *bounding box*, nama kelas, dan nilai *confidence*. Hasil deteksi inilah yang selanjutnya digabungkan dengan informasi kedalaman dari *stereo depth* untuk memperoleh estimasi jarak setiap objek, sebagaimana dijelaskan pada subbab 3.3.5.

3.3.5 Penggabungan Deteksi Objek dengan kedalaman

Tahap akhir *pipeline* persepsi adalah menggabungkan hasil deteksi objek YOLO dengan informasi kedalaman hasil komputasi *stereo depth*. Kedua sumber

informasi ini bersifat saling melengkapi deteksi YOLO memberikan identitas dan posisi objek dalam *frame* namun tidak mengetahui jaraknya, sedangkan *depth map* memberikan informasi jarak setiap *piksel* namun tidak mengetahui objek apa yang berada pada *piksel* tersebut. Dengan menggabungkan keduanya, sistem memperoleh informasi lengkap mengenai jenis objek, posisi, dan jaraknya terhadap AUV secara bersamaan.

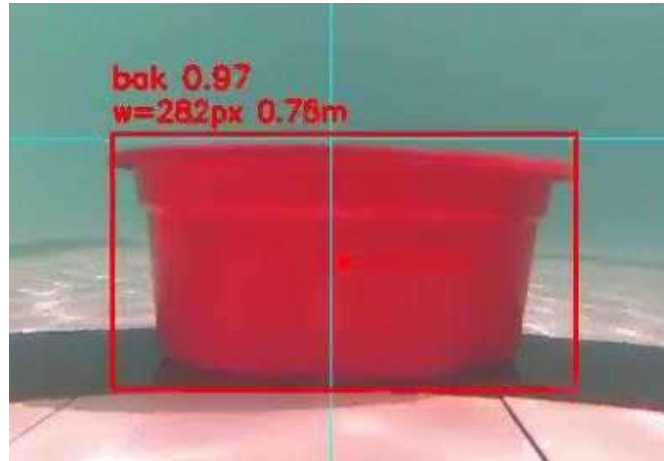
Proses penggabungan dilakukan dengan mengambil nilai kedalaman pada area *bounding box* hasil deteksi YOLO di *depth map*. Karena kedua proses bekerja pada citra kiri terektifikasi yang sama, koordinat *bounding box* pada citra deteksi bersesuaian langsung dengan koordinat pada *depth map* tanpa memerlukan transformasi tambahan. Nilai jarak objek ditentukan dari nilai statistik *piksel-piksel* kedalaman yang valid di dalam area *bounding box*, bukan dari satu *piksel* tunggal, untuk mengurangi pengaruh *noise* dan *piksel* yang gagal terestimasi. Nilai jarak yang diperoleh selanjutnya telah melalui koreksi kalibrasi bawah air sehingga merepresentasikan jarak sebenarnya di dalam air.

Selain jarak, posisi lateral objek dalam *frame* dinormalisasi menjadi nilai *cx_norm* pada rentang -1 hingga $+1$, dimana nilai nol menunjukkan objek berada tepat di tengah *frame*, nilai negatif menunjukkan objek berada di sisi kiri, dan nilai positif di sisi kanan. Normalisasi ini menjadikan informasi posisi tidak bergantung pada resolusi citra dan dapat langsung digunakan oleh sistem navigasi sebagai *error* pelurusan arah hadap (*heading*) semakin besar nilai mutlak *cx_norm*, semakin besar koreksi *yaw* yang diperlukan agar AUV menghadap tepat ke objek target. Hasil akhir persepsi untuk setiap kelas objek dipublikasikan melalui *topic* */perception/gate*, */perception/buoy*, dan */perception/bin* dalam format larik tujuh elemen:

$$[cx, cy, dist, w, cx_{norm}, cy_{norm}, conf] \quad (3.3)$$

dengan *cx* dan *cy* menyatakan koordinat *piksel* titik tengah *bounding box*, *dist* adalah jarak objek hasil penggabungan dengan *depth map* (meter), *w* adalah lebar *bounding box* (*piksel*), *cx_norm* dan *cy_norm* adalah posisi ternormalisasi, serta *conf* adalah nilai keyakinan deteksi. Apabila suatu objek tidak terdeteksi pada *frame* tertentu, sistem mempublikasikan larik dengan nilai penanda kosong agar konsumen *topic* dapat membedakan kondisi objek tidak terdeteksi dari kondisi deteksi yang valid. Dengan format ini, *mission_manager_node* memperoleh

seluruh informasi persepsi yang dibutuhkan untuk pengambilan keputusan navigasi identitas objek dari klasifikasi YOLO, jarak objek dari *stereo depth*, posisi lateral untuk koreksi *heading*, serta tingkat keyakinan untuk menyaring deteksi yang tidak andal.



Gambar 3. 14 Hasil Penggabungan Deteksi YOLO dengan Informasi Kedalaman
(Sumber: Dokumentasi Pribadi)

Gambar 3.14 menunjukkan hasil akhir *pipeline* persepsi, dimana setiap objek yang terdeteksi ditampilkan dengan *bounding box*, nama kelas, nilai *confidence*, serta estimasi jarak hasil penggabungan dengan *depth map*. Informasi jarak yang tampil pada setiap objek membuktikan bahwa kedua sumber persepsi telah terintegrasi objek tidak hanya dikenali jenisnya, tetapi juga diketahui posisinya terhadap AUV secara *real-time*.

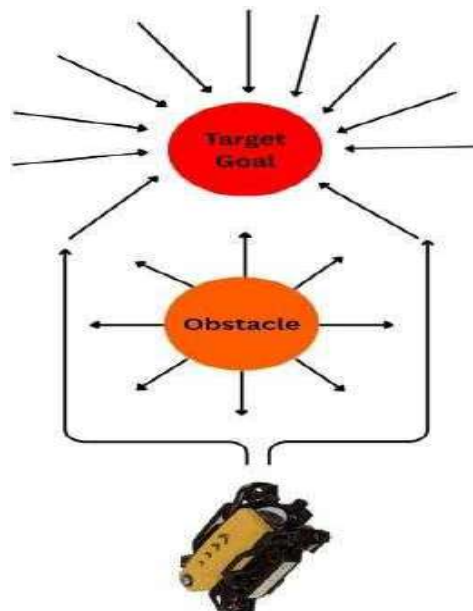
3.3.6 Representasi Lingkungan Lokal Berbasis Zona Kedalaman Lingkungan AUV

Representasi lingkungan lokal berbasis zona kedalaman lingkungan merupakan representasi internal kondisi lingkungan di sekitar Autonomous Underwater Vehicle (AUV) yang dibangun berdasarkan hasil persepsi sensor. Representasi lingkungan lokal berbasis zona kedalaman berfungsi sebagai jembatan antara sistem persepsi dan sistem perencanaan lintasan, sehingga AUV dapat memahami posisi relatifnya terhadap target serta rintangan yang ada di lingkungan.

Pada penelitian ini, Representasi lingkungan lokal berbasis zona kedalaman dibangun menggunakan dua sumber informasi persepsi yang saling melengkapi. Sumber pertama adalah zona jarak rintangan hasil komputasi *stereo depth*, yang

merepresentasikan keberadaan dan jarak rintangan pada setiap arah relatif terhadap AUV informasi ini menjadi dasar perhitungan gaya tolak semu. Sumber kedua adalah hasil deteksi objek YOLO dalam format larik tujuh elemen sebagaimana dijelaskan pada subbab 3.3.5, yang memberikan identitas objek target, jaraknya, serta posisi lateralnya (cx_norm) dalam *frame* kamera informasi ini menjadi dasar penentuan arah gaya tarik semu menuju target misi yang aktif. Kedua informasi tersebut dikelola oleh *mission_manager_node* dan diperbarui pada setiap siklus komputasi, sehingga representasi lingkungan lokal berbasis zona kedalaman selalu merepresentasikan kondisi lingkungan terkini. Representasi ini tidak membentuk peta global yang kompleks, melainkan bersifat lokal dan dinamis sesuai dengan kondisi lingkungan yang terdeteksi secara *real-time*.

Pendekatan representasi lingkungan lokal berbasis zona kedalaman lokal ini dipilih karena lebih ringan secara komputasi dan sesuai untuk sistem yang beroperasi dengan keterbatasan sumber daya pada platform *embedded* seperti Jetson Orin NX. Selain itu, sifat lokal dan dinamis dari representasi lingkungan lokal berbasis zona kedalaman memungkinkan AUV merespons perubahan lingkungan secara langsung tanpa perlu memperbarui peta global yang membutuhkan waktu komputasi lebih besar.



Gambar 3. 15 Ilustrasi Representasi Lingkungan Lokal Berbasis Zona Kedalaman AUV pada Kerangka Artificial Potential Field (APF)

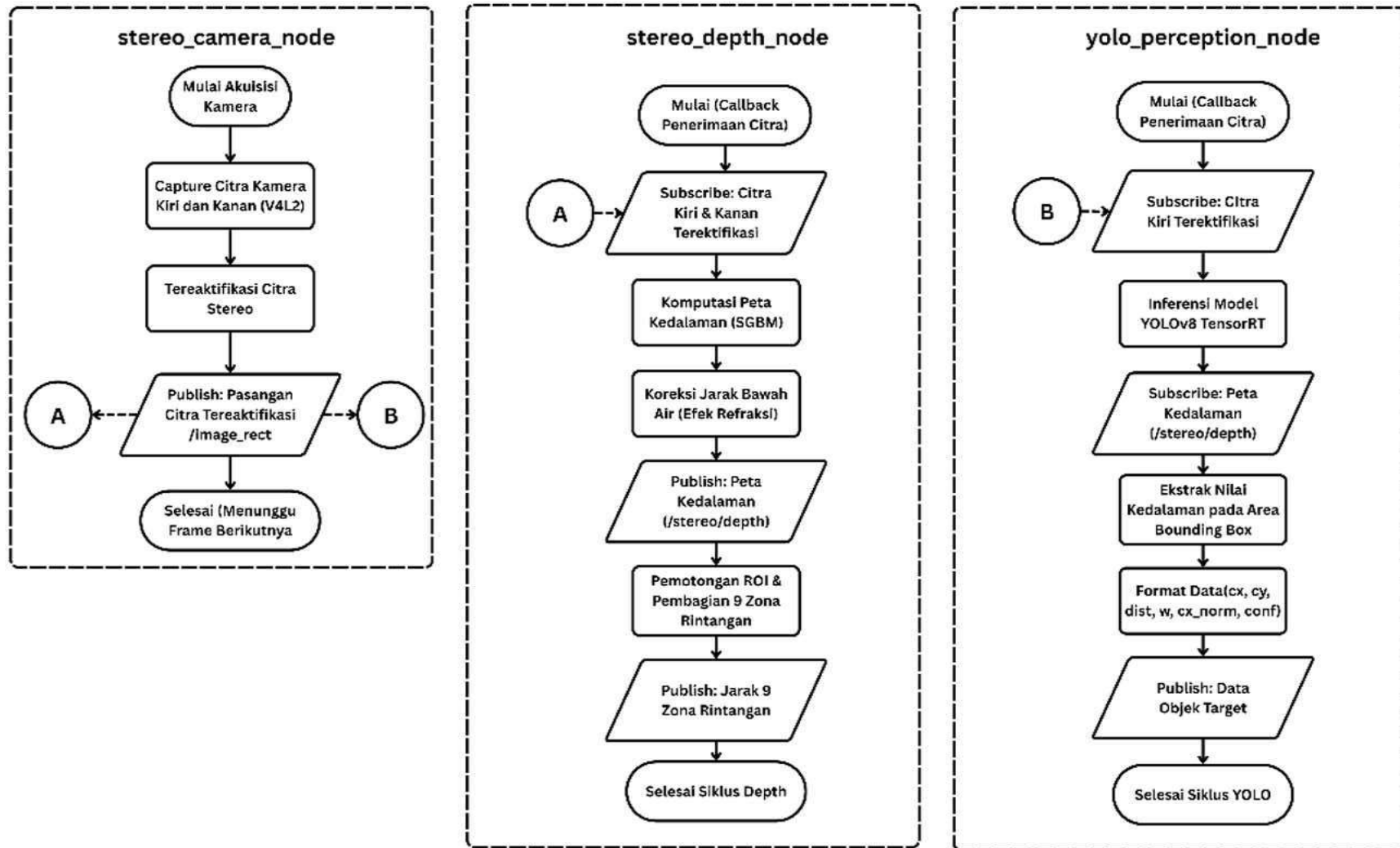
(Sumber : Dokumentasi Pribadi)

Gambar 3.15 menunjukkan ilustrasi konsep representasi lingkungan lokal berbasis zona kedalaman yang digunakan dalam penelitian ini. Pada gambar tersebut, objek target misi yang teridentifikasi oleh YOLO direpresentasikan sebagai sumber gaya tarik semu (*attractive force*) yang menarik AUV menuju titik tujuan, sedangkan zona rintangan yang terdeteksi oleh *stereo depth* direpresentasikan sebagai sumber gaya tolak semu (*repulsive force*) yang mendorong AUV menjauhi area berbahaya. AUV berada pada pusat sistem koordinat lokal, sementara gaya-gaya yang dihasilkan oleh target dan rintangan dihitung berdasarkan posisi relatif yang tersimpan di dalam representasi lingkungan lokal berbasis zona kedalaman.

Representasi lingkungan lokal berbasis zona kedalaman ini selanjutnya digunakan sebagai masukan utama bagi algoritma perencanaan lintasan lokal berbasis Artificial Potential Field (APF). Dengan adanya representasi lingkungan lokal berbasis zona kedalaman, sistem tidak hanya mengetahui keberadaan rintangan, tetapi juga mampu menentukan arah dan intensitas gaya yang memengaruhi pergerakan AUV. Pendekatan ini memungkinkan AUV mengambil keputusan navigasi secara adaptif, stabil, dan responsif terhadap perubahan lingkungan di sekitarnya tanpa memerlukan peta lingkungan global yang lengkap.

3.3.7 Alur Kerja Node Persepsi

Sistem persepsi AUV dijalankan oleh tiga node yang bekerja secara berantai, yaitu *stereo_camera_node*, *stereo_depth_node*, dan *yolo_perception_node*. Alur kerja ketiga *node* tersebut ditunjukkan pada Gambar 3.16. *Node stereo_camera_node* mengawali proses ini dengan mengakuisisi data citra mentah dari kedua lensa secara simultan dan melakukan tahap rektifikasi untuk mengeliminasi distorsi optik. Citra yang telah dikalibrasi tersebut kemudian didistribusikan menuju *stereo_depth_node* untuk diproses lebih lanjut menggunakan algoritma pencocokan blok guna mengekstraksi peta kedalaman spasial. Pada komputasi yang berjalan paralel, bingkai citra dari kamera utama juga dikonsumsi oleh *yolo_perception_node* untuk mendeteksi serta mengklasifikasikan objek target misi yang berada di dalam area pandang. Keluaran informasi berupa jarak rintangan dan koordinat target dari kedua *subsistem* persepsi ini selanjutnya difusikan ke dalam satu kerangka data terpadu.



Gambar 3. 16 Alur Kerja *Node* Persepsi pada Sistem AUV

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 3.16, proses persepsi dimulai pada `stereo_camera_node` yang mengakuisisi citra dari kamera kiri dan kanan melalui antarmuka Video4Linux2 (V4L2) secara bersamaan. Pasangan citra tersebut kemudian direktifikasi menggunakan parameter hasil kalibrasi stereo sehingga garis *epipolar* kedua citra menjadi sejajar horizontal. Hasil rektifikasi dipublikasikan pada topik `/image_rect` dan menjadi masukan bagi dua node berikutnya secara paralel, yaitu `stereo_depth_node` melalui penghubung A dan `yolo_perception_node` melalui penghubung B. Setelah publikasi selesai, node kembali menunggu bingkai citra berikutnya sehingga proses berlangsung terus-menerus.

Pada `stereo_depth_node`, pasangan citra terekstifikasi diproses menggunakan algoritma *Semi-Global Block Matching* (SGBM) untuk menghasilkan peta kedalaman. Nilai kedalaman hasil komputasi kemudian dikoreksi menggunakan persamaan koreksi jarak bawah air guna mengompensasi efek refraksi air, lalu dipublikasikan pada topik `/stereo/depth`. Peta kedalaman tersebut selanjutnya dipotong pada wilayah yang diamati (*region of interest*) dan dibagi menjadi sembilan zona rintangan, kemudian jarak setiap zona dipublikasikan sebagai keluaran akhir node ini. Pemisahan antara publikasi peta kedalaman dan publikasi jarak sembilan zona dilakukan agar peta kedalaman tetap dapat digunakan node lain, sedangkan jarak per zona langsung dapat dipakai algoritma APF tanpa perlu mengolah kembali seluruh citra.

Pada `yolo_perception_node`, citra kiri terekstifikasi diproses melalui inferensi model YOLOv8 yang telah dioptimasi menggunakan TensorRT untuk memperoleh kotak pembatas (*bounding box*) ketiga kelas objek target. Node ini juga berlangganan peta kedalaman dari topik `/stereo/depth`, kemudian mengekstrak nilai kedalaman pada area di dalam setiap kotak pembatas. Hasil penggabungan tersebut diformat menjadi data objek target yang memuat koordinat pusat objek (*cx* dan *cy*), jarak objek (*dist*), lebar kotak pembatas (*w*), posisi horizontal ternormalisasi (*cx_norm*), dan nilai kepercayaan deteksi (*conf*), lalu dipublikasikan sebagai data objek target. Dengan struktur ini, informasi identitas objek dari YOLO dan informasi jarak dari *stereo depth* tergabung dalam satu keluaran yang siap digunakan oleh mesin keadaan misi.

3.4 Perencanaan Lintasan dan *Obstacle Avoidance*

Subbab ini menjelaskan metode perencanaan lintasan dan *obstacle avoidance* yang diterapkan pada sistem AUV, meliputi konsep perencanaan lintasan lokal serta implementasi algoritma Artificial Potential Field (APF).

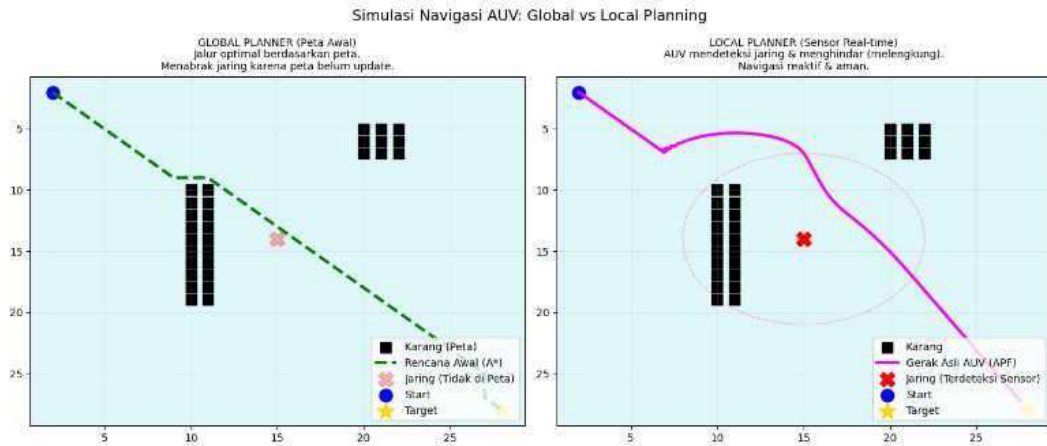
3.4.1 Perencanaan Lintasan Lokal

Perencanaan lintasan lokal merupakan proses penentuan arah dan kecepatan gerak robot secara langsung berdasarkan kondisi lingkungan di sekitar robot yang diperoleh melalui sensor secara *real-time*. Berbeda dengan perencanaan lintasan global yang memanfaatkan peta lingkungan untuk merencanakan jalur secara menyeluruh sebelum robot bergerak, perencanaan lintasan lokal bersifat reaktif dan adaptif terhadap perubahan lingkungan yang dinamis. Pendekatan ini sangat penting bagi Autonomous Underwater Vehicle (AUV) yang beroperasi pada lingkungan bawah air, di mana informasi peta global sering kali tidak tersedia atau tidak lengkap.

Dalam konteks navigasi AUV, perencanaan lintasan lokal berperan sebagai komponen utama dalam sistem *obstacle avoidance*. AUV secara kontinu menerima informasi lingkungan dari sistem persepsi, seperti data kedalaman hasil pengolahan kamera *stereo*, untuk mendeteksi keberadaan rintangan di sekitarnya. Informasi tersebut kemudian digunakan untuk menentukan keputusan navigasi secara instan, termasuk perubahan arah dan kecepatan gerak, sehingga AUV dapat menghindari rintangan sambil tetap bergerak menuju target yang dituju. Pada penelitian ini, metode yang diimplementasikan untuk menjalankan fungsi perencanaan reaktif tersebut adalah algoritma Artificial Potential Field (APF). Algoritma ini mengonversi representasi ruang rintangan dari sensor kedalaman menjadi medan gaya tolak semu, sementara posisi target diformulasikan sebagai pusat gaya tarik semu. Interaksi matematis dari kedua gaya buatan ini secara kontinu menghasilkan vektor resultan yang memandu pergerakan wahana untuk bermanuver secara mulus dan aman dari risiko tabrakan.

Karakteristik utama dari perencanaan lintasan lokal adalah kemampuannya untuk merespons rintangan yang muncul secara tiba-tiba, baik yang bersifat statis maupun dinamis. Lintasan yang dihasilkan umumnya bersifat lebih fleksibel dan tidak kaku, serta dapat berubah sewaktu-waktu mengikuti kondisi lingkungan

aktual. Hal ini menjadikan perencanaan lintasan lokal sangat sesuai untuk diterapkan pada sistem navigasi AUV yang harus beroperasi secara mandiri dalam lingkungan bawah air yang kompleks dan tidak terstruktur.



Gambar 3. 17 Ilustrasi perbedaan perencanaan lintasan global dan perencanaan lintasan lokal pada sistem navigasi AUV

(Sumber : Dokumentasi Pribadi)

Ilustrasi perbedaan antara perencanaan lintasan global dan perencanaan lintasan lokal ditunjukkan pada Gambar 3.17 Pada perencanaan lintasan global, jalur menuju target direncanakan berdasarkan peta awal dan bersifat optimal secara geometris, namun kurang adaptif terhadap keberadaan rintangan baru. Sebaliknya, perencanaan lintasan lokal memanfaatkan informasi sensor secara *real-time* untuk menyesuaikan lintasan AUV secara dinamis, sehingga jalur yang dihasilkan lebih aman dan mampu menghindari rintangan yang terdeteksi selama pergerakan.

Berdasarkan pertimbangan tersebut, penelitian ini memfokuskan pada penggunaan perencanaan lintasan lokal sebagai pendekatan utama dalam sistem navigasi AUV. Perencanaan lintasan lokal digunakan untuk menghasilkan respon gerak yang adaptif dan *real-time*, yang selanjutnya diintegrasikan dengan algoritma *obstacle avoidance* berbasis Artificial Potential Field (APF) untuk menentukan arah dan kecepatan gerak AUV secara aman dan efisien.

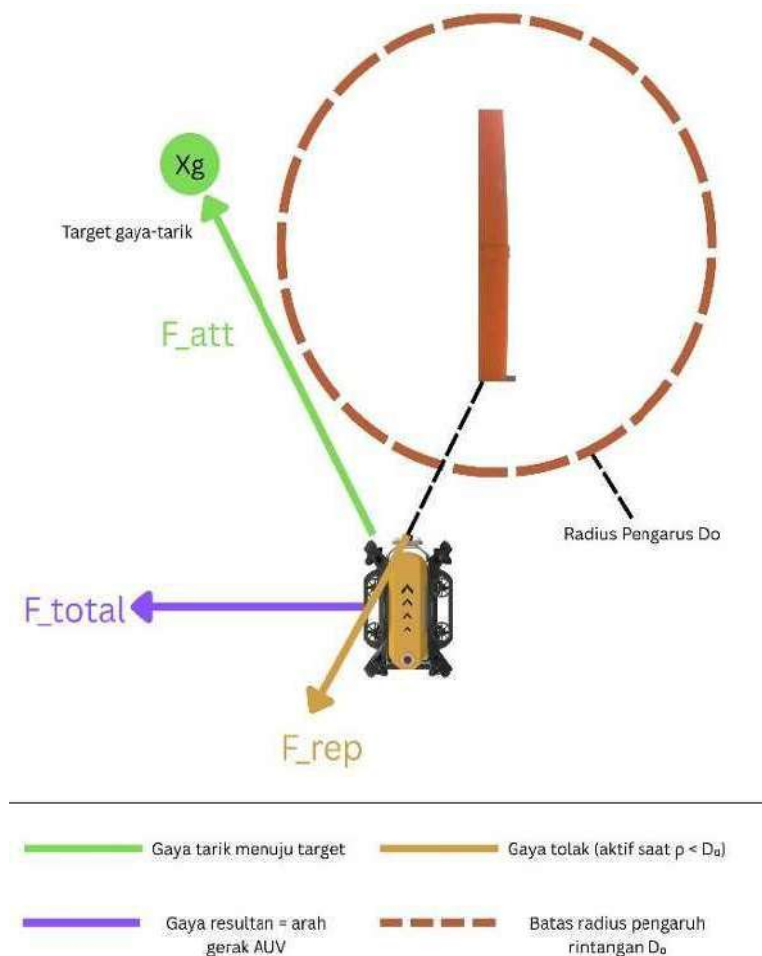
3.4.2 *Obstacle Avoidance* Menggunakan Artificial Potential Field (APF)

Artificial Potential Field (APF) merupakan metode perencanaan lintasan lokal yang banyak digunakan pada navigasi robot otonom karena struktur algoritmanya sederhana, komputasinya ringan, dan mampu menghasilkan respons secara *real-time*. Pada metode ini lingkungan dimodelkan sebagai kombinasi

medan gaya tarik semu dan gaya tolak semu. Dalam penelitian ini AUV diperlakukan sebagai partikel yang bergerak di dalam medan potensial tersebut, dimana titik tujuan menghasilkan gaya tarik semu (*attractive force*) yang menarik AUV menuju target, sedangkan setiap rintangan menghasilkan gaya tolak semu (*repulsive force*) yang menjauhkan AUV dari potensi tabrakan. Gaya tarik semu dirancang sebanding dengan jarak antara AUV dan target, sehingga semakin jauh posisi AUV dari target semakin besar gaya yang mendorongnya menuju tujuan.

Berbeda dengan gaya tarik semu, gaya tolak semu hanya bekerja ketika AUV berada di dalam jarak tertentu dari rintangan yang disebut radius pengaruh (*influence radius*). Di luar radius tersebut gaya tolak semu bernilai nol sehingga tidak memengaruhi pergerakan AUV, sedangkan di dalamnya gaya tolak semu meningkat seiring berkurangnya jarak. Arah dan kecepatan gerak AUV ditentukan oleh gaya resultan semu, yaitu hasil penjumlahan vektor antara gaya tarik semu dan gaya tolak semu, sehingga AUV dapat bergerak menuju target sekaligus menghindari rintangan secara simultan tanpa memerlukan perencanaan lintasan global. Pendekatan kalkulasi vektor secara kontinu ini memastikan bahwa wahana selalu menjaga margin keamanan yang dinamis meskipun menghadapi objek penghalang yang tidak terpetakan sebelumnya. Pada tahap implementasi, nilai matematis dari gaya resultan semu tersebut kemudian ditransformasikan ke dalam parameter kecepatan translasi dan rotasi agar dapat dieksekusi secara langsung oleh subsistem propulsi. Proses transformasi ini melibatkan penguraian vektor gaya resultan semu menjadi komponen sumbu longitudinal untuk mengatur kecepatan maju dan sumbu lateral untuk mengendalikan laju belok AUV. Manuver penghindaran ini dikalibrasi agar tidak terlalu agresif, sehingga osilasi gerak wahana dapat diminimalisir dan stabilitas pembacaan kedalaman oleh kamera stereo tetap terjaga. Selain dinamika gerak, implementasi metode APF ini juga dihadapkan pada kelemahan inheren berupa risiko terjebaknya wahana pada titik minimum lokal (*local minima*). Kondisi jebakan ini terjadi apabila gaya tolak semu rintangan dan gaya tarik semu target saling berlawanan arah dengan besaran yang sama, sehingga gaya resultan semu menjadi nol dan menghentikan pergerakan AUV. Untuk mengatasi anomali tersebut, sistem navigasi dibekali dengan mekanisme penyuntikan gaya virtual (*perturbation force*) secara temporer guna

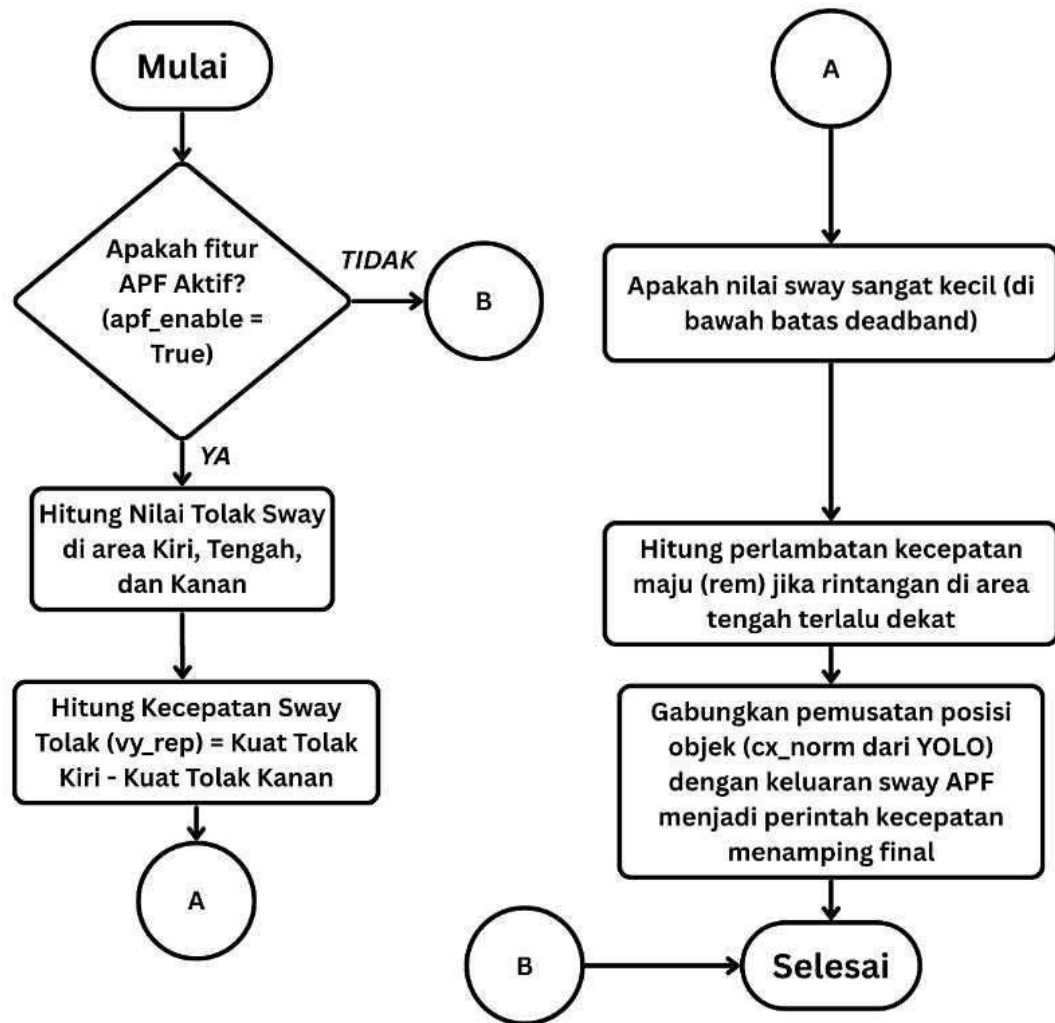
mendorong wahana keluar dari titik ekuilibrium menuju target. Visualisasi ketiga gaya tersebut ditunjukkan pada Gambar 3.18.



Gambar 3. 18 Visualisasi medan gaya Artificial Potential Field (APF)

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 3.18, gaya tarik semu F_{att} mengarah dari posisi AUV menuju titik target X_g , sedangkan gaya tolak semu F_{rep} mengarah menjauhi rintangan dan hanya aktif ketika rintangan berada di dalam batas radius pengaruh D_0 yang digambarkan sebagai garis putus-putus. Penjumlahan vektor kedua gaya menghasilkan gaya resultan semu F_{total} yang menjadi arah gerak AUV, yaitu arah yang menyimpang dari garis lurus menuju target agar AUV tetap menjauhi rintangan. Pada penelitian ini gaya tolak semu diperoleh dari informasi jarak rintangan hasil komputasi *stereo depth*, sedangkan gaya tarik semu diperoleh dari posisi horizontal objek target hasil deteksi YOLO. Alur perhitungan algoritma APF yang diimplementasikan ditunjukkan pada Gambar 3.19.



Gambar 3. 19 Diagram alir algoritma Artificial Potential Field (APF)

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 3.19, proses dimulai dengan pemeriksaan parameter *apf_enable*. Apabila parameter bernilai salah, perhitungan nilai tolak *sway* dilewati sehingga kecepatan gerak ditentukan sepenuhnya oleh perintah mesin keadaan misi; percabangan inilah yang menjadi dasar pelaksanaan pengujian pembandingan pada Bab 4. Apabila bernilai benar, sistem menghitung nilai tolak pada wilayah kiri, tengah, dan kanan berdasarkan jarak minimum sembilan zona kedalaman, kemudian menghitung keluaran *sway* sebagai selisih $vy_{rep} = \text{nilai tolak kiri} - \text{nilai tolak kanan}$. Dengan perumusan tersebut rintangan di wilayah kiri menghasilkan nilai positif sehingga AUV didorong ke kanan, dan sebaliknya, sehingga arah penghindaran mengikuti sisi tempat rintangan terdeteksi tanpa perlu ditetapkan lebih dahulu.

Nilai vy_{rep} selanjutnya dibandingkan terhadap batas *deadband* agar AUV tidak bereaksi terhadap fluktuasi pembacaan yang tidak signifikan. Apabila nilai tersebut berada di bawah batas *deadband*, keluaran *sway* dinolkan sehingga wahana tidak melakukan gerakan menyamping. Terlepas dari nilai *sway* tersebut, sistem tetap menghitung perlambatan kecepatan maju ketika rintangan pada wilayah tengah berada terlalu dekat, karena rintangan tepat di depan AUV tidak dapat diatasi hanya dengan gerak menyamping. Pada kondisi wilayah tengah sangat dekat dan salah satu sisi lebih lapang, sistem menambahkan dorongan menyamping ke sisi yang lebih lapang untuk mengatasi kondisi gaya seimbang (*local minimum*).

Tahap akhir adalah penggabungan pemusatan posisi objek target berdasarkan posisi horizontal hasil deteksi (cx_{norm}) dengan keluaran *sway* hasil perhitungan APF menjadi perintah kecepatan menyamping final. Perintah kecepatan tersebut diteruskan ke *node* kinematika, yang menggabungkannya dengan keluaran pengendali PID untuk kedalaman dan arah hadap guna menghasilkan nilai PWM bagi setiap *thruster*. Nilai seluruh parameter yang digunakan dirangkum pada Tabel 3.4.

Tabel 3. 4 Parameter *Algoritma* Artificial Potential Field (APF)

No.	Parameter	Nilai	Fungsi
1	K_{rep}	120	Konstanta penguat gaya tolak semu
2	$D0$	1,8 m	Radius pengaruh; di luar jarak ini gaya tolak semu bernilai nol
3	Batas gaya sway	$\pm 250 \mu s$	Saturasi perintah geser lateral agar manuver tidak berlebihan
4	d_{min}	0,30 m	Batas bawah jarak pada perhitungan agar nilai gaya tolak semu tidak menuju tak hingga
5	$EMA \alpha$	0,4	Konstanta penghalusan pembacaan jarak

No.	Parameter	Nilai	Fungsi
			zona terhadap fluktuasi
6	<i>Deadband</i>	60 μs	Ambang perintah geser; di bawah nilai ini perintah diabaikan
7	Pembagian zona	9 zona (3 x 3)	Pemisahan lateral dan vertikal peta kedalaman
8	Statistik zona	Persentil ke-25	Nilai jarak perwakilan tiap zona
9	Ambang <i>piksel</i> valid	80 <i>piksel</i>	Jumlah minimum <i>piksel</i> agar zona dianggap valid

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 3.4, jarak pengaruh D_0 ditetapkan sebesar 1,8 meter dengan mempertimbangkan lebar lintasan dan kecepatan gerak wahana agar tersedia waktu yang cukup untuk melakukan manuver menyamping. Batas perintah geser sebesar $\pm 250 \mu s$ membatasi simpangan lebar pulsa terhadap titik netral 1500 μs , sehingga lebar pulsa yang dikirimkan ke thruster berada pada rentang 1250 hingga 1750 μs dan manuver menyamping tidak menjadi berlebihan ketika rintangan berada sangat dekat. Batas bawah d_{min} mencegah nilai gaya tolak semu menuju tak hingga pada jarak yang sangat kecil, sementara penghalusan Exponential Moving Average dan *deadband* sebesar 60 μs menjaga agar wahana hanya bereaksi terhadap rintangan yang benar-benar mengancam. Seluruh nilai tersebut ditentukan secara empiris melalui pengujian bertahap pada simulasi dan kolam nyata.

3.5 Sistem Kendali dan Implementasi

Subbab ini menjelaskan bagaimana setpoint gerak hasil perencanaan Artificial Potential Field (APF) direalisasikan menjadi gerakan nyata AUV, melalui pengendali PID, konversi kinematika, dan komunikasi dengan ESP32.

3.5.1 Sistem Kendali AUV

Pada sistem ini, APF berperan sebagai perencana (*planner*) yang menghasilkan *setpoint* gerak arah (*yaw*), kedalaman, dan kecepatan bukan sebagai

penggerak aktuator langsung. *Setpoint* tersebut direalisasikan oleh pengendali PID yang bekerja pada empat derajat kebebasan (*depth, yaw, roll, pitch*).

Prinsip kerjanya mengikuti skema kendali umpan balik tertutup (*closed-loop*): *setpoint* dari APF dibandingkan dengan kondisi aktual AUV dari sensor (kedalaman dari MS5837, orientasi dari IMU HWT905) sehingga diperoleh sinyal error. Pengendali PID mengoreksi error tersebut secara kontinu agar AUV mengikuti *setpoint* secara halus dan stabil, meredam gangguan seperti arus air atau ketidakseimbangan gaya dorong. Keluaran PID selanjutnya dikonversi oleh perhitungan kinematika menjadi delapan nilai PWM sesuai konfigurasi *thruster* (empat horizontal untuk gerak maju dan belok, empat vertikal untuk kedalaman). Diagram blok sistem kendali ditunjukkan pada Gambar 3.18.

Metode PID dipilih karena strukturnya sederhana, ringan secara komputasi, dan telah terbukti andal untuk pengendalian kendaraan bawah air. Penyetelan nilai parameter PID pada penelitian ini dilakukan secara empiris melalui percobaan langsung pada wahana dengan berpedoman pada karakteristik pengaruh masing-masing konstanta sebagaimana dijelaskan pada subbab 2.4. Prosedur penyetelan dimulai dengan menaikkan konstanta proporsional hingga wahana menanggapi perubahan nilai acuan dengan cukup cepat, kemudian menambahkan konstanta derivatif untuk meredam gerak berlebih di sekitar nilai acuan, dan terakhir menambahkan konstanta integral bernilai kecil untuk menghilangkan galat tunak pada penjagaan kedalaman. Pendekatan empiris dipilih karena penurunan fungsi alih wahana memerlukan identifikasi parameter hidrodinamika yang berada di luar lingkup penelitian ini sebagaimana disebutkan pada Batasan Penelitian. Kriteria keberhasilan penyetelan yang digunakan adalah kemampuan wahana mempertahankan kedalaman dan arah hadap secara stabil selama misi berlangsung, sehingga citra stereo yang dihasilkan tetap konsisten dan estimasi jarak objek tidak terganggu oleh osilasi wahana. Kriteria tersebut terpenuhi sebagaimana ditunjukkan pada hasil pengujian Bab 4.

Nilai parameter pengendali PID yang digunakan pada masing-masing sumbu kendali dirangkum pada Tabel 3.5. Seluruh nilai ditentukan secara empiris melalui penyetelan langsung pada wahana.

Tabel 3. 5 Parameter Pengendali PID pada Setiap Sumbu Kendali

No.	Sumbu Kendali	Kp	Ki	Kd	Batas Keluaran
1	Kedalaman (depth)	9,0	0,5	0,5	150
2	Arah hadap (yaw)	4,0	0,35	0,6	300
3	Guling (roll)	1,0	0,01	0,01	300
4	Angguk (pitch)	15,0	0,07	1,2	300

(Sumber : Dokumentasi Pribadi)

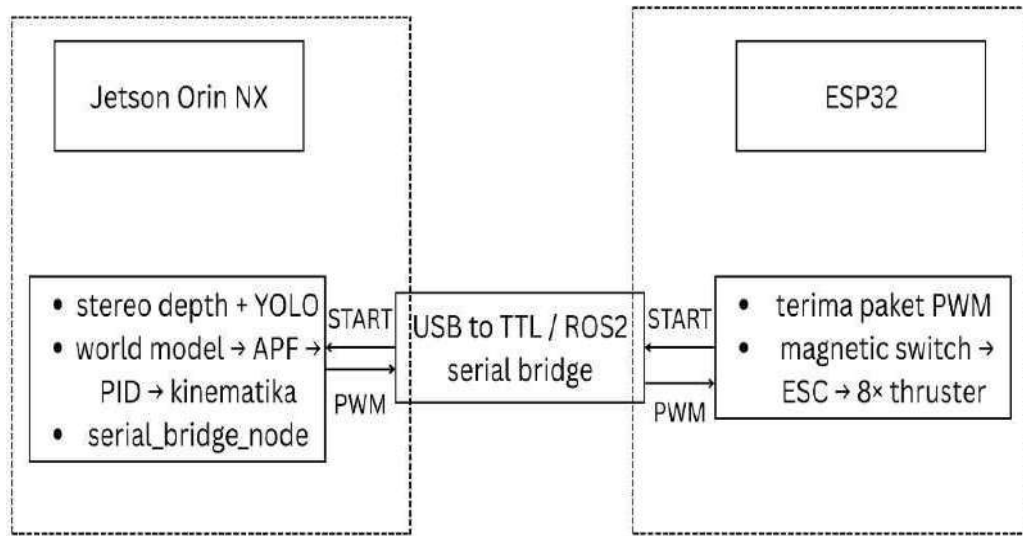
Berdasarkan Tabel 3.5, keluaran pengendali kedalaman dibatasi pada nilai maksimum 150. Pembatasan ini diterapkan karena gerak vertikal berkaitan langsung dengan penjagaan kedalaman, sehingga keluaran yang terlalu besar menyebabkan wahana melewati kedalaman acuan (overshoot) dan memicu osilasi vertikal. Osilasi tersebut tidak hanya mengganggu kestabilan wahana, tetapi juga menyebabkan pergeseran sudut pandang kamera stereo sehingga estimasi jarak objek menjadi tidak konsisten. Sesuai penjelasan pada subbab 2.4, pembatasan keluaran ini sekaligus mencegah terjadinya integral windup pada suku integral pengendali.

3.5.2 Komunikasi Jetson Orin NX dan ESP32

Sistem menggunakan arsitektur terdistribusi dengan pembagian tugas yang tegas. Jetson Orin NX sebagai pengendali tingkat tinggi menjalankan seluruh pemrosesan persepsi visual (*stereo depth* dan YOLO), representasi lingkungan lokal berbasis zona, perencanaan APF, kendali PID, hingga konversi kinematika dan menghasilkan delapan nilai PWM. ESP32 sebagai pengendali tingkat rendah hanya menerima nilai PWM tersebut, meneruskannya ke ESC untuk menggerakkan *thruster*, serta memantau *magnetic switch* secara independen sebagai pemicu mode otonom sekaligus *kill switch*. Penempatan fungsi keselamatan pada ESP32 memastikan *kill switch* tetap bekerja tanpa bergantung pada kondisi Jetson.

Komunikasi berlangsung dua arah melalui antarmuka serial USB to TTL. Pada sisi ROS2, *serial_bridge_node* mengemas nilai PWM dari *topic /thruster/pwm* menjadi paket serial berformat $P\{m_1\}, \dots, \{m_8\}E$, dan menerima

paket status dari ESP32 yang dipublikasikan melalui topic `/serial/feedback`. Diagram komunikasi ditunjukkan pada Gambar 3.20.

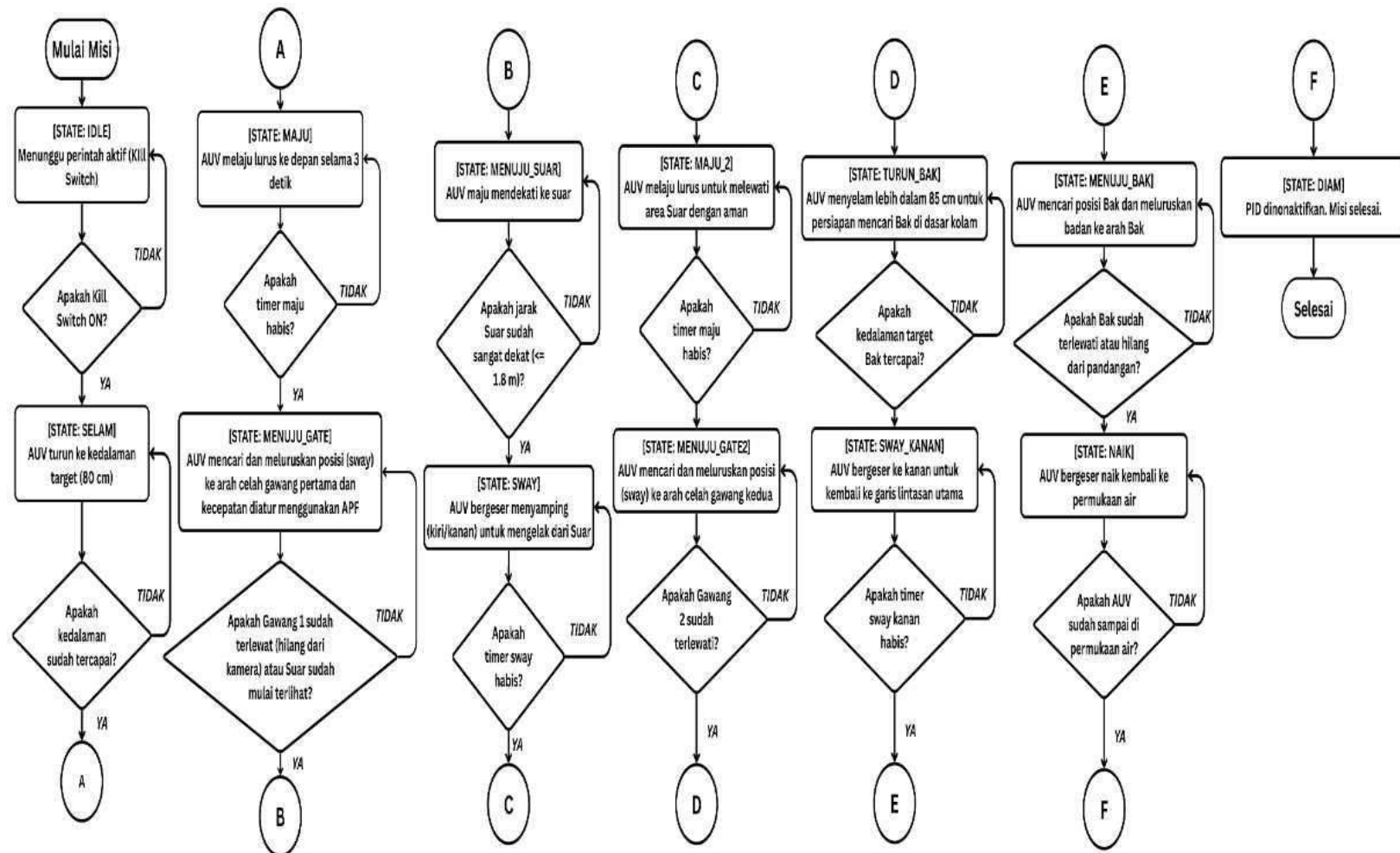


Gambar 3. 20 Diagram komunikasi antara Jetson Orin NX dan ESP32 pada sistem AUV

(Sumber : Dokumentasi Pribadi)

3.5.3 Alur Mesin Keadaan *Mission Manager*

Pelaksanaan misi AUV dikendalikan oleh mesin keadaan (*state machine*) pada *mission_manager_node*. Arsitektur *Finite State Machine* (FSM) ini dipilih untuk mengorkestrasi berbagai fase pergerakan wahana secara sekuensial dan terstruktur. Dalam operasinya, *node* ini secara kontinu memantau status sistem dan data umpan balik persepsi untuk mengevaluasi terpenuhi atau tidaknya parameter kondisi transisi. Sebagai contoh, sistem akan beranjak dari *state* siaga menuju *state* penyelaman, lalu beralih ke mode navigasi lintasan apabila kedalaman operasional telah tercapai. Pada setiap *state* yang aktif, manajer misi akan menyiarkan (*publish*) set instruksi referensi spesifik, seperti target orientasi dan kecepatan linier, kepada subsistem kendali PID di tingkat bawah. Pendekatan logika transisi ini sangat krusial untuk mencegah terjadinya tumpang tindih perintah manuver, sehingga AUV dapat menyelesaikan seluruh skenario misi secara otonom dan aman. Selain itu, FSM ini juga terintegrasi dengan mekanisme pengamanan (*fail-safe*) yang secara otomatis memicu *state* berhenti atau mengambang ke permukaan apabila sistem mendeteksi anomali kritis pada pembacaan sensor. Alur perpindahan antar *state* beserta syarat transisinya ditunjukkan pada Gambar 3.21.



Gambar 3. 21 Diagram Alir Mesin Keadaan *Mission Manager*

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 3.21, misi dimulai dari *state IDLE*, yaitu kondisi wahana menunggu perintah aktif dari *kill switch*. Selama *kill switch* belum aktif, sistem tetap berada pada *state* ini sehingga *thruster* tidak menghasilkan gaya dorong. Ketika *kill switch* aktif, sistem berpindah ke *state SELAM* dan wahana menyelam hingga mencapai kedalaman acuan 80 *sentimeter*. Setelah kedalaman tercapai, sistem masuk ke *state MAJU* yang menggerakkan wahana lurus ke depan selama tiga detik sebagai fase awal meninggalkan titik start.

Selanjutnya sistem masuk ke *state MENUJU_GATE*, dimana wahana mencari celah gawang pertama dan meluruskan posisinya terhadap celah tersebut, sementara kecepatannya diatur oleh algoritma APF. Transisi keluar dari *state* ini terjadi apabila gawang pertama telah hilang dari pandangan kamera atau suar mulai terlihat. Penggunaan dua syarat alternatif tersebut dimaksudkan agar transisi tetap terjadi meskipun salah satu penanda gagal terdeteksi. Sistem kemudian masuk ke *state MENUJU_SUAR* untuk mendekati suar, dan ketika jarak suar telah mencapai 1,8 meter atau lebih dekat, sistem berpindah ke *state SWAY* yang menggerakkan wahana menyamping ke kiri atau ke kanan untuk mengelak dari suar sesuai arah gaya tolak semu APF. Setelah pewaktu *sway* berakhir, sistem masuk ke *state MAJU_2* untuk melintasi area suar, dilanjutkan ke *state MENUJU_GATE2* untuk mencari dan meluruskan posisi

terhadap celah gawang kedua. Ketika gawang kedua telah terlewati, sistem masuk ke *state TURUN_BAK* dan wahana menyelam lebih dalam hingga 85 *sentimeter* sebagai persiapan mencari bak di dasar kolam. Setelah kedalaman sasaran tercapai, *state SWAY_KANAN* menggeser wahana ke kanan untuk kembali ke garis lintasan utama, kemudian *state MENUJU_BAK* mengarahkan wahana pada posisi bak. Apabila bak telah terlewati atau hilang dari pandangan, sistem masuk ke *state NAIK* untuk kembali ke permukaan air, dan setelah wahana mencapai permukaan sistem masuk ke *state DIAM* yang menonaktifkan pengendali PID sebagai tanda misi selesai.

Seluruh transisi pada mesin keadaan ini menggunakan syarat berbasis pewaktu, pembacaan kedalaman, atau status deteksi objek, dan tidak menggunakan konfirmasi posisi absolut karena sistem tidak dilengkapi *odometri* maupun koordinat global. Konsekuensi dari pendekatan ini adalah *state* dapat terus

bertransisi meskipun wahana secara fisik tertahan, sebagaimana teramati pada beberapa *trial* pengujian yang dibahas pada Bab 4.

3.6 Perancangan Sistem

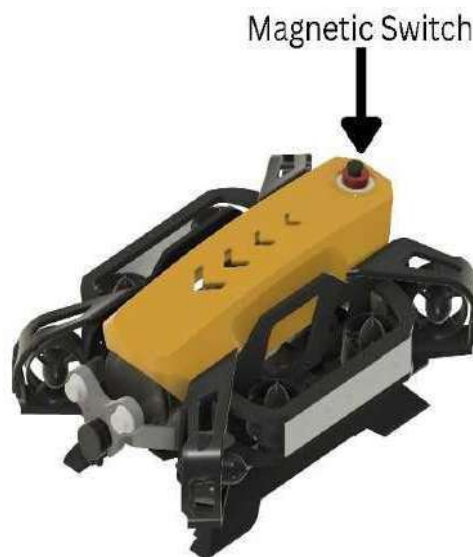
Pada subbab ini membahas bagaimana sistem AUV dirancang secara teknis sebelum diimplementasikan, baik dari sisi mekanik, elektrik, maupun perangkat lunak. Fokusnya adalah desain dan pemilihan rancangan.

3.6.1 Perancangan Sistem Mekanik

Desain prototype Autonomous Underwater Vehicle (AUV) pada penelitian ini atau tugas akhir ini dibuat dengan panduan rules KRBAI. Berikut merupakan ukuran dan dimensi AUV.

1. Panjang : 64,7 cm
2. Lebar : 44,4 cm
3. Tinggi : 28,1 cm
4. Berat : 19 kg

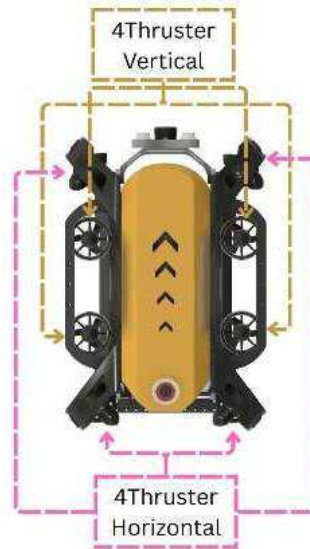
Berikut merupakan desain Autonomous Underwater Vehicle (AUV) yang digunakan pada penelitian ini atau tugas akhir ini



Gambar 3. 22 *Desain AUV tampak Isometri*

(Sumber : Dokumentasi Pribadi)

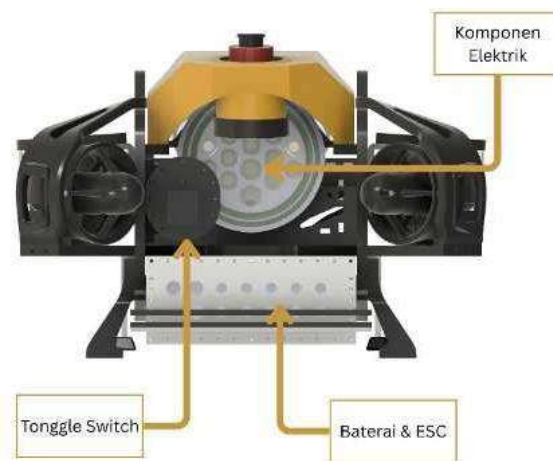
Berdasarkan tampilan isometri pada Gambar 3.22, magnetic switch ditempatkan pada bagian belakang ballast. Sementara itu, pada bagian tengah ballast dibuat lubang berbentuk segitiga yang berfungsi sebagai saluran keluarnya udara ketika AUV melakukan proses penyelaman ke bawah permukaan air.



Gambar 3. 23 *Desain AUV tampak atas*

(Sumber : Dokumentasi Pribadi)

Berdasarkan tampilan atas pada Gambar 3.23, AUV dilengkapi dengan delapan buah *thruster* yang digunakan untuk mendukung kemampuan manuver. Empat buah *thruster* tipe T200 dipasang secara *horizontal* dan berfungsi untuk menghasilkan gerakan translasi AUV ke arah maju, mundur, serong, dan berputar. Sementara itu, empat buah *thruster* T200 lainnya dipasang secara vertikal untuk mendukung pergerakan AUV ke arah naik dan turun.

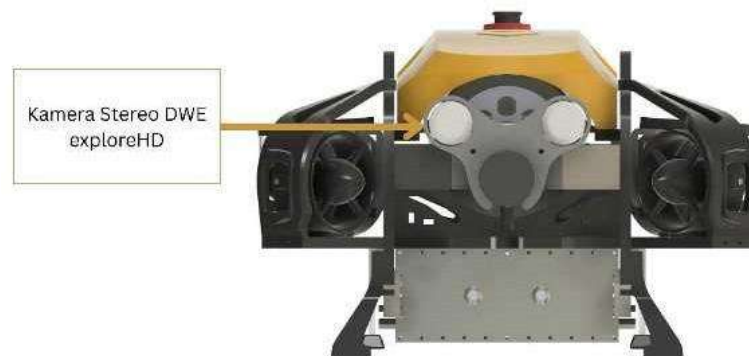


Gambar 3. 24 *Desain AUV tampak belakang*

(Sumber : Dokumentasi Pribadi)

Berdasarkan tampilan belakang pada Gambar 3.24, beberapa komponen utama ditempatkan di dalam *enclosure* utama, antara lain modul ESP32, baterai *Li-*

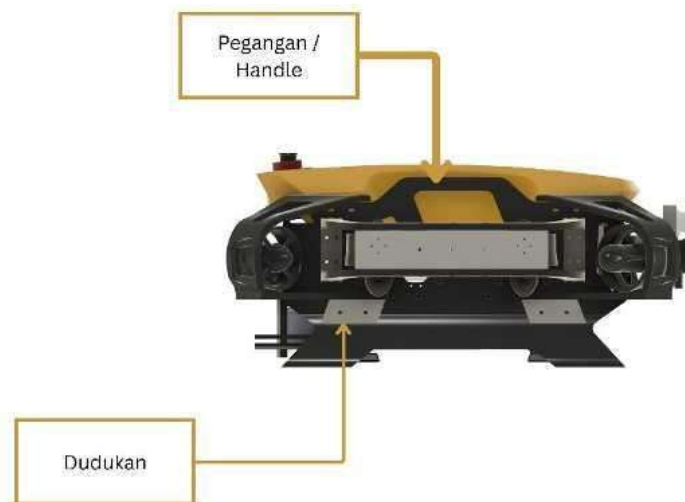
ion 3 sel, sensor IMU, Jetson Orin NX, serta komponen pendukung sistem kelistrikan lainnya. Selain itu, pada *enclosure* bagian bawah ditempatkan komponen seperti *Electronic Speed Controller* (ESC), baterai *Li-ion* 4 sel, dan relay. Pada bagian penutup *enclosure* utama dipasang sensor tekanan (*pressure sensor*) untuk mendukung pemantauan kedalaman. Seluruh sistem yang berada pada *enclosure* utama dan enclosure bawah dikendalikan melalui sebuah *toggle switch* yang dipasang pada bagian belakang AUV.



Gambar 3. 25 *Desain AUV tampak depan*

(Sumber : Dokumentasi Pribadi)

Berdasarkan tampilan depan pada Gambar 3.25, AUV dilengkapi dengan *bracket* kamera depan yang digunakan untuk memasang 2 kamera yang akan dijadikan stereo DWE *exploreHD* .

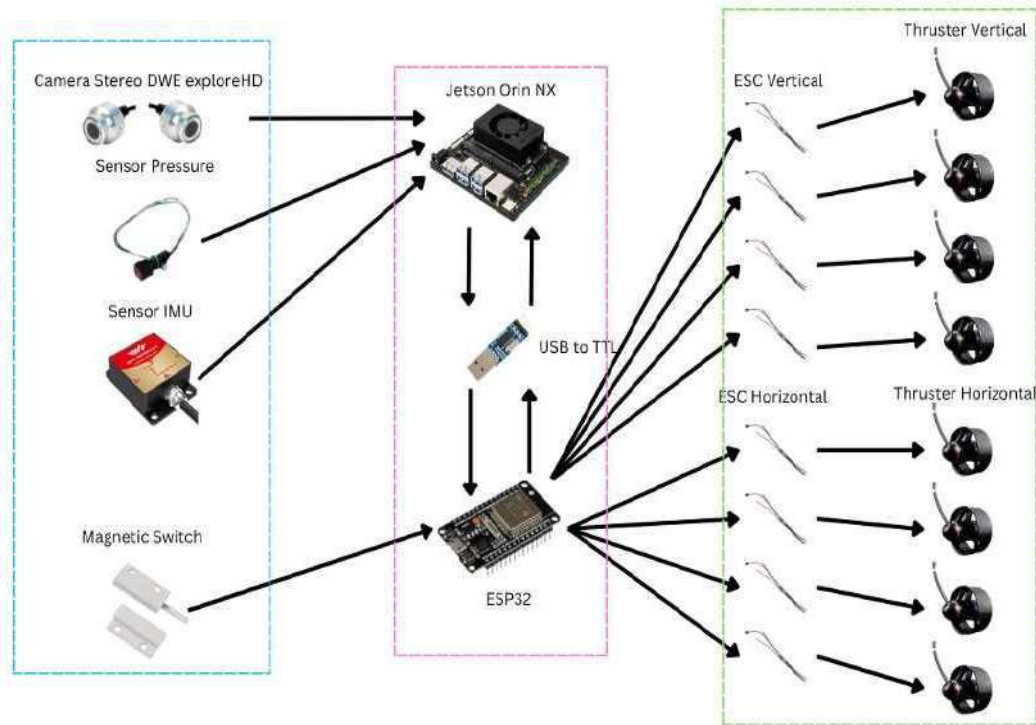


Gambar 3. 26 *Desain AUV tampak samping*

(Sumber : Dokumentasi Pribadi)

Berdasarkan tampilan samping pada Gambar 3.26, AUV dirancang secara efisien dengan dilengkapi handle pegangan yang berfungsi untuk memudahkan proses pengangkatan maupun penurunan AUV selama penanganan di luar air.

3.6.2 Perancangan Sistem Elektrik



Gambar 3. 27 Diagram Blok pada komponen elektrik

(Sumber : Dokumentasi Pribadi)

Pada penelitian ini, perancangan perangkat keras kelistrikan AUV diilustrasikan secara komprehensif pada Gambar 3.27. Sistem elektrik tersebut dirancang secara khusus untuk mengintegrasikan komponen persepsi, unit pemrosesan utama, mikrokontroler tingkat rendah, hingga jajaran aktuator agar seluruh misi berjalan lancar.

Pada subsistem pertama, sistem menerima aliran data dari beberapa komponen sensor utama yang bertindak sebagai indera wahana. Dua unit kamera DWE exploreHD dipasang sejajar pada bagian lambung depan AUV untuk membentuk konfigurasi penglihatan stereo (*stereo pair*). Sepasang kamera ini terhubung langsung ke unit Jetson Orin NX melalui antarmuka USB guna menjamin transmisi data berkecepatan tinggi. Citra visual dari kamera tersebut ditujukan untuk mendukung komputasi peta kedalaman spasial sekaligus deteksi objek target secara *real-time*. Selain komponen visual, sistem ini juga

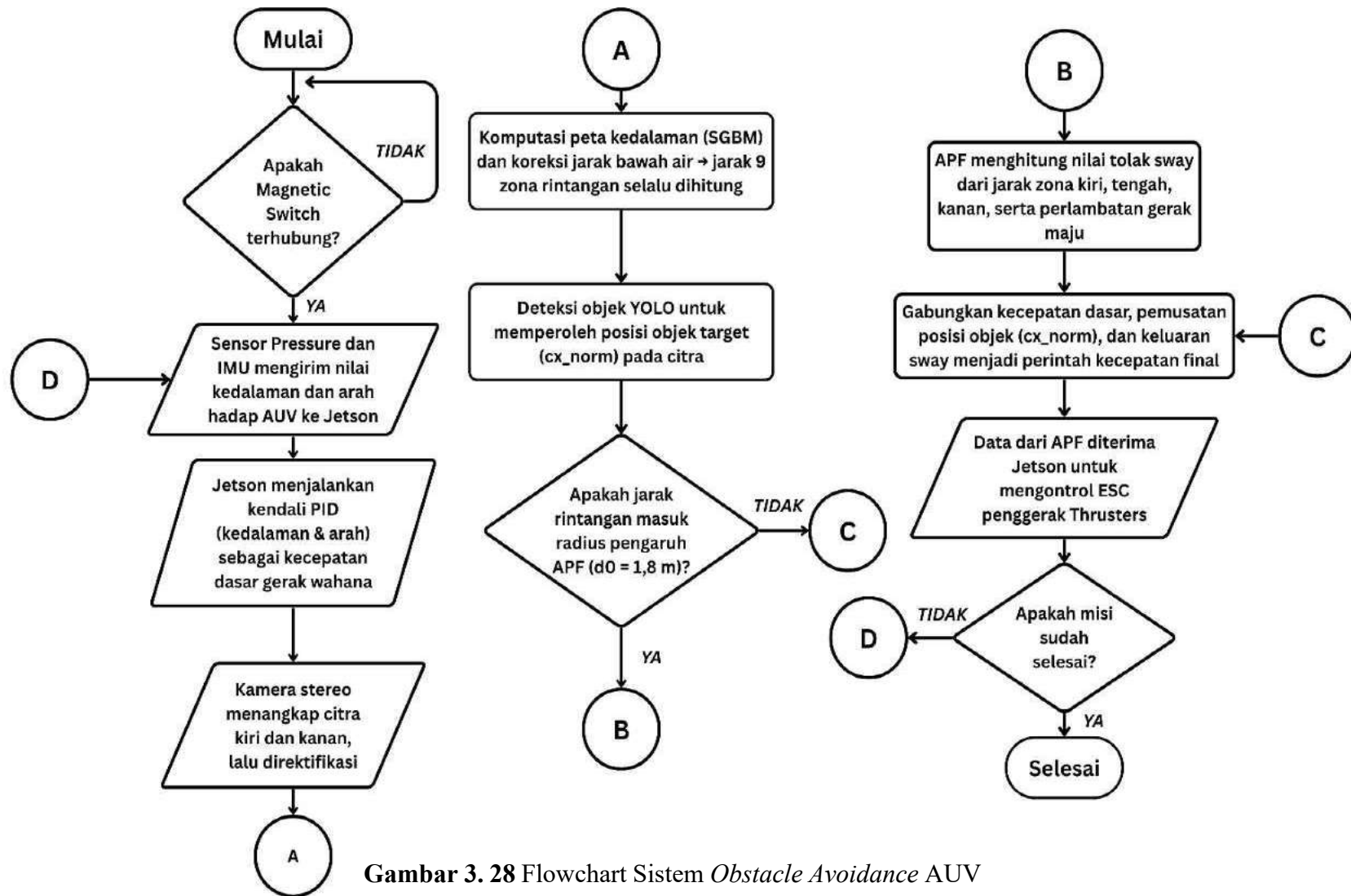
memanfaatkan sensor IMU HWT905 untuk mengakuisisi parameter orientasi wahana berupa sudut *roll*, *pitch*, dan *yaw*. Untuk kebutuhan manuver vertikal, sensor tekanan MS5837 ditugaskan untuk mengukur kedalaman AUV secara akurat saat beroperasi di bawah permukaan air. Seluruh data mentah dari IMU dan sensor tekanan tersebut dialirkan ke Jetson Orin NX agar dapat difusikan dalam algoritma navigasi tingkat tinggi. Di luar komponen sensoris tersebut, terdapat sebuah *magnetic switch* yang dihubungkan secara eksklusif ke ESP32. Sakelar magnetik ini mengemban fungsi ganda sebagai pemicu dimulainya mode operasi otonom sekaligus sebagai *kill switch* darurat untuk memutus daya aktuator secara instan.

Pada tahapan subsistem kedua, Jetson Orin NX mengambil peran sentral sebagai komputer utama yang memikul seluruh beban komputasi berat. Perangkat ini secara simultan mengeksekusi algoritma *stereo depth* dan model deteksi You Only Look Once (YOLO) yang diakselerasi menggunakan TensorRT. Berdasarkan perpaduan informasi visual, koreksi *heading* dari IMU, dan data kedalaman, Jetson menyusun representasi lingkungan lokal yang berbasis pada zona kedalaman. Representasi spasial tersebut kemudian diolah oleh algoritma Artificial Potential Field (APF) untuk menentukan manuver penghindaran rintangan dan pencapaian target. Hasil perhitungan matematis dari APF dan sistem kendali PID selanjutnya dikonversi melalui persamaan kinematika menjadi paket sinyal PWM yang ditransmisikan ke ESP32 melalui antarmuka serial menggunakan jembatan USB to TTL.

Pada subsistem ketiga, ESP32 bertindak sebagai unit kendali tingkat rendah (*low-level controller*) yang didedikasikan murni untuk membangkitkan sinyal operasional tanpa terbebani komputasi navigasi. ESP32 secara presisi meneruskan instruksi PWM tersebut menuju jajaran *Electronic Speed Controller* (ESC) yang berfungsi untuk mendistribusikan suplai arus listrik dari sumber tegangan. ESC kemudian mengontrol kecepatan putaran motor *brushless* pada empat *thruster* horizontal dan empat *thruster* vertikal demi menghasilkan gaya dorong mekanis yang terkoordinasi sesuai perintah pergerakan wahana..

3.6.3 Flowchart Sistem Keseluruhan

Keseluruhan alur kerja sistem *obstacle avoidance* pada AUV, mulai dari inisialisasi hingga misi selesai, ditunjukkan pada Gambar 3.28.



Gambar 3. 28 Flowchart Sistem *Obstacle Avoidance* AUV

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 3.28, sistem diawali dengan pemeriksaan status *magnetic switch*. Selama *magnetic switch* belum terhubung, sistem menahan diri pada tahap pemeriksaan ini sehingga tidak ada perintah penggerak yang dikirimkan. Ketika *magnetic switch* terhubung, sensor tekanan MS5837 dan sensor IMU HWT905 mulai mengirimkan nilai kedalaman dan arah hadap wahana ke Jetson Orin NX. Berdasarkan data tersebut, Jetson menjalankan pengendali PID untuk kedalaman dan arah hadap, yang menghasilkan kecepatan dasar gerak wahana. Secara bersamaan, kamera stereo menangkap pasangan citra kiri dan kanan yang kemudian direktifikasi sebagai masukan bagi tahap berikutnya melalui penghubung A.

Pada tahap kedua, pasangan citra terektifikasi diproses menggunakan algoritma SGBM untuk menghasilkan peta kedalaman, yang selanjutnya dikoreksi terhadap efek refraksi bawah air dan dibagi menjadi sembilan zona rintangan. Perhitungan jarak ini dilakukan pada setiap siklus, sehingga informasi jarak rintangan selalu tersedia. Secara paralel, model YOLO melakukan inferensi untuk memperoleh posisi horizontal objek target (cx_norm) pada citra. Sistem kemudian mengevaluasi apakah jarak rintangan telah masuk ke dalam radius pengaruh algoritma APF, yaitu $d_0 = 1,8$ meter. Apabila rintangan masih berada di luar radius tersebut, tidak ada nilai tolak yang dihitung dan wahana melanjutkan gerak dengan kecepatan dasar melalui penghubung C. Apabila rintangan telah berada di dalam radius pengaruh, alur dilanjutkan ke tahap perhitungan APF melalui penghubung B.

Pada tahap ketiga, algoritma APF menghitung nilai tolak *sway* berdasarkan jarak zona kiri, tengah, dan kanan, serta menghitung perlambatan gerak maju apabila rintangan pada zona tengah berada terlalu dekat. Kecepatan dasar hasil pengendali PID, pemusatan posisi objek target (cx_norm), dan keluaran *sway* hasil perhitungan APF kemudian digabungkan menjadi perintah kecepatan final. Perintah kecepatan tersebut dikonversi menjadi nilai PWM yang dikirimkan ke ESP32, untuk selanjutnya diteruskan ke *Electronic Speed Controller* (ESC) sebagai penggerak kedelapan *thruster*. Setelah perintah gerak dieksekusi, sistem memeriksa apakah misi telah selesai; apabila belum, alur kembali ke tahap pembacaan sensor dan mengulangi keseluruhan siklus.

Perlu ditegaskan bahwa seluruh proses komputasi, meliputi rektifikasi citra, komputasi peta kedalaman, inferensi YOLO, perhitungan APF, dan pengendalian PID, dijalankan sepenuhnya pada Jetson Orin NX, sedangkan ESP32 hanya berperan meneruskan nilai PWM ke ESC dan memantau status *magnetic switch*.

BAB 4

HASIL DAN PEMBAHASAN

4.1 Hasil Perancangan Hardware

4.1.1 Hasil Perancangan Mekanik

Setelah tahap perancangan mekanik diselesaikan pada bab 3, tahap berikutnya adalah realisasi fisik dan pengujian sistem mekanik untuk memastikan seluruh komponen dapat bekerja sesuai fungsinya. Hasil dari perancangan telah direalisasikan dalam bentuk fisik robot dengan mempertahankan dimensi yang sesuai dengan regulasi.



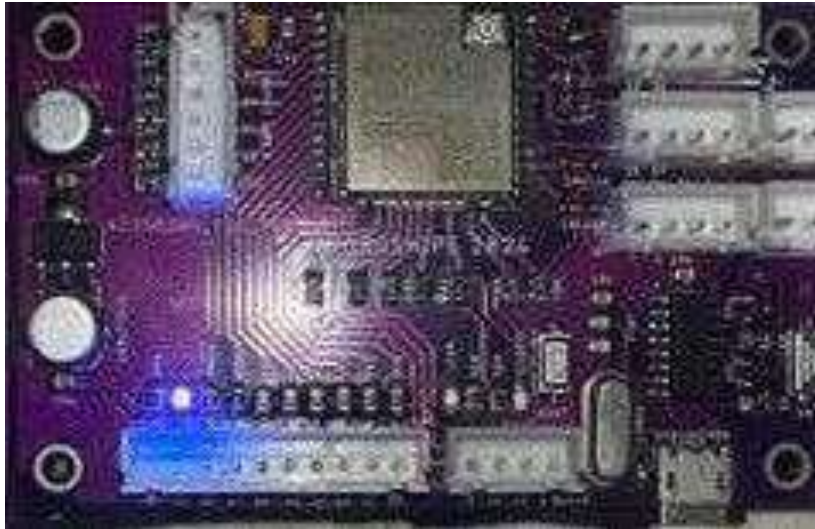
Gambar 4. 1 Tampak depan Robot

(Sumber : Dokumentasi Pribadi)

Pada gambar 4.1 menunjukkan tampak depan robot, posisi pemasangan kamera stereo dan posisi thruster yang simetris.

4.1.2 Hasil Perancangan Sistem Elektrik

Sistem elektrik merupakan bagian penting dari robot karena menghubungkan seluruh sensor, actuator, dan unit pemrosesan. Tujuan utama dari system ini Adalah agar robot dapat bergerak secara otomatis, membaca lingkungan sekitarnya, dan mengambil Keputusan berdasarkan input sensor. Setelah proses perancangan, system elektrik kemudian direalisasikan dalam bentuk fisik dan untuk memastikan seluruh rangkaian bekerja sesuai dengan fungsinya.



Gambar 4. 2 Papan PCB AUV

(Sumber : Dokumentasi Pribadi)

Gambar 4.2 menunjukkan rancangan *Printed Circuit Board* (PCB) yang digunakan untuk mengintegrasikan mikrokontroler, esc, sensor, dan komponen pendukung lainnya. Rancangan PCB ini dibuat agar lebih rapih stabil, dan mempermudah proses perakitan serta perawatan sistem.

Hasil pengujian sistem menunjukkan bahwa semua komponen bekerja sesuai fungsinya. *Thruster* merespons kendali arah dan kecepatan dengan baik, kamera berhasil mendeteksi objek dan jarak dengan akurat. Dengan system ini, AUV mampu menyelesaikan lintasan KRBAI dari posisi *start* hingga *finish*.

4.1.3 Hasil Perancangan ROS 2 Humble

Penelitian ini dibangun menggunakan arsitektur *Robot Operating System 2* (ROS 2) Humble yang menerapkan komunikasi antar node melalui mekanisme topic-based communication. Setiap *node* memiliki fungsi yang berbeda, seperti akuisisi data sensor, komunikasi dengan antara ESP32 dengan Jetson, Melalui mekanisme *publisher-subscriber*, setiap node dapat saling bertukar informasi secara terdistribusi sehingga sistem menjadi lebih modular dan mudah dikembangkan. Selain itu, penggunaan middleware berbasis *Data Distribution Service* (DDS) pada ROS 2 memungkinkan komunikasi data berlangsung dengan latensi rendah sehingga sesuai untuk kebutuhan AUV. Alur komunikasi antar *node* divisualisasikan menggunakan *RQT Graph*, yang menampilkan hubungan antara seluruh *node* beserta *topic* yang digunakan selama sistem beroperasi.



(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.3, *graf node* ROS2 menampilkan sebelas *node* aktif beserta aliran data antar *node* melalui mekanisme *publish-subscribe*, dengan struktur yang sesuai terhadap rancangan arsitektur pada Gambar 3.3. Alur persepsi terlihat dari *stereo_camera_node* yang mempublikasikan pasangan citra terektifikasi melalui *topic* */stereo/left/image_rect* dan */stereo/right/image_rect*, yang kemudian dikonsumsi secara paralel oleh *stereo_depth_node* dan *yolo_perception_node*. *Node stereo_depth_node* mempublikasikan peta kedalaman melalui */stereo/depth* yang menjadi masukan fusi jarak bagi *yolo_perception_node*, serta zona jarak rintangan melalui */obstacle/zone_distances* yang menjadi masukan perintah geser APF pada *mission_manager_node*. *Node yolo_perception_node* mempublikasikan hasil deteksi ketiga objek target melalui *topic* */perception/gate*, */perception/buoy*, dan */perception/bin* yang seluruhnya mengalir ke *mission_manager_node*. Pada sisi sensor, *witmotion_imu_node* mempublikasikan data orientasi melalui *topic* */imu/yaw*, */imu/pitch*, dan */imu/roll*, sedangkan *depth_sensor_node* mempublikasikan data kedalaman melalui */depth/data*; kedua kelompok data ini dikonsumsi baik oleh *mission_manager_node* untuk kebutuhan representasi lingkungan lokal berbasis zona kedalaman maupun oleh *pid_controller_node* sebagai umpan balik kendali. Alur kendali terbentuk berurutan dari *mission_manager_node* yang mempublikasikan */pid/setpoint* dan */cmd/velocity*, diproses *pid_controller_node* menjadi */pid/output*, dikonversi *kinematics_node* menjadi */thruster/pwm*, hingga diteruskan *serial_bridge_node* ke ESP32 dengan umpan balik status melalui */serial/feedback*. *Node mission_recorder_node* dan *auv_full_dashboard* *men-subscribe* seluruh *topic* aktif untuk pemantauan *real-time*. Kesesuaian graf hasil implementasi terhadap diagram perancangan menunjukkan bahwa integrasi seluruh *node* telah berjalan sebagaimana dirancang.

Tabel 4. 1 Daftar *Node* ROS2 yang Aktif pada Sistem AUV KRBAI

No	Nama Node	Fungsi
1	<i>/auv_full_dashboard</i>	Visualisasi status sistem secara <i>real-time</i>
2	<i>/depth_sensor_node</i>	Akuisisi data kedalaman dari sensor MS5837

No	Nama Node	Fungsi
3	<i>/kinematics_node</i>	Konversi keluaran PID menjadi nilai PWM 8 <i>thruster</i>
4	<i>/mission_manager_node</i>	Pengelola urutan misi, representasi lingkungan lokal berbasis zona kedalaman, dan algoritma APF
5	<i>/mission_recorder_node</i>	Perekaman seluruh <i>topic</i> kemudian disimpan pada <i>rosvbag</i>
6	<i>/pid_controller_node</i>	Kendali PID untuk DOF <i>depth</i> , <i>yaw</i> , <i>roll</i> , <i>pitch</i>
7	<i>/serial_bridge_node</i>	Komunikasi serial dua arah Jetson–ESP32
8	<i>/stereo_camera_node</i>	Akuisisi dan rektifikasi citra stereo DWE exploreHD
9	<i>/stereo_depth_node</i>	Komputasi <i>depth map</i> SGBM dan zona jarak rintangan
10	<i>/wimotion_imu_node</i>	Akuisisi data orientasi dari sensor HWT905
11	<i>/yolo_perception_node</i>	Deteksi objek YOLO TensorRT dan fusi jarak

(Sumber : Dokumentasi Pribadi)

4.1.4 Spesifikasi dan Kemampuan AUV

Autonomous Underwater Vehicle yang dirancang pada penelitian ini merupakan wahana bawah air kelas ringan yang dibangun mengacu pada regulasi Kontes Robot Bawah Air Indonesia. Rangka dan *enclosure* dibuat dari material *High Density Polyethylene* (HDPE) yang dikombinasikan dengan komponen hasil pencetakan tiga dimensi *bracket* kamera,udukan *thruster*, dan penopang sensor. Spesifikasi teknis beserta kemampuan operasional wahana dirangkum pada Tabel 4.2.

Tabel 4. 2 Spesifikasi dan Kemampuan Autonomous Underwater Vehicle

No	Parameter	Spesifikasi
1	Dimensi (P × L × T)	64,7 × 44,4 × 28,1 cm
2	Berat di udara	19 kg
3	Material rangka dan enclosure	HDPE dan komponen cetak 3D
4	Unit pemrosesan utama	Jetson Orin NX 16 GB (Ubuntu 22.04, ROS2 Humble)

No	Parameter	Spesifikasi
5	Unit kendali tingkat rendah	ESP32 (penerus PWM dan pemantau kill switch)
6	Sistem penggerak	8 × thruster T200 (4 horizontal, 4 vertikal)
7	Rentang sinyal kendali thruster	1100 – 1500 – 1900 μ s
8	Sensor kedalaman	MS5837-30BA (I ² C)
9	Sensor orientasi	Witmotion HWT905, mode 6-axis (UART)
10	Sistem persepsi visual	2 × kamera DWE exploreHD sebagai <i>stereo pair</i>
11	Baseline dan panjang fokus stereo	98,51 mm dan 350,99 piksel
12	Rentang kerja estimasi jarak	0,4 – 2,0 m
13	Sistem keamanan	Magnetic switch MC-38 sebagai kill switch
14	Catu daya sistem kendali	Li-ion VTC6 3S6P (18 sel); 10,8 V; 18 Ah
15	Catu daya sistem penggerak	Li-ion VTC6 4S8P (32 sel); 14,4 V; 24 Ah
16	Durasi operasi	± 3 jam dengan 11 node ROS2 aktif
17	Derajat kebebasan	6 DOF (surge, sway, heave, roll, pitch, yaw)
18	Kecepatan maju dan mundur	0,455 m/s dan 0,417 m/s
19	Kecepatan menyamping	0,369 m/s
20	Kecepatan naik dan turun	0,261 m/s dan 0,229 m/s
21	Kecepatan berputar (yaw)	60,0 °/s
22	Kemampuan navigasi	Obstacle avoidance otonom berbasis <i>stereo depth</i> dan APF

(Sumber : Dokumentasi Pribadi)

4.2 Pengujian Sensor dan Aktuator

Pada Tugas Akhir ini dilakukan proses pengujian terhadap sensor dan aktuator yang digunakan dalam sistem. Pengujian ini bertujuan untuk mengetahui tingkat akurasi dari setiap komponen yang digunakan dalam mendukung kinerja sistem. Selain itu, pengujian juga dilakukan untuk menganalisis besarnya kesalahan (*error*) yang dihasilkan oleh masing-masing komponen. Hasil dari pengujian sensor dan aktuator tersebut kemudian digunakan sebagai dasar untuk mengevaluasi kelayakan komponen dalam mendukung sistem navigasi dan *obstacle avoidance* AUV.

4.2.1 Pengujian Sensor MS5837 Pressure

Pengujian sensor tekanan MS5837 dilakukan untuk mengetahui kemampuan sensor dalam mengukur kedalaman air berdasarkan perubahan tekanan yang diterima sensor. Proses pengujian dilakukan dengan membandingkan nilai kedalaman aktual yang diukur menggunakan penggaris dengan nilai kedalaman yang dibaca oleh sensor.

Pengujian dilakukan pada sebuah bak berisi air dengan variasi kedalaman tertentu. Sensor MS5837 dimasukkan ke dalam air secara bertahap sesuai dengan nilai kedalaman yang telah ditentukan. Kedalaman aktual diukur menggunakan penggaris atau meteran sebagai acuan pengukuran.



Gambar 4. 4 Pengujian Sensor Pressure MS5837

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.4, terlihat bahwa proses pengujian dilakukan dengan menempatkan sensor MS5837 pada bak air, kemudian kedalaman sensor diukur menggunakan penggaris yang diletakkan secara vertikal. Penggaris tersebut digunakan sebagai referensi untuk mengetahui kedalaman aktual sensor di dalam air, sehingga hasil pembacaan sensor dapat dibandingkan dengan nilai kedalaman sebenarnya.

Tabel 4. 3 Hasil Pengujian Sensor Pressure MS5837

Pengujian	Jarak Pengujian (cm)	Pembacaan Sensor (cm)	Nilai Error (%)
1	0	0	0
2	3	2,95	1,67
3	6	6,08	1,33

Pengujian	Jarak Pengujian (cm)	Pembacaan Sensor (cm)	Nilai Error (%)
4	9	9,15	1,67
5	12	12,10	0,83
6	15	14,92	0,53
7	18	18,30	1,67
8	21	21,12	0,57
9	24	23,85	0,63
10	27	27,40	1,48
Rata-rata Error			1,04

(Sumber : Dokumentasi Pribadi)

Hasil pengujian yang ditunjukkan pada Tabel 4.3 menunjukkan bahwa nilai pembacaan sensor mendekati nilai kedalaman aktual. Contoh studi kasus, diambil di mana sensor ditempatkan pada kedalaman aktual $h = 10 \text{ cm}$ ($0,10\text{m}$) di dalam bak air tawar. Berdasarkan pembacaan data mentah sensor pada kondisi tersebut, tekanan total (P) yang diterima elemen sensor adalah sekitar 102315 Pa ($1023,15 \text{ mbar}$), dengan tekanan atmosfer referensi permukaan (P_0) bernilai 101325 Pa ($1013,25 \text{ mbar}$). Dengan menggunakan parameter massa jenis air tawar $\rho = 997 \text{ kg/m}^3$ dan percepatan gravitasi $g = 9,81 \text{ m/s}^2$, perhitungan kedalaman (h) dilakukan sebagai berikut:

$$h = \frac{P - P_0}{\rho \cdot g} \quad (4.1)$$

$$h = \frac{102315 \text{ Pa} - 101325 \text{ Pa}}{997 \text{ kg/m}^3 \times 9,81 \text{ m/s}^2} \quad (4.2)$$

$$h = \frac{990 \text{ Pa}}{9780,57 \text{ N/m}^3} \approx 0,1012 \text{ m} = 10,12 \text{ cm} \quad (4.3)$$

Rata-rata *error* yang diperoleh dari pengujian ini sebesar 1.04%, yang menunjukkan bahwa sensor MS5837 memiliki tingkat akurasi yang cukup baik dalam mengukur kedalaman air. Dengan demikian, sensor ini dapat digunakan untuk mendukung sistem yang dikembangkan pada penelitian ini.

4.2.2 Pengujian Sensor HWT905

Pengujian sensor IMU HWT905 dilakukan untuk mengetahui tingkat akurasi sensor dalam mengukur sudut orientasi berupa *pitch*, *roll*, dan *yaw* atau sumbu X, Y, dan Z. Pengujian dilakukan dengan membandingkan sudut yang dibaca oleh sensor dengan sudut referensi yang diukur menggunakan busur derajat. Metode ini

digunakan untuk mengetahui selisih pembacaan sensor terhadap nilai sudut sebenarnya sehingga dapat dihitung nilai *error* dari sensor.

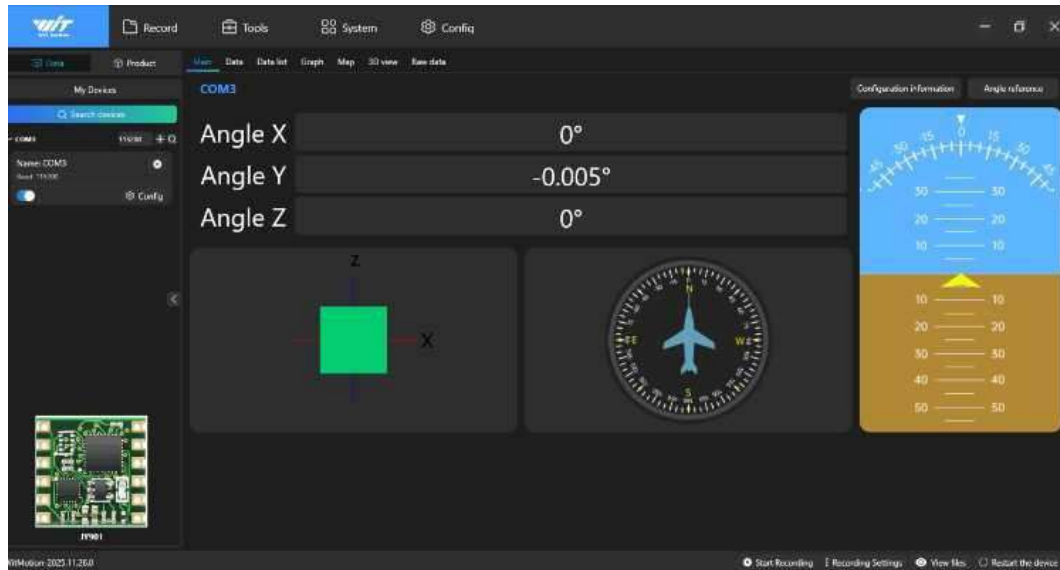
Pada proses pengujian Roll dan Pitch, sensor ditempatkan pada busur derajat 180° kemudian dimiringkan sesuai dengan sudut pengujian yang telah ditentukan. Sedangkan untuk pengujian Yaw, digunakan busur derajat 360° untuk mengetahui perubahan arah rotasi sensor terhadap sumbu vertikal. Pengambilan data dilakukan secara bertahap pada beberapa titik sudut referensi guna memastikan konsistensi linieritas pembacaan di seluruh rentang ukur. Pada setiap titik pengujian, posisi sensor ditahan selama beberapa saat untuk memberikan waktu bagi algoritma fusi internal dalam menstabilkan keluaran datanya. Hasil pengukuran dari berbagai variasi sudut tersebut kemudian ditabulasi untuk menghitung rata-rata persentase galat (*error*) secara keseluruhan. Tingkat presisi yang divalidasi melalui pengujian ini sangat krusial sebagai jaminan bahwa sistem kendali PID nantinya menerima umpan balik orientasi yang sangat akurat selama wahana bermanuver.



Gambar 4. 5 Pengujian Nilai Sensor HWT905 dengan Busur

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.5, terlihat bahwa sensor IMU HWT905 ditempatkan pada busur derajat sebagai acuan sudut pengujian. Sensor kemudian dimiringkan atau diputar secara bertahap sesuai dengan sudut yang diinginkan. Nilai sudut yang ditunjukkan oleh busur derajat digunakan sebagai sudut referensi, sedangkan nilai yang terbaca pada sistem merupakan hasil pembacaan sensor. Pengujian sensor Wot-Motion HWT-905 menggunakan software bawaan yaitu *witmotion.exe*.



Gambar 4. 6 Software WitMotion

(Sumber : Dokumentasi Pribadi)

Dengan memanfaatkan software tersebut, kita dapat mengamati sudut hasil pembacaan dari sensor. Komunikasi antara sensor dan laptop dilakukan melalui kabel USB to TTL dengan menggunakan protokol komunikasi UART. Dari hasil pengujian antara hasil pembacaan sensor dan busur didapatkan data dilihat pada Tabel 4.4, Tabel 4.5, dan Tabel 4.6.

Tabel 4. 4 Pengujian Sumbu X

Pengujian	Sudut Referensi (°)	Pembacaan Sensor (°)	Error (%)
1	0	0	0
2	20	20,31	1,55
3	40	39,78	0,55
4	60	60,42	0,70
5	80	80,36	0,45
6	100	100,21	0,21
7	120	119,92	0,07
8	140	140,28	0,20
9	160	159,74	0,16
10	180	179,48	0,29
Rata – rata Error Sumbu X			0,42

(Sumber : Dokumentasi Pribadi)

Berdasarkan hasil pengujian pada Tabel 4.4, diperoleh rata-rata *error* sebesar 0,42%. Hal ini menunjukkan bahwa sensor mampu mengukur kemiringan pada sumbu X dengan tingkat akurasi yang cukup baik. Pengujian pada sumbu X

dilakukan hanya sampai pada sudut 180° dikarenakan sensor WitMotion HWT905 akan memberikan nilai negatif 180° sampai 0° Ketika sensor melewati sudut 180° . Pengujian dilakukan dengan memiringkan sensor pada busur derajat 180° sehingga sudut referensi dapat dibandingkan langsung dengan hasil pembacaan sensor.

Tabel 4. 5 Pengujian Sumbu Y

Pengujian	Sudut Referensi ($^\circ$)	Pembacaan Sensor ($^\circ$)	Error (%)
1	0	0	0
2	10	10,41	1,40
3	20	20,19	0,95
4	30	29,82	0,60
5	40	40,08	0,20
6	50	49,95	0,10
7	60	60,11	0,18
8	70	69,87	0,18
9	80	79,91	0,11
10	90	89,63	0,41
Rata – rata Error (%) <i>Pitch</i>			0,41

(Sumber : Dokumentasi Pribadi)

Pada Tabel 4.5, diperoleh rata-rata *error* sebesar 0,41%. Nilai error yang relatif kecil menunjukkan bahwa sensor mampu mendeteksi perubahan sudut kemiringan pada sumbu Y dengan baik. Pengujian pada sumbu Y dengan hasil pembacaan sudut sampai pada sudut 90° dikarenakan sensor WitMotion HWT905 akan memberikan nilai negatif 90° sampai dengan 0° ketika sensor melewati sudut 90° . Proses pengujian dilakukan dengan metode yang sama seperti pengujian sumbu X, yaitu menggunakan busur derajat 180° sebagai acuan sudut.

Tabel 4. 6 Pengujian Sumbu Z

Pengujian	Sudut Referensi ($^\circ$)	Pembacaan Sensor ($^\circ$)	Error (%)
1	0	0	0
2	30	30,16	0,53
3	45	45,17	0,37
4	60	60,35	0,57
5	90	90,22	0,24
6	110	110,35	0,31
7	135	135,12	0,08
8	150	150,17	0,11
9	160	160,10	0,10
10	180	179,23	0,16

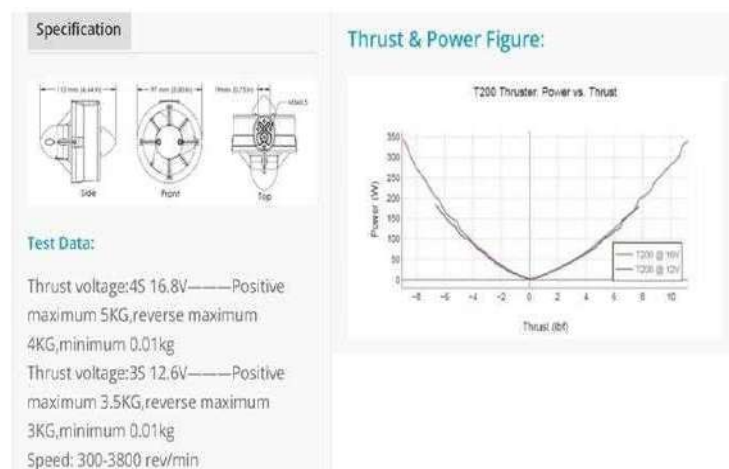
Pengujian	Sudut Referensi (°)	Pembacaan Sensor (°)	Error (%)
Rata – rata Error (%) <i>Yaw</i>			0,20

(Sumber : Dokumentasi Pribadi)

Selanjutnya, pada Tabel 4.6, diperoleh rata-rata *error* sebesar 0,20%, yang merupakan nilai *error* paling kecil dibandingkan pengujian pada sumbu lainnya. Pengujian pada sumbu Z dengan hasil pembacaan sudut sampai pada sudut 180° dikarenakan sensor WitMotion HWT905 akan memberikan nilai negatif 180° sampai dengan 0° ketika sensor melewati sudut 180°. Pengujian sumbu Z dilakukan menggunakan busur derajat 360° dengan cara memutar sensor mengikuti arah sudut pada sudut pada busur derajat sehingga perubahan orientasi sensor dapat diamati. Berdasarkan hasil pengujian tersebut, sensor IMU HWT905 menunjukkan performa yang cukup baik dalam mengukur sumbu X, Y, dan Z. Nilai rata-rata error yang berada di bawah 1% menunjukkan bahwa sensor memiliki tingkat akurasi yang tinggi sehingga dapat digunakan sebagai sensor orientasi pada sistem yang dikembangkan dalam penelitian ini.

4.2.3 Pengujian Motor *Thruster*

Pada pengujian motor *thruster* BLDC ROVMAKER T200 dilakukan dengan mengatur *input* PWM dengan suplai daya berupa tegangan sebesar 16V DC. Ketika motor mulai berputar, pengukuran jumlah RPM akan dilakukan menggunakan *tachometer* untuk menentukan seberapa baik motor thruster BLDC dapat beroperasi pada variasi *input* PWM.



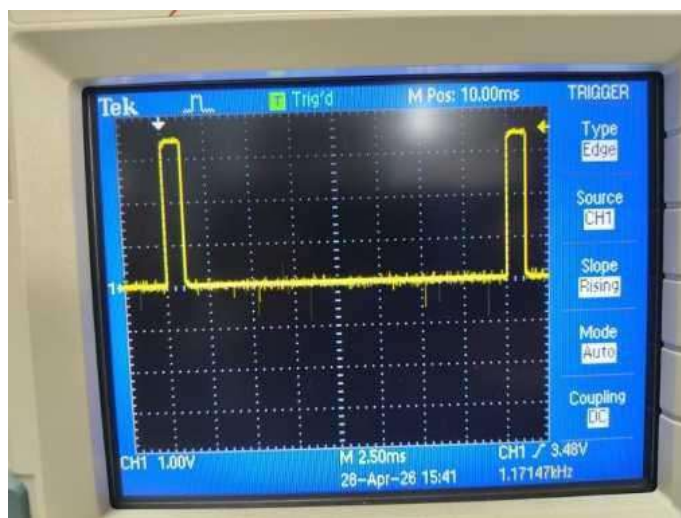
Gambar 4. 7 Spesifikasi Motor *Thruster* BLDC ROVMAKER T200

(Sumber : (Rovmaker, 2023b))

Sebelum melakukan pengujian pada motor *thruster* BLDC divalidasi terlebih dahulu bahwa komponen tersebut bekerja dengan baik sesuai spesifikasi teknis yang tertera pada Gambar 4.7. Pada nilai PWM maksimal yaitu 1900 μs dapat berputar sebesar 3800 RPM dengan suplai 16 VDC, sehingga dapat dihitung secara matematis menggunakan konsep perbandingan berikut.

Berdasarkan spesifikasi teknis pada Gambar 4.7, motor *thruster* T200 menghasilkan putaran maksimum sebesar 3800 RPM pada input PWM 1900 μs dengan suplai 16 VDC. Nilai ini digunakan sebagai acuan pembandingan terhadap hasil pengukuran RPM actual.

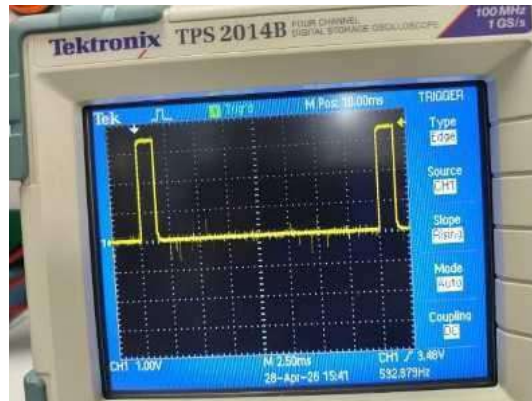
Hasil perhitungan diatas akan divalidasi dengan menggunakan osiloskop. Pengujian sinyal PWM dilakukan menggunakan osiloskop Tektronix TPS 2014B dengan pengaturan skala vertikal 1 V/div dan skala horizontal 2,5 ms/div. Pengujian dilakukan pada tiga titik representatif yang mewakili rentang operasional motor *thruster*, yaitu PWM 1100 μs (*reverse* maksimum), 1500 μs (netral), dan 1900 μs (*forward* maksimum).



Gambar 4. 8 Sinyal PWM Motor *Thruster* 1100 μs

(Sumber : Dokumentasi Pribadi)

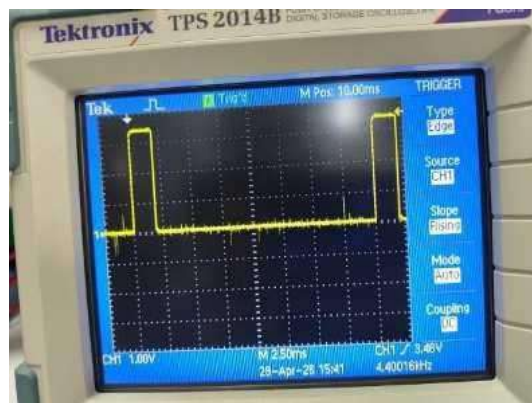
Pada Gambar 4.8 merupakan sinyal PWM dengan nilai 1100 μs yang dihasilkan oleh mikrokontroler ESP32. Dapat diketahui periode gelombang yang terjadi yaitu 20,0 ms dengan perbesaran horizontal (time/div) 2,5 ms dan perbesaran vertikal 1 volt/div. Lebar pulsa aktif (t_{on}) sebesar 1,1 ms sehingga *duty cycle* yang dihasilkan adalah 5,5%. Pada kondisi ini motor *thruster* berputar ke arah *reverse* dengan kecepatan maksimum.



Gambar 4. 9 Sinyal PWM Motor *Thruster* 1500 μ s

(Sumber : Dokumentasi Pribadi)

Pada Gambar 4.9 merupakan sinyal PWM dengan nilai 1500 μ s yang merupakan titik netral (*deadband*) ESC. Periode gelombang tetap 20,0 ms dengan lebar pulsa aktif (t_{on}) sebesar 1,5 ms sehingga *duty cycle* yang dihasilkan adalah 7,5%. Pada kondisi PWM 1500 μ s, motor *thruster* tidak berputar karena berada tepat di titik netral ESC.



Gambar 4. 10 Sinyal PWM Motor *Thruster* 1900 μ s

(Sumber : Dokumentasi Pribadi)

Pada Gambar 4.10 merupakan sinyal PWM dengan nilai 1900 μ s yang dihasilkan oleh mikrokontroler ESP32. Lebar pulsa aktif (t_{on}) sebesar 1,9 ms sehingga *duty cycle* yang dihasilkan adalah 9,5%. Pada kondisi ini motor *thruster* berputar ke arah *forward* dengan kecepatan maksimum.

Hasil dari analisis dan validasi perhitungan serta luaran gelombang pada ketiga titik pengujian menunjukkan bahwa frekuensi sinyal PWM konsisten sebesar 50 Hz dengan amplitudo 3,3 V sesuai level logika TTL ESP32. Perhitungan frekuensi dan *duty cycle* dapat dilihat pada persamaan berikut:

$$f = \frac{1}{T} = \frac{1}{20 \text{ ms}} = 50 \text{ Hz} \quad (4.4)$$

Dan perhitungan untuk mengetahui persentase *duty cycle* yang terjadi maka menggunakan rumus sebagai berikut:

$$D = \frac{t_{on}}{T} \times 100\% \quad (4.5)$$

Tabel 4. 7 Hasil Analisis Sinyal PWM Menggunakan Osiloskop

PWM Input (μ s)	Periode (ms)	Frekuensi (Hz)	t _{on} (ms)	Duty Cycle (%)	Kondisi Motor
1100	20	50	1,1	5,5	<i>Reverse</i> maksimum
1500	20	50	1,5	7,5	Netral
1900	20	50	1,9	9,5	<i>Forward</i> maksimum

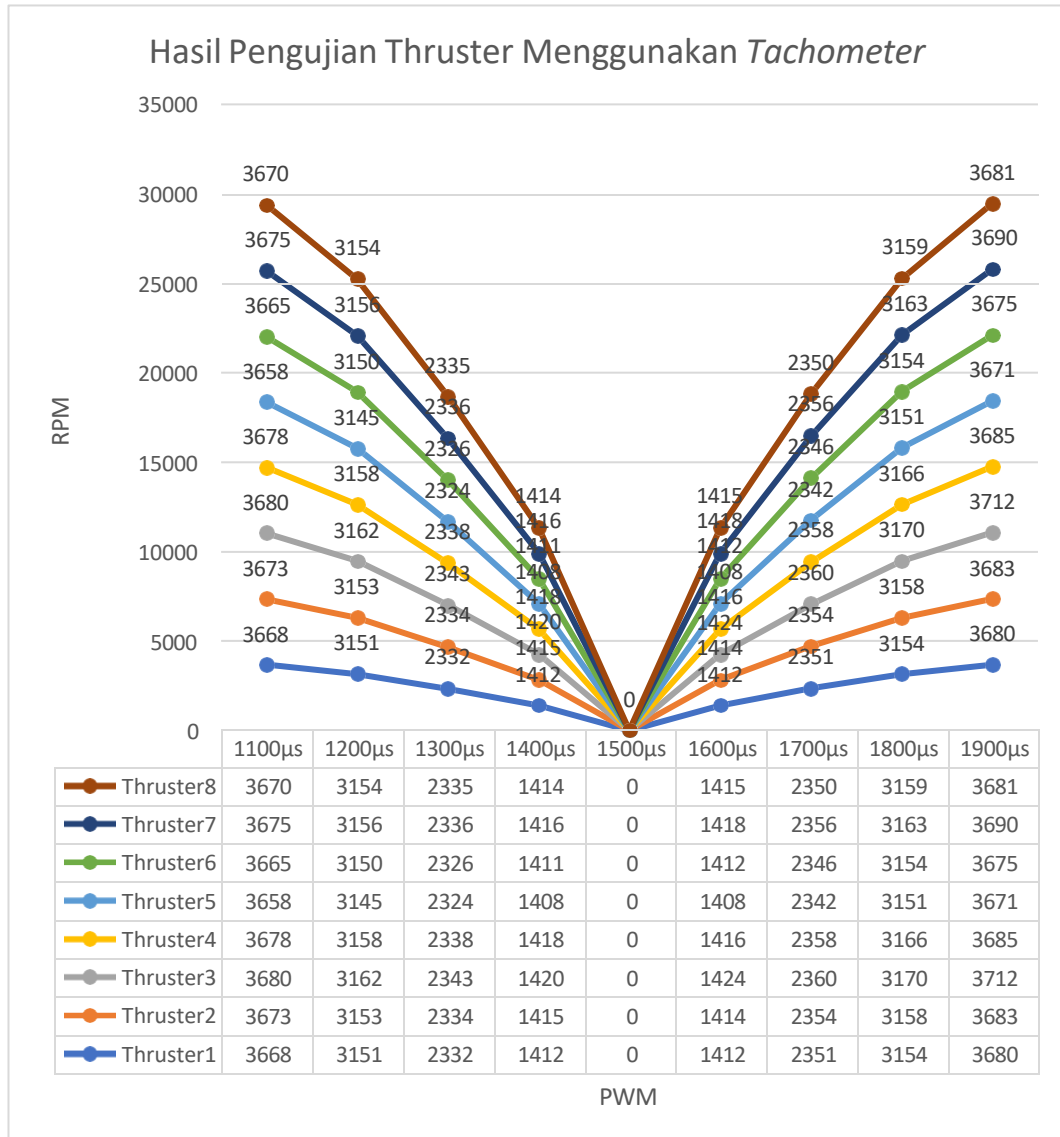
(Sumber : Dokumentasi Pribadi)

Keberhasilan validasi sinyal menjadi dasar untuk melakukan pengujian fisik kecepatan putaran (RPM) motor *thruster* BLDC menggunakan instrumen *tachometer*. Pengukuran ini bertujuan membandingkan RPM aktual dengan spesifikasi teknis guna menghitung persentase galat (*error*). Evaluasi dilakukan secara menyeluruh pada kedelapan motor *thruster* (didokumentasikan pada Gambar 4.11) untuk mengetahui deviasi respons antar motor ketika diberikan nilai PWM yang sama. Konsistensi respons ini sangat krusial karena kedelapan aktuator tersebut bertindak sebagai penggerak utama AUV yang dikendalikan secara terpusat oleh mikrokontroler ESP32. Rincian kuantitatif dari pengujian ini dirangkum dalam tabel hasil pengujian.



Gambar 4. 11 Pengujian Motor *Thruster* Menggunakan Kontroler ESP32

(Sumber : Dokumentasi Pribadi)



Gambar 4. 12 Grafik Hasil Pembacaan *Tachometer* pada *Thruster*

(Sumber : Dokumentasi Pribadi)

Dapat dilihat pada Gambar 4.12 merupakan hasil pengujian ke delapan motor *thruster* BLDC ROVMAKER T200 dengan variasi input PWM menggunakan mikrokontroler ESP32 yang akan digunakan pada sistem propulsi AUV. Hasil pengujian pada Gambar 4.12 didapatkan nilai RPM maksimal pada motor adalah 3712 RPM (Motor 3) dengan variabel input PWM sebesar 1900 µs. Apabila mengacu pada Gambar 4.7 spesifikasi motor *thruster* BLDC ROVMAKER T200 dengan variabel input PWM 1900 µs nilai RPM maksimal seharusnya yang didapat adalah 3800 RPM. Dengan perbandingan nilai dari spesifikasi dan uji coba maka didapatkan nilai rata-rata error sebesar 3,04%.

4.2.4 Pengujian Gerak Dasar dan Kecepatan AUV

Pengujian gerak dasar dilakukan untuk memverifikasi bahwa konfigurasi delapan *thruster* T200 mampu menghasilkan seluruh mode gerak yang dibutuhkan AUV selama misi, sekaligus mengukur kecepatan yang dicapai pada setiap arah. Data kecepatan ini menjadi acuan dalam menginterpretasikan hasil pengujian *obstacle avoidance*, karena algoritma Artificial Potential Field (APF) menghasilkan keluaran berupa arah dan besar kecepatan gerak. Mode gerak yang diuji meliputi maju, mundur, serong kiri, serong kanan, naik, turun, dan berputar (*yaw*).

Gerak mendatar diuji menggunakan nilai perintah kecepatan 300 pada kedalaman operasi ± 50 cm. Nilai tersebut dipilih karena sama dengan nilai yang digunakan pada pengujian lintasan, sehingga kecepatan yang terukur merepresentasikan kondisi operasi misi yang sebenarnya dan bukan kecepatan maksimum teoritis wahana. Gerak vertikal tidak dikendalikan melalui nilai perintah langsung, melainkan melalui keluaran pengendali PID kedalaman yang dibatasi (*clamping*) pada nilai maksimum 150 guna mencegah wahana melewati kedalaman acuan (*overshoot*) dan menghindari osilasi vertikal yang mengganggu kestabilan citra stereo. Pada saat pengujian, baterai sistem penggerak berada pada tegangan 16,3 V dan baterai sistem kendali pada tegangan 12 V, sehingga karakteristik PWM terhadap putaran *thruster* yang diperoleh pada subbab 4.2.3 tetap berlaku sebagai acuan.

Kecepatan gerak mendatar diperoleh dari waktu tempuh wahana melewati jarak acuan tetap pada lintasan uji berdasarkan rekaman video, sedangkan kecepatan gerak vertikal diperoleh dari selisih pembacaan sensor kedalaman MS5837 terhadap selang waktu perpindahannya dengan enam kali pengulangan. Kecepatan translasi dihitung menggunakan Persamaan 4.6 dan kecepatan sudut menggunakan Persamaan 4.7.

$$v = \frac{s}{t} \quad (4.6)$$

$$\omega = \frac{\Delta\theta}{t} \quad (4.7)$$

dengan v adalah kecepatan gerak translasi (m/s), s adalah jarak tempuh (m), t adalah waktu tempuh (s), ω adalah kecepatan sudut ($^{\circ}/s$), dan $\Delta\theta$ adalah perubahan sudut yang ditempuh ($^{\circ}$). Khusus untuk gerak naik dan turun, pengujian diulang sebanyak enam kali dan nilai yang dilaporkan merupakan rata-rata dari

keseluruhan pengulangan tersebut. Hasil pengujian kecepatan gerak AUV ditunjukkan pada Tabel 4.8.



Gambar 4. 13 Proses Pengujian Gerak Dasar AUV pada Kolam Renang

(Sumber : Dokumentasi Pribadi)

Tabel 4. 8 Hasil Pengujian Kecepatan Gerak AUV

No.	Mode Gerak	Nilai Kendali	Jarak / Sudut	Waktu	Kecepatan	Satuan
1	Maju	300	5,0 m	11	0,455	<i>m/s</i>
2	Mundur	300	5,0 m	12	0,417	<i>m/s</i>
3	Serong kiri	300	4,8 m	13	0,369	<i>m/s</i>
4	Serong kanan	300	4,8 m	13	0,369	<i>m/s</i>
5	Naik	PID maks. 150	110 cm	4,21	0,261	<i>m/s</i>
6	Turun	PID maks. 150	107 cm	4,67	0,229	<i>m/s</i>
7	Berputar	300	180°	3	60,0	°/s

(Sumber : Dokumentasi Pribadi)

Keterangan: pengujian gerak mendatar dilakukan pada kedalaman operasi $\pm$ 50 cm dengan tegangan baterai penggerak 16,3 V dan tegangan baterai sistem kendali 12 V. Nilai kecepatan naik dan turun merupakan rata-rata dari enam kali pengujian berdasarkan pembacaan sensor kedalaman, sedangkan kecepatan gerak mendatar dan berputar diperoleh dari waktu tempuh pada rekaman video.

Berdasarkan Tabel 4.7, kecepatan maju sebesar 0,455 m/s merupakan yang tertinggi di antara seluruh mode gerak mendatar, diikuti gerak mundur 0,417 m/s serta gerak serong kiri dan kanan yang masing-masing 0,369 m/s. Perbedaan tersebut dapat dijelaskan melalui luas penampang wahana yang menghadap arah gerak. Dengan dimensi 64,7 cm $\times$ 44,4 cm $\times$ 28,1 cm, luas penampang arah maju adalah 0,1248 m² sedangkan arah menyamping 0,1818 m², atau 1,457 kali lebih besar. Karena gaya hambat air sebanding dengan luas penampang dan kuadrat

kecepatan, pada gaya dorong yang sama kecepatan menyamping diperkirakan sebesar $0,455 \div \sqrt{1,457} = 0,377$ m/s, sedangkan kecepatan terukur adalah 0,369 m/s dengan selisih hanya 2,1 %. Kesesuaian ini menunjukkan bahwa perbedaan kecepatan tersebut merupakan konsekuensi geometri lambung wahana, bukan akibat ketidakseimbangan gaya dorong *thruster*. Kesamaan kecepatan serong kiri dan kanan juga menunjukkan konfigurasi *thruster* yang simetris pada kedua sisi, meskipun perlu dicatat bahwa pencatatan waktu dilakukan dengan resolusi satu detik sehingga kesetaraan tersebut berlaku dalam batas ketelitian pengukuran. Sementara itu, kecepatan mundur yang 8,4 % lebih rendah dibandingkan maju disebabkan geometri baling-baling T200 yang dirancang optimal untuk arah putaran maju, sehingga gaya dorong arah mundur lebih kecil meskipun luas penampang wahana pada kedua arah identik.

Pada gerak vertikal, kecepatan naik sebesar 0,261 m/s lebih besar 14,0 % dibandingkan kecepatan turun sebesar 0,229 m/s. Perbedaan ini disebabkan kondisi apung positif (*positively buoyant*) wahana, sehingga pada gerak naik gaya apung bekerja searah dengan arah gerak dan membantu dorongan *thruster*, sedangkan pada gerak turun gaya apung harus diatasi oleh *thruster*. Kondisi apung positif tersebut merupakan karakteristik yang dikehendaki karena berfungsi sebagai mekanisme keselamatan pasif, yaitu wahana akan mengapung ke permukaan dengan sendirinya apabila terjadi kehilangan daya. Kecepatan vertikal yang lebih rendah dibandingkan gerak mendatar juga sesuai dengan pembatasan keluaran pengendali PID kedalaman pada nilai 150, yaitu setengah dari nilai perintah yang digunakan pada gerak mendatar.

Gerak berputar menghasilkan kecepatan sudut sebesar 60,0 °/s, diperoleh dari rotasi 180° yang ditempuh dalam waktu 3 detik. Pengukuran ini dilakukan berdasarkan rekaman video dan tidak menggunakan pembacaan *yaw* sensor IMU HWT905, karena pada rotasi menerus pembacaan *yaw* mengalami lompatan nilai (*wrap-around*) saat melewati batas $\pm 180^\circ$. Hal tersebut merupakan karakteristik representasi sudut pada rentang -180° hingga $+180^\circ$ dan bukan indikasi kegagalan sensor, karena pada operasi misi sensor IMU digunakan untuk mempertahankan arah hadap (*heading hold*) di sekitar suatu nilai acuan, bukan untuk mengukur rotasi menerus.

Secara keseluruhan, hasil pengujian membuktikan bahwa AUV mampu melaksanakan ketujuh mode gerak dasar yang dibutuhkan untuk manuver penghindaran rintangan. Berdasarkan hasil pengujian pada subbab 4.7, perintah geser mulai dihasilkan ketika rintangan berada pada jarak 0,9 meter. Dengan kecepatan maju 0,455 m/s, wahana memiliki selang waktu sekitar 1,98 detik sejak perintah geser mulai aktif hingga mencapai posisi rintangan. Dalam selang waktu tersebut, wahana mampu bergeser menyamping sejauh 0,73 meter pada kecepatan 0,369 m/s, sedangkan pergeseran yang dibutuhkan agar badan wahana melewati suar adalah sekitar 0,29 meter. Dengan demikian tersedia margin sekitar 2,5 kali lipat, sehingga kemampuan manuver wahana memadai untuk melakukan penghindaran secara aman.

4.2.5 Pengujian Komunikasi Jetson Orin NX dan ESP32

Jalur komunikasi serial antara Jetson Orin NX dan ESP32 merupakan satusatunya penghubung antara hasil perhitungan algoritma pada Jetson dengan aktuator *thruster*. Pengujian dilakukan untuk memastikan jalur tersebut bekerja pada laju yang stabil, meneruskan nilai perintah tanpa perubahan, serta mampu melaporkan perubahan status *kill switch*. Laju data umpan balik diukur menggunakan perintah *ros2 topic hz* pada topik */serial/feedback* dan hasilnya ditunjukkan pada Gambar 4.14 serta Tabel 4.8.

```

auv@auv: ~
auv@auv: ~
auv@auv: ~
auv@auv: $ ros2 topic hz /serial/feedback
average rate: 50.480
  min: 0.010s max: 0.021s std dev: 0.00313s window: 52
average rate: 50.150
  min: 0.010s max: 0.024s std dev: 0.00302s window: 102

Executing task in folder auv esp32: platformio device monitor --environment esp32dev --
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:OFF | Pin:HIGH | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
Kill switch ON - magnet terpasang
[STATUS] Kill:ON | Pin:LOW | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:ON | Pin:LOW | PWM: 1500,1500,1500,1500,1500,1500,1500,1500
[STATUS] Kill:ON | Pin:LOW | PWM: 1500,1500,1500,1500,1500,1500,1500,1500

```

Gambar 4. 14 Pengukuran Laju Data Umpan Balik pada Topik */serial/feedback*

(Sumber : Dokumentasi Pribadi)

No.	Paramater	Pengukuran 1	Pengukuran 2
1	Laju data rata-rata	50, 480 Hz	50, 150 Hz
2	Interval minimum – maksimum	10 – 21 ms	10 – 24 ms
3	Simpangan baku interval	3,13 ms	3,02 ms
4	Jumlah sampel	52	102

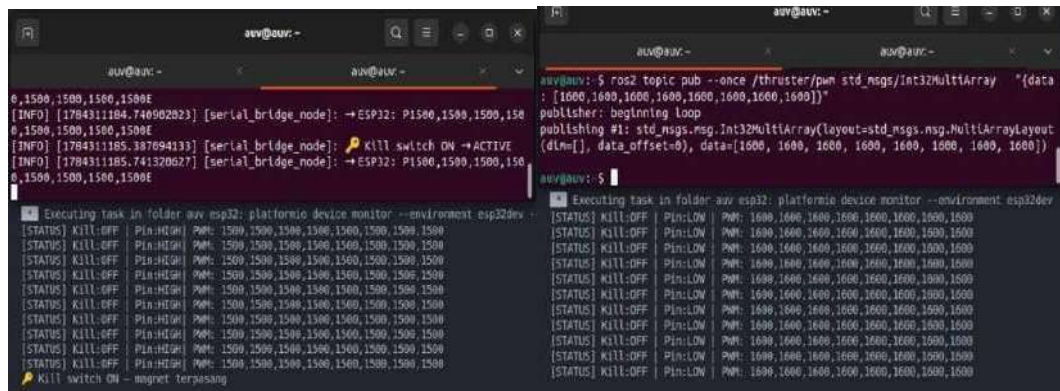
Berdasarkan Tabel 4.9, laju data umpan balik terukur sebesar 50,480 Hz dan 50,150 Hz, sesuai dengan laju yang dirancang sebesar 50 Hz. Nilai tersebut sama dengan frekuensi sinyal PWM yang telah divalidasi menggunakan osiloskop pada subbab 4.2.3, sehingga setiap bingkai PWM yang dikirim ke ESC bersesuaian dengan satu kali pembaruan perintah dari Jetson. Interval antar data berada pada rentang 10 ms hingga 24 ms dengan simpangan baku sekitar 3 ms. Interval maksimum 24 ms merupakan kondisi keterlambatan terburuk, yaitu hanya 0,61 persen dari selang waktu reaksi wahana terhadap rintangan sebesar 3,96 detik pada jarak pengaruh $D_0 = 1,8$ m, sehingga keterlambatan komunikasi tidak berpengaruh terhadap kemampuan manuver penghindaran secara *real-time*.

[illegible]

(Sumber : Dokumentasi Pribadi)

119

membandingkannya dengan nilai yang diterima ESP32 pada serial monitor. Pengujian ini hanya mengamati penerimaan data pada sisi ESP32 dan tidak mengamati pergerakan *thruster*, karena tujuannya adalah memastikan perintah dari ROS2 dapat diterima ESP32 secara langsung dan utuh. Berdasarkan Gambar 4.15, nilai tersebut diterima identik pada kedelapan kanal thruster. Hasil ini memverifikasi bahwa protokol serial P{m1},...,{m8}E dibentuk dan diuraikan dengan benar, urutan pemetaan kedelapan kanal sesuai antara pengirim dan penerima, serta nilai perintah diteruskan tanpa penskalaan maupun pembulatan. Pengujian dilakukan tanpa melalui *mission_manager_node* agar jalur penerusan data teruji secara terisolasi, sesuai perancangan dimana ESP32 hanya meneruskan nilai PWM ke ESC sedangkan seluruh logika pengamanan dijalankan pada sisi Jetson.



Gambar 4. 16 Pengujian Pelaporan Status *Kill Switch*: (kiri) magnet terpasang, (kanan) magnet terlepas

(Sumber : Dokumentasi Pribadi)

Pengujian *kill switch* dilakukan dengan memasang dan melepas magnet secara bergantian. Berdasarkan Gambar 4.16(kiri), pemasangan magnet mengubah pin masukan ESP32 menjadi *LOW* dan Jetson mencatat pesan “*Kill switch ON → ACTIVE*” yang menandakan sistem siap menerima perintah penggerak. Sebaliknya pada Gambar 4.16(kanan), pelepasan magnet mengembalikan pin ke kondisi *HIGH* dan Jetson mencatat pesan “*Kill switch OFF → STOP*”, dengan nilai PWM kedelapan *thruster* dipertahankan pada nilai netral 1500 μ s sehingga tidak menghasilkan gaya dorong. Kedua arah perubahan status berhasil dideteksi dan dilaporkan, dan mekanisme inilah yang digunakan operator untuk menghentikan tiga *trial* yang gagal pada pengujian dengan APF nonaktif.

4.3 Pengujian Kamera Stereo Menggunakan 2 Kamera DWE exploreHD

Pengujian kamera stereo bertujuan memastikan dua kamera DWE exploreHD dapat berfungsi sebagai satu sistem penglihatan stereo yang menghasilkan informasi kedalaman. Karena menggunakan dua kamera terpisah, seluruh proses pembentukan kedalaman dilakukan secara mandiri. Pengujian dibagi menjadi tiga tahap berurutan: pengaksesan kedua kamera pada Jetson Orin NX (4.3.1), kalibrasi untuk memperoleh parameter geometris kamera (4.3.2), serta pengujian peta kedalaman hasil algoritma SGBM (4.3.3).

4.3.1 Pengetesan Kamera DWE exploreHD

Kamera DWE exploreHD merupakan kamera USB yang bersifat *UVC-compliant* (USB Video Class), sehingga dapat langsung dikenali oleh sistem operasi tanpa memerlukan pemasangan driver khusus. Pada penelitian ini digunakan dua unit kamera DWE exploreHD yang dipasang berdampingan membentuk konfigurasi stereo, dan keduanya terhubung ke Jetson Orin NX melalui antarmuka USB.

Setiap kamera dikenali oleh sistem sebagai perangkat video terpisah pada direktori perangkat, yaitu `/dev/video0` untuk kamera kiri dan `/dev/video4` untuk kamera kanan. Identifikasi perangkat diperiksa menggunakan perintah `v4l2-ctl --list-devices` untuk memastikan kedua kamera terdeteksi dan tidak tertukar. Karena penomoran perangkat dapat berubah saat kamera dilepas-pasang, penetapan kamera kiri dan kanan dilakukan secara eksplisit di dalam program agar konsisten.

Pengaksesan citra dilakukan melalui pustaka OpenCV dengan backend V4L2. Setiap kamera dibuka pada resolusi 640×480 *piksel* dengan format kompresi MJPG (Motion JPEG) untuk menjaga kestabilan laju frame pada koneksi USB. Pengujian pada tahap ini bertujuan memastikan bahwa: (1) kedua kamera dapat diakses secara bersamaan, (2) citra dari kamera kiri dan kanan terbaca dengan benar sesuai posisinya, dan (3) laju frame mencukupi untuk pemrosesan stereo secara *real-time*.

Hasil pengujian menunjukkan kedua kamera berhasil diakses secara simultan dan menghasilkan pasangan citra kiri–kanan yang siap diproses pada tahap kalibrasi dan pembentukan peta kedalaman. Keberhasilan sinkronisasi pengambilan gambar dari kedua perangkat ini menjadi fondasi penting untuk memastikan tidak

adanya galat (*error*) pada perhitungan nilai disparitas akibat selisih waktu penangkapan citra saat AUV bergerak.



Gambar 4. 17 Tampilan Citra Kiri dan Kanan

(Sumber : Dokumentasi Pribadi)

4.3.2 Proses Kalibrasi Kamera DWE exploreHD

Kalibrasi bertujuan memperoleh parameter geometris kedua kamera yang diperlukan untuk proses rektifikasi dan pembentukan peta kedalaman. Proses kalibrasi menggunakan pola papan catur (*checkerboard*) dengan ukuran kotak 25 mm sebagai objek acuan berpola teratur.

Data kalibrasi diambil dengan menampilkan papan catur pada berbagai posisi, jarak, dan sudut kemiringan di hadapan kedua kamera secara serentak, sehingga diperoleh sejumlah pasangan citra kiri-kanan. Variasi posisi ini penting agar proses optimasi memperoleh cukup informasi untuk memodelkan parameter kamera secara akurat.



Gambar 4. 18 Proses Pengambilan data Papan Catur

(Sumber : Dokumentasi Pribadi)

Dari pasangan citra tersebut, program kalibrasi menghitung parameter intrinsik masing-masing kamera (panjang fokus, titik pusat, dan koefisien distorsi) serta parameter ekstrinsik berupa rotasi dan translasi antar kamera. Karena kamera DWE exploreHD menggunakan lensa fisheye dengan sudut pandang lebar,

kalibrasi dilakukan menggunakan model distorsi fisheye. Kualitas kalibrasi diukur melalui nilai galat reproyeksi (RMS reprojection error); kalibrasi dianggap baik apabila nilai RMS di bawah 1 *piksel*. Hasil kalibrasi yang digunakan pada sistem dirangkum pada Tabel 4.10.

Tabel 4. 10 Hasil kalibrasi kamera stereo DWE exploreHD

Parameter	Nilai
Model distorsi	Fisheye (Kannala-Brandt)
Ukuran kotak papan catur	25 mm
RMS reprojection error kiri	0.518 <i>piksel</i>
RMS reprojection error kanan	0.497 <i>piksel</i>
RMS stereo	0.6799 <i>piksel</i>
Panjang fokus (rektifikasi)	350,99 <i>piksel</i>
<i>Baseline</i>	98,51 mm

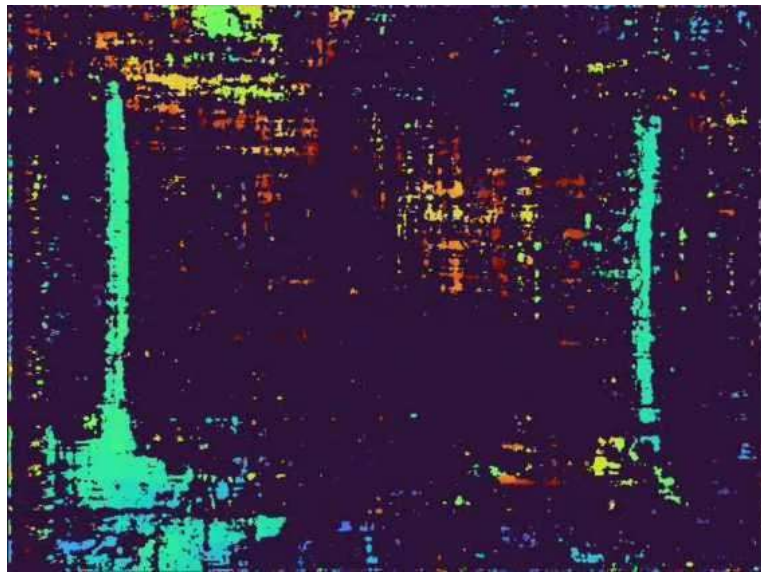
(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.9, nilai RMS reprojection error kamera kiri dan kanan masing-masing sebesar 0,518 *piksel* dan 0,497 *piksel*, sedangkan RMS stereo sebesar 0,680 *piksel*. Ketiga nilai tersebut berada di bawah 1 *piksel*, sehingga kalibrasi dinyatakan baik. Nilai RMS kamera kiri dan kanan yang hampir setara menunjukkan kedua kamera terkalibrasi dengan kualitas yang seimbang, sedangkan nilai RMS stereo yang sedikit lebih besar merupakan hal yang wajar karena kalibrasi stereo turut memodelkan posisi relatif kedua kamera.

Selain nilai RMS, kesahihan hasil kalibrasi juga ditunjukkan oleh nilai *baseline* sebesar 98,51 mm yang konsisten dengan jarak fisik antar kedua kamera hasil pengukuran langsung. Kesesuaian ini memastikan bahwa geometri hasil kalibrasi merepresentasikan susunan kamera yang sebenarnya. Parameter inilah panjang fokus hasil rektifikasi sebesar 350,99 *piksel* dan *baseline* 98,51 mm yang selanjutnya digunakan untuk membentuk peta rektifikasi serta menjadi dasar perhitungan jarak pada algoritma SGBM. Proses rektifikasi yang memanfaatkan parameter ini berfungsi untuk menyelaraskan kedua citra secara matematis sehingga memiliki proyeksi *epipolar* yang sejajar secara horizontal. Penyelarasan garis *epipolar* ini sangat krusial karena menyederhanakan pencarian titik padanan (*pixel correspondence*) antara citra kiri dan kanan menjadi pencarian satu dimensi saja. Reduksi ruang pencarian dari dua dimensi menjadi satu dimensi tersebut secara signifikan mampu menurunkan beban komputasi dari algoritma *Semi-Global Block Matching* (SGBM). Efisiensi komputasi ini sangat dibutuhkan oleh perangkat

Jetson Orin NX agar sistem mampu memproduksi peta kedalaman yang padat (*dense depth map*) secara *real-time*. Selain itu, tingkat akurasi dari matriks intrinsik dan ekstrinsik hasil kalibrasi ini akan secara langsung mendikte rentang resolusi kedalaman (*depth resolution*) yang dapat diukur secara andal oleh wahana. Oleh karena itu, validitas kalibrasi stereo ini menjadi fondasi geometris yang solid sebelum sistem mengompensasi efek pembelokan cahaya pada tahapan kalibrasi bawah air selanjutnya.

4.3.3 Hasil Depth Map SGBM



Gambar 4. 19 Tampilan *Stereo Depth* pada Bawah Air menghadap Gawang 1

(Sumber : Dokumentasi Pribadi)

Gambar 4.19 menampilkan hasil peta kedalaman (*depth map*) yang dihasilkan algoritma SGBM ketika kamera menghadap gawang. Perbedaan warna pada peta merepresentasikan perbedaan jarak objek terhadap kamera, dengan warna kehijauan menunjukkan objek yang lebih dekat dan warna gelap (ungu) menunjukkan area yang tidak memiliki nilai kedalaman.

Pada peta kedalaman tersebut, kedua tiang gawang terpetakan dengan jelas sebagai dua garis vertikal yang menonjol terhadap latar belakang. Kedua tiang terbaca secara utuh dari atas hingga bawah dengan nilai kedalaman yang konsisten, yang menunjukkan bahwa algoritma SGBM mampu memetakan objek bertekstur dan berstruktur jelas dengan baik. Hal ini membuktikan bahwa sistem stereo mampu memperoleh informasi kedalaman objek nyata.

Di sisi lain, peta kedalaman juga memperlihatkan sejumlah derau (*noise*) berupa bercak-bercak warna yang tersebar, terutama pada bagian tengah (di antara kedua tiang) dan pada area latar belakang. Derau ini merupakan karakteristik yang melekat pada metode stereo pasif dan dipengaruhi oleh beberapa faktor sebagai berikut:

- Tekstur permukaan yang rendah. Algoritma SGBM bekerja dengan mencocokkan pola *piksel* antara citra kiri dan kanan. Pada area yang seragam dan tidak memiliki pola khas seperti bagian tengah bukaan gawang yang berupa air terbuka tanpa permukaan padat proses pencocokan menjadi ambigu karena banyak titik memiliki kemiripan, sehingga nilai *disparitas* yang dihasilkan tidak stabil dan muncul sebagai bercak. Faktor ini merupakan penyebab utama derau pada sistem ini.
- Hamburan cahaya dan partikel dalam air. Hamburan balik cahaya (*backward scattering*) oleh partikel yang melayang di dalam air menimbulkan pola visual semu yang berbeda antara citra kiri dan kanan, sehingga menghasilkan pencocokan yang keliru pada area yang sebenarnya kosong.
- Pencahayaan bawah air yang tidak merata. Distribusi cahaya di dalam air yang tidak seragam menyebabkan perbedaan intensitas antara citra kiri dan kanan pada titik yang sama, sehingga menurunkan ketepatan pencocokan *piksel*.
- Keterbatasan jangkauan pengukuran. Objek yang berada terlalu jauh menghasilkan disparitas yang kecil sehingga rentan terhadap kesalahan, sedangkan area di luar rentang pengukuran tidak menghasilkan nilai kedalaman yang valid.

Meskipun peta kedalaman mengandung derau, hal tersebut tidak mengganggu fungsi utama sistem. Pembacaan jarak untuk pengambilan keputusan tidak diambil dari setiap *piksel*, melainkan diolah pada tingkat zona menggunakan nilai persentil dengan syarat jumlah *piksel* valid minimum, sehingga bercak-bercak derau yang tersebar akan tersaring dan tidak menghasilkan pembacaan jarak yang keliru, sebagaimana telah diuraikan pada BAB 3. Selain itu, karakteristik gawang yang berongga pada bagian tengahnya menjadi dasar pertimbangan bahwa estimasi

jarak gawang tidak diandalkan untuk penghindaran, melainkan navigasi gawang menggunakan posisi horizontal objek hasil deteksi.

4.4 Hasil Pembahasan YOLO

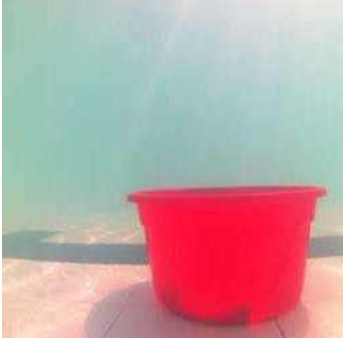
4.4.1 Pengumpulan Dataset Pelatihan YOLO

Dataset yang digunakan dalam penelitian ini terdiri dari Kumpulan gambar bola yang diklasifikasikan berdasarkan karakteristik visual, yaitu warna dan bentuk. Gambar-gambar ini dikelompokkan dalam kelas deteksi yaitu gawang, suar, dan bak untuk mendukung pelatihan model dalam system visi komputer pada AUV. Seperti yang terlihat pada Tabel 4.10, klasifikasi ini berfungsi untuk membedakan objek gawang, suar, dan bak dari latar belakang atau objek lain yang mungkin ada di lapangan. Gambar – gambar dikumpulkan dari berbagai sumber dan melalui proses seleksi serta pengolahan untuk memastikan kualitas dan konsistensinya. Setiap gambar diberi label sesuai kategorinya untuk mempermudah pelatihan model YOLO dalam mengenali objek gawang, suar, dan bak secara spesifik. Pengelompokan berdasarkan warna dominan dan bentuk bertujuan untuk meningkatkan akurasi deteksi, karena algoritma dapat mempelajari ciri-ciri visual khas dari objek bola yang menjadi fokus AUV. Selain itu, dataset ini telah melalui tahapan pengolahan data seperti pelabelan manual, augmentasi data seperti pencahayaan dan kontras, yang bertujuan untuk meningkatkan variasi kondisi citra dan memperkuat ketahanan model terhadap perubahan cahaya dan kondisi air di kolam renang.

Dengan dataset yang terstruktur dan relevan dengan konteks lomba autonomous underwater vehicle, diharapkan mampu mendeteksi objek, khususnya mendeteksi halang rintang (gawang, suar, dan bak), dapat berjalan optimal, baik dalam hal kecepatan maupun akurasi. Ini sangat penting agar AUV dapat merepons halang rintang secara real-time dalam situasi kompetitif di kolam.

Pengujian tambahan juga dilakukan pada jarak tertentu pada kolam renang, mulai dari jarak dekat hingga 2.0 m. Hasil dari pengujian ini kemudian dianalisis untuk menilai keandalan sistem deteksi dalam berbagai jarak. Model juga divalidasi menggunakan data uji terpisah untuk menghindari overfitting dan memastikan generalisasi yang baik terhadap data baru.

Tabel 4. 11 Dataset Gawang, Suar, dan Bola

Class	Dataset	Jumlah
Gawang		2000
Suar		2000
Bak		2000
Jumlah Dataset		6000

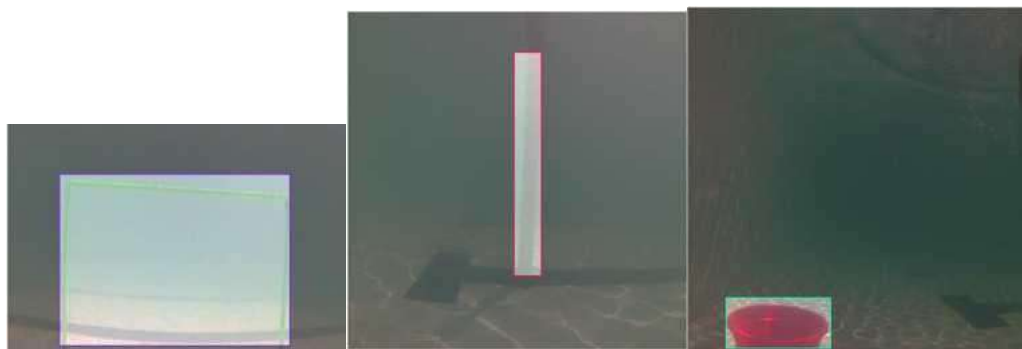
(Sumber : Dokumentasi Pribadi)

4.4.2 Pra-pemrosesan Data

Pada tahap ini, dilakukan analisis terhadap hasil deteksi objek menggunakan model YOLO yang telah dilatih dengan 6.000 gambar. Gambar-gambar ini berisi tiga objek utama, yaitu gawang, suar, dan bak yang menjadi fokus dalam sistem penglihatan AUV. Setiap gambar diberi tanda kotak di sekitar objek untuk menunjukkan lokasi gawang, suar, dan bak serta dilabeli untuk membedakan ketiganya. Proses pelabelan dilakukan secara manual dengan bantuan perangkat lunak khusus seperti roboflow, memungkinkan system untuk belajar dari data yang telah diberi tanda dengan benar.

Tanda kotak pada gawang akan mempermudah AUV untuk masuk ketengah gawang untuk melewati gawang, sedangkan kotak pada suar akan mempermudah

AUV untuk menghindari suar, dan yang terakhir tanda kotak pada bak akan mempermudah AUV tahu apakah posisi bak sudah tepat dibawah AUV. Dengan ketiga objek ini, sistem dapat membedakan posisi gawang, suar, dan bak dalam satu gambar. Kemampuan ini sangat penting agar AUV dapat bergerak melewati halang rintang hingga titik finish yang ditentukan. Hasil pengujian menunjukkan bahwa system dapat mengenali gawang, suar, dan bak dengan baik, selama objek terlihat jelas dan tidak saling menutupi. Pelabelan yang teratur dan konsisten dalam menyiapkan data terbukti meningkatkan akurasi sistem dalam mendeteksi dan melacak objek selama pertandingan.



Gambar 4. 20 Anotasi Pelabelan Objek Gawang, Suar, dan Bak

(Sumber : Dokumentasi Pribadi)

Sebelum melatih sistem deteksi objek, dataset gambar dibagi menjadi tiga bagian, yaitu data pelatihan (*train*), data validasi (*valid*), dan data pengujian (*test*). Dari total 6.000 gambar yang tersedia, sekitar 70% atau 4.200 gambar digunakan untuk pelatihan. Data pelatihan ini digunakan untuk mengajarkan sistem dalam mengenali pola objek gawang, suar, dan bak, sehingga sistem dapat mengidentifikasi dan membedakan kedua objek tersebut dalam gambar dengan akurat. Kemudian, 20% atau sekitar 1.200 gambar digunakan sebagai data validasi, yang bertujuan untuk memeriksa apakah sistem sudah belajar dengan baik selama proses pelatihan dan untuk mencegah terjadinya kesalahan belajar yang berulang. Data validasi juga digunakan untuk menyempurnakan hyperparameter model agar sistem dapat berfungsi secara optimal. Sisanya, 10% atau 600 gambar digunakan sebagai data uji, yang tidak pernah diakses oleh sistem sebelumnya, untuk mengukur seberapa baik sistem bekerja dengan gambar baru di luar proses pelatihan, dan untuk memastikan generalisasi model pada data yang belum pernah dilihat sebelumnya. Pembagian data ini dirancang untuk memastikan bahwa sistem

deteksi tidak hanya mengenali objek yang telah dilatih, tetapi juga dapat beradaptasi dengan kondisi gambar yang berbeda-beda saat diterapkan di kolam atau lapangan. Dengan pembagian data ini, diharapkan system deteksi dapat belajar secara optimal, mengenali objek yang telah dikenal, dan tetap bekerja akurat dalam kondisi gambar yang bervariasi di kolam. Pendekatan ini juga memastikan bahwa hasil yang diperoleh mencerminkan kemampuan sistem secara keseluruhan.

Tabel 4. 12 Pembagian Jumlah Dataset

Class	Train (70%)	Valid (20%)	Test (10%)
Gawang	2.100	600	300
Suar	2.100	600	300
Bak	2.100	600	300

(Sumber : Dokumentasi Pribadi)

- *Train* (70%): Data pelatihan ini digunakan untuk melatih model YOLO. Model belajar dari gambar yang sudah dilabeli untuk mengenali objek dan menentukan posisi serta kelas objek dalam gambar. Dataset pelatihan digunakan untuk mengoptimalkan bobot model selama proses pelatihan menggunakan teknik seperti backpropagation.
- *Valid* (20%): Data validasi digunakan untuk mengukur kinerja model selama pelatihan. Tujuannya adalah untuk memonitor kinerja model terhadap data yang tidak terlihat (belum digunakan dalam pelatihan) untuk menghindari overfitting. Data validasi membantu dalam memilih model terbaik berdasarkan metrik seperti mAP (*mean Average Precision*) tanpa mempengaruhi pelatihan langsung.
- *Test* (10%): Setelah pelatihan dan validasi selesai, data uji digunakan untuk mengevaluasi kemampuan model dalam mendeteksi objek pada gambar yang sepenuhnya baru, yang tidak digunakan dalam pelatihan atau validasi. Ini memberikan gambaran nyata tentang bagaimana model YOLO akan bekerja.

4.4.3 Pelatihan Model YOLO

Proses pelatihan data dilakukan dengan memanfaatkan platform Google Colab. Parameter-parameter yang digunakan dalam pelatihan ini dapat dilihat pada Tabel 4.12.

Tabel 4. 13 *Hyperparameter* Pelatihan Model YOLO

Parameter	Value
Image Size	640
Batch Size	16

Paramter	Value
Epoch	100

(Sumber : Dokumentasi Pribadi)

Dengan konfigurasi paramter seperti pada Tabel 4.15, proses pelatihan mencapai performa terbaik setelah melewati lebih dari 100 iterasi, sebagaimana ditunjukkan oleh tren penurunan nilai train/cls_loss dan val/cls_loss. Evaluasi performa model dilakukan menggunakan beberapa metrik utama berikut:

- *Presisi (Precision)*: Metrik ini menunjukkan seberapa akurat prediksi positif yang dilakukan oleh model, yaitu seberapa banyak dari hasil prediksi positif yang benar-benar positif.
- *Recall (Sensitivitas)*: Metrik ini mengukur kemampuan model dalam mendeteksi seluruh kasus positif yang sebenarnya, atau dengan kata lain, seberapa baik model menemukan seluruh data positif.
- *F1 Score*: Merupakan rata-rata harmonik dari presisi dan recall. Metrik ini penting ketika terdapat ketidakseimbangan antar kelas, karena memperhitungkan baik ketepatan maupun kelengkapan model.
- *Akurasi (Accuracy)*: Mengukur proporsi prediksi yang benar dari keseluruhan data yang diprediksi. Metrik ini memberikan gambaran umum tentang performa model.



Gambar 4. 21 Grafik Confusion Matrix

(Sumber : Dokumentasi Pribadi)

Berdasarkan hasil *confusion matrix* dan proses *training* model YOLOv8n, diperoleh nilai parameter evaluasi yang menunjukkan performa deteksi objek berjalan dengan baik.

Confusion matrix merupakan metode evaluasi yang digunakan untuk menilai kinerja model klasifikasi dengan membandingkan hasil prediksi terhadap data aktual. Melalui *confusion matrix*, dapat diketahui Tingkat ketepatan model dalam mengklasifikasikan objek. Perhitungan kinerja didasarkan pada empat komponen utama, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Keempat komponen tersebut menjadi dasar dalam mengevaluasi performa model. Visualisasi confusion mtrix pada penelitian ini ditampilkan pada Gambar 4.21.

Tabel 4. 14 Pembacaan Nilai Berdasarkan Confussion Matrix

Predicted / True	Gawang	Suar	Bak	Background
Gawang	0,98	0	0	0,15
Suar	0	0,99	0	0,61
Bak	0	0	0,98	0,25
Background	0,01	0,01	0,02	0

(Sumber : Dokumentasi Pribadi)

Pada *confusion matrix* yang ditunjukkan pada Tabel 4.13, terdapat tiga kategori utama, yaitu gawang, suar, bak, dan background.

- TP (*True Positive*) : objek berhasil terdeteksi dikenali dengan benar
- TN (*True Negative*) : data bukan objek berhasil dikenali dengan benar
- FP (*False Positive*) : kesalahan deteksi objek
- FN (*Flase Negative*) : objek gagal terdeteksi

Berdasarkan confusion matrix tersebut dilakukan tahapan perhitungan beberapa *metrix* evaluasi model sebagai berikut :

- Perhitungan accuracy gawang dilakukan menggunakan persamaan :

$$TP = 0,98$$

$$FP = 0,15$$

$$FN = 0,01$$

TN = semua sel yang tidak dibaris mapun kolam gawang. Ini menunjukkan sisa 9 sel: kolom suar ($0,99+0+0,01 = 1,0$) + kolom bak ($0+0,98+0,02 = 1,0$) + kolom background ($0,61+0,25+0 = 0,86$) jadi $1,0 + 1,0 + 0,86 = 2,86$

$$Accuracy_{gawang} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (4.8)$$

$$Accuracy_{gawang} = \frac{0,98 + 2,86}{0,98 + 2,86 + 0,15 + 0,01} \times 100\% \quad (4.9)$$

$$Accuracy_{gawang} = \frac{3,84}{4} \times 100\% = 96\% \quad (4.10)$$

- Perhitungan precision untuk kelas gawang :

$$Precision_{gawang} = \frac{TP}{TP + FP} \times 100\% \quad (4.11)$$

$$Precision_{gawang} = \frac{0,98}{0,98 + 0,15} \times 100\% \quad (4.12)$$

$$Precision_{gawang} = \frac{0,98}{1,13} \times 100\% = 86,73\% \quad (4.13)$$

- Perhitungan recall untuk kelas gawang :

$$Recall_{gawang} = \frac{TP}{TP + FN} \times 100\% \quad (4.14)$$

$$Recall_{gawang} = \frac{0,98}{0,98 + 0,01} \times 100\% \quad (4.15)$$

$$Recall_{gawang} = \frac{0,98}{0,99} \times 100\% = 98,99\% \quad (4.16)$$

- Perhitungan accuracy suar dilakukan menggunakan persamaan :

$$TP = 0,99$$

$$FP = 0,61$$

$$FN = 0,01$$

TN = semua sel yang tidak dibaris maupun kolom suar. Ini menunjukkan sisa 9 sel: kolom gawang (0,98+0+0,01 = 0,99) + kolom bak (0+0,98+0,02 = 1,0) + kolom background (0,61+0,25+0 = 0,86) jadi 0,99+ 1,0 + 0,86 = 2,85

$$Accuracy_{suar} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (4.17)$$

$$Accuracy_{suar} = \frac{0,99 + 2,85}{0,99 + 2,85 + 0,61 + 0,01} \times 100\% \quad (4.18)$$

$$Accuracy_{suar} = \frac{3,84}{4,46} \times 100\% = 86,09\% \quad (4.19)$$

- Perhitungan Precision suar dilakukan menggunakan persamaan :

$$Precision_{suar} = \frac{TP}{TP + FP} \times 100\% \quad (4.20)$$

$$Precision_{suar} = \frac{0,99}{0,99 + 0,61} \times 100\% \quad (4.21)$$

$$Precision_{suar} = \frac{0,99}{1,6} \times 100\% = 61,87\% \quad (4.22)$$

- Perhitungan recall suar dilakukan menggunakan persamaan :

$$Recall_{suar} = \frac{TP}{TP + FN} \times 100\% \quad (4.23)$$

$$Recall_{suar} = \frac{0,99}{0,99 + 0,01} \times 100\% \quad (4.24)$$

$$Recall_{suar} = \frac{0,99}{1} \times 100\% = 99\% \quad (4.25)$$

- Perhitungan accuracy bak dilakukan menggunakan persamaan :

$$TP = 0,98$$

$$FP = 0,25$$

$$FN = 0,02$$

TN = semua sel yang tidak dibaris maupun kolom bak. Ini menunjukkan sisa 9 sel: kolom gawang (0,98+0+0,01 = 0,99) + kolom suar (0+0,99+0,01 = 1,0) + kolom background (0,61+0,25+0 = 0,86) jadi 0,99+ 1,0 + 0,86 = 2,85

$$Accuracy_{bak} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (4.26)$$

$$Accuracy_{bak} = \frac{0,98 + 2,86}{0,98 + 2,85 + 0,25 + 0,02} \times 100\% \quad (4.27)$$

$$Accuracy_{bak} = \frac{3,84}{4,1} \times 100\% = 93,65\% \quad (4.28)$$

- Perhitungan precision bak dilakukan menggunakan persamaan :

$$Precision_{bak} = \frac{TP}{TP + FP} \times 100\% \quad (4.29)$$

$$Precision_{bak} = \frac{0,98}{0,98 + 0,25} \times 100\% \quad (4.30)$$

$$Precision_{bak} = \frac{0,98}{1,23} \times 100\% = 79,67\% \quad (4.31)$$

- Perhitungan recall bak dilakukan menggunakan persamaan :

$$Recall_{bak} = \frac{TP}{TP + FN} \times 100\% \quad (4.32)$$

$$Recall_{bak} = \frac{0,98}{0,98 + 0,02} \times 100\% \quad (4.33)$$

$$Recall_{bak} = \frac{0,98}{1} \times 100\% = 98\% \quad (4.34)$$

Tabel 4. 15 Rekapitulasi Hasil Evaluasi Model YOLOv8n

Kelas	Accuracy (%)	Precision (%)	Recall (%)
Gawang	96	86,73	98,99
Suar	86,09	61,87	99
Bak	93,65	79,67	98

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.14, ketiga kelas menunjukkan nilai recall yang tinggi, yaitu di atas 98%. Nilai recall yang tinggi menunjukkan bahwa hampir seluruh objek yang benar-benar ada berhasil terdeteksi oleh model, atau dengan kata lain

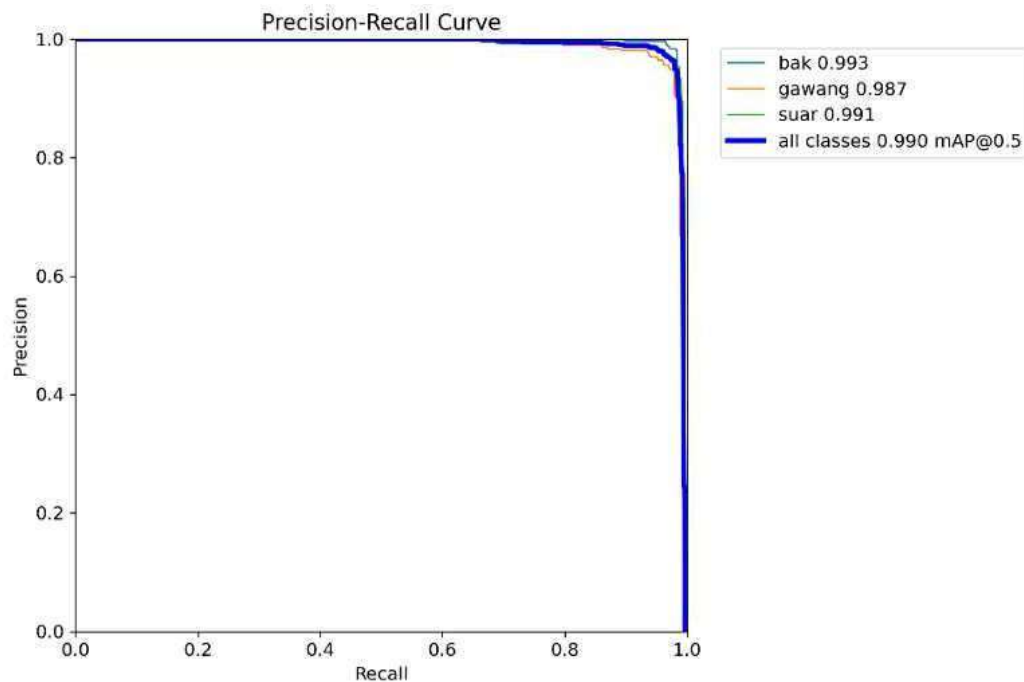
jumlah objek yang gagal terdeteksi (*false negative*) sangat kecil. Hal ini merupakan karakteristik yang penting bagi sistem penghindaran rintangan, karena kegagalan mendeteksi rintangan lebih berisiko dibandingkan kesalahan deteksi lainnya.

Nilai *precision* terlihat lebih bervariasi antar kelas, yaitu 86,73% pada gawang, 79,67% pada bak, dan 61,87% pada suar. Perbedaan ini disebabkan oleh banyaknya kasus latar belakang (*background*) yang keliru dikenali sebagai objek (*false positive*). Semakin banyak area *background* yang salah terdeteksi sebagai suatu kelas, semakin rendah nilai *precision* kelas tersebut. Pada kelas suar, nilai *false positive* paling tinggi sehingga *precision*-nya paling rendah; hal ini terjadi karena bentuk dan warna suar yang relatif sederhana membuat sebagian area latar bawah air seperti pantulan cahaya dan partikel dalam air memiliki kemiripan visual dengan suar, sehingga sesekali dikenali secara keliru sebagai objek. Sebaliknya, gawang yang memiliki struktur lebih khas lebih jarang tertukar dengan latar sehingga *precision*-nya paling tinggi.

Nilai *accuracy* pada ketiga kelas tergolong tinggi, yaitu 96% pada gawang, 93,65% pada bak, dan 86,09% pada suar. Namun demikian, pada kasus deteksi objek nilai *accuracy* cenderung tinggi karena banyaknya kasus *true negative*, yaitu ketika suatu area yang bukan objek dengan benar tidak dikenali sebagai objek tersebut. Oleh karena itu, pada evaluasi deteksi objek, nilai *precision* dan *recall* lebih representatif dibandingkan *accuracy* dalam menggambarkan kemampuan model mengenali objek.

Secara keseluruhan, kombinasi *recall* yang tinggi dan *precision* yang memadai menunjukkan bahwa model YOLOv8n mampu mendeteksi ketiga objek dengan andal. Kesalahan yang muncul didominasi oleh *false positive* terhadap latar belakang, bukan kegagalan mendeteksi objek, sehingga tidak mengganggu fungsi utama sistem dalam mengenali keberadaan rintangan. Pengaruh kesalahan deteksi tersebut juga diminimalkan pada tahap estimasi jarak, karena jarak objek dihitung dari peta kedalaman pada area deteksi, sehingga deteksi yang tidak konsisten dapat tersaring.

Proses pelatihan (*training*) model berlangsung selama kurang lebih 4 jam. Setelah pelatihan selesai, sistem menghasilkan kurva Precision-Recall yang ditunjukkan pada Gambar 4.22.

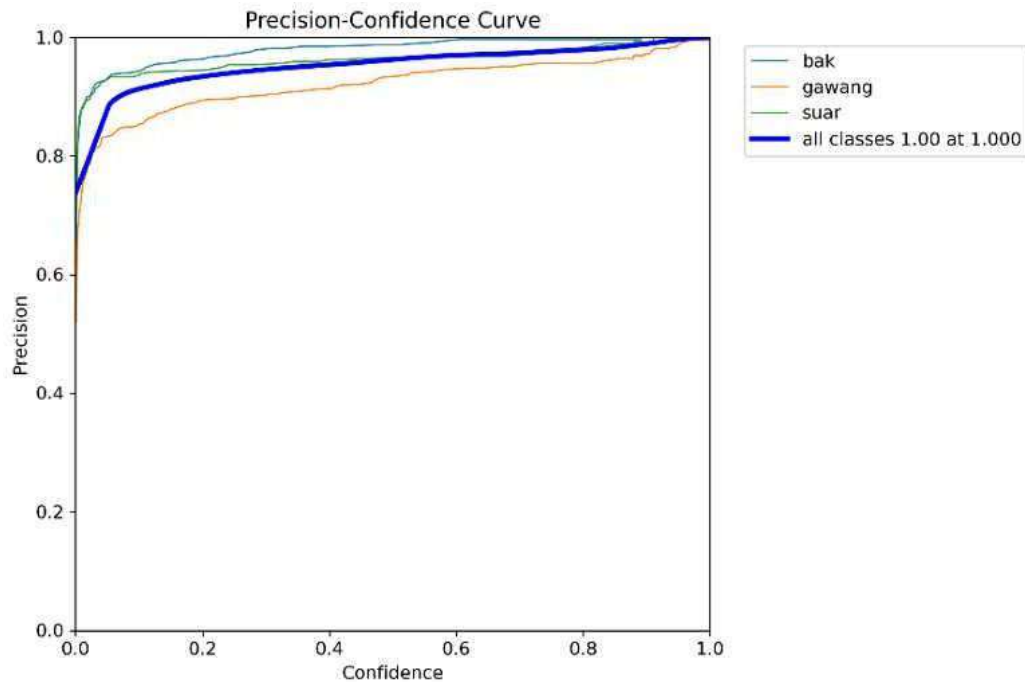


Gambar 4. 22 *Precision-Recall Curve*

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.22, diperoleh nilai *mean Average Precision* (Map@0.5) sebesar 99%, dengan rincian nilai *Average Precision* untuk kelas gawang sebesar 98,7%, kelas suar sebesar 99,1%, dan kelas bak sebesar 99,3%. Nilai tersebut menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mendeteksi dan mengklasifikasikan objek sesuai dengan kelasnya. Tingginya nilai Map mengindikasikan bahwa model mampu menghasilkan prediksi yang akurat dengan keseimbangan yang baik antara *precision* dan *recall*. Selain itu, perbedaan nilai antar kelas menunjukkan bahwa model lebih optimal dalam mendeteksi suar dan bak dibandingkan gawang, yang kemungkinan dipengaruhi oleh variasi bentuk dan ukuran objek. Struktur fisik gawang yang berupa kerangka berongga membuatnya lebih rentan mengalami distorsi visual akibat pergerakan air dibandingkan dengan suar dan bak yang memiliki wujud lebih pejal. Meskipun demikian, capaian akurasi di atas 98 persen untuk seluruh kelas ini secara efektif menjamin minimnya deteksi palsu (*false positive*) yang umumnya dipicu oleh pantulan cahaya maupun partikel tersuspensi di dalam air. Keandalan identifikasi yang tinggi ini sangat krusial karena stabilitas kotak pembatas (*bounding box*) merupakan syarat mutlak agar titik pusat target dapat diekstrak secara presisi untuk

memandu arah manuver wahana. Secara keseluruhan, hasil ini menunjukkan bahwa model deteksi objek yang dilatih memiliki performa yang baik dan dapat diandalkan untuk digunakan dalam sistem navigasi AUV untuk melewati rintangan pada lintasan KRBAI.



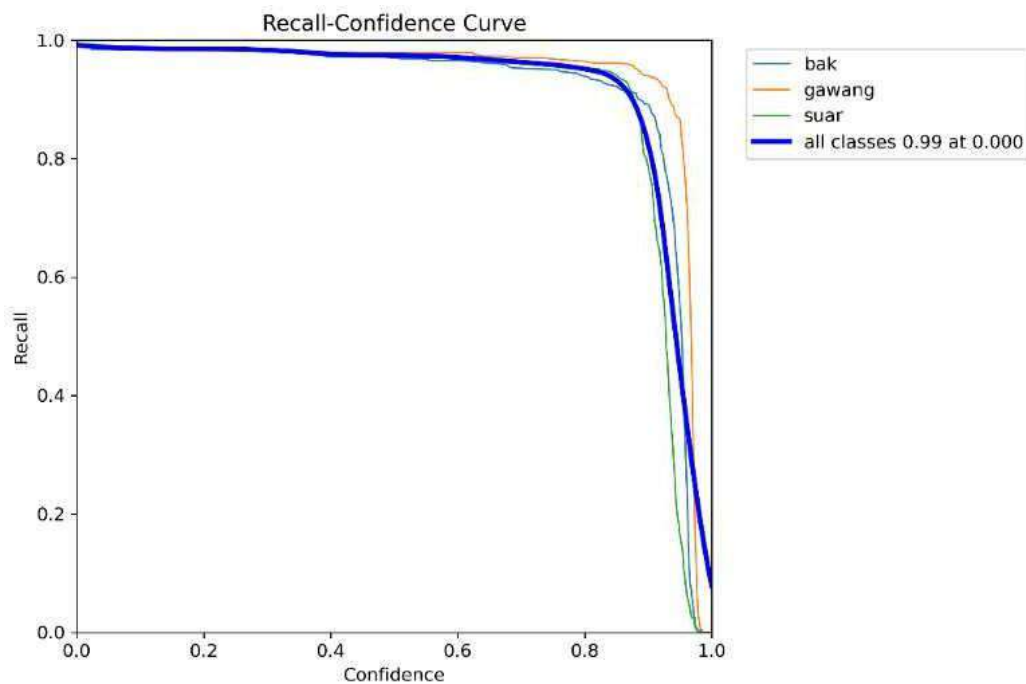
Gambar 4. 23 *Precision-Confidence Curve*

(Sumber : Dokumentasi Pribadi)

Gambar 4.23 menampilkan *Precision-Confidence Curve*, yaitu grafik yang menunjukkan hubungan anatar nilai *confidence* (tingkat keyakinan model terhadap prediksi) dengan *precision* (tingkat ketepatan prediksi). Dari kurva tersebut terlihat bahwa baik untuk kelas gawang, suar, dan bak. Model memiliki nilai precision yang tinggi pada Sebagian besar rentang confidence.

Nilai precision terus meningkat seiring bertambahnya *confidence* dan mencapai nilai maksimum sebesar 1.00 pada confidence sekitar 0.851, sebagaimana ditunjukkan oleh kurva *all classes*. Hal ini menunjukkan bahwa pada Tingkat *confidence* tersebut, model mampu menghasilkan prediksi yang sangat akurat tanpa kesalahan (*false positive*). Secara keseluruhan, hal ini mengindikasikan bahwa model memiliki performa deteksi yang sangat baik, khususnya dalam menjaga Tingkat ketepatan prediksi. Dengan demikian, model dapat diandalkan untuk

digunakan dalam sistem deteksi objek pada AUV yang membutuhkan akurasi tinggi dan respons yang stabil.

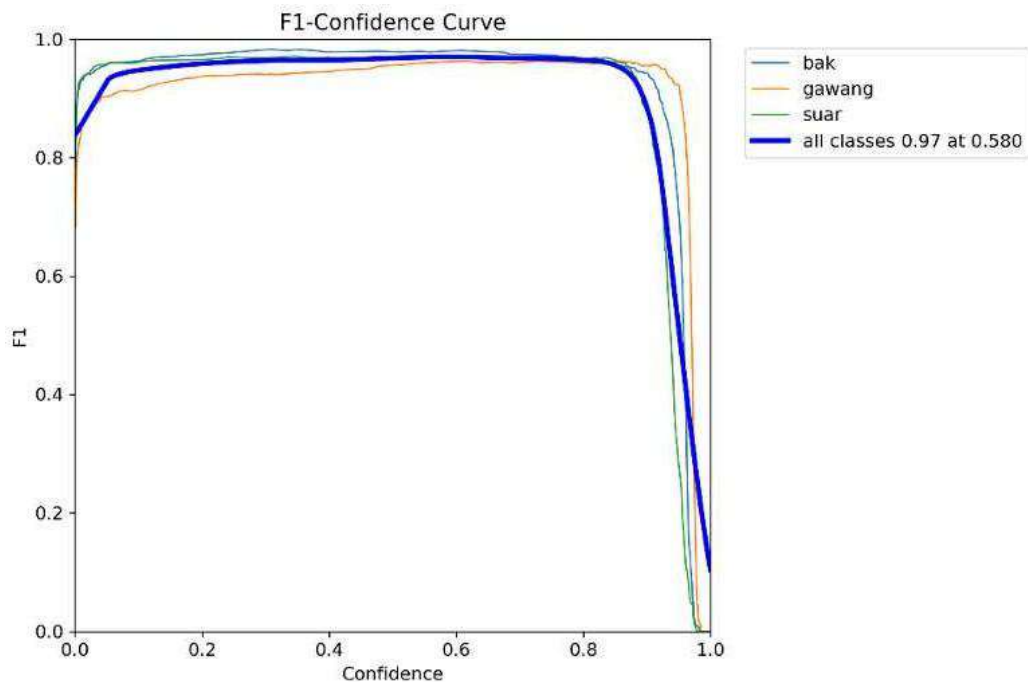


Gambar 4. 24 *Recall - Confidence Curve*

(Sumber : Dokumentasi Pribadi)

Gambar 4.24 menampilkan *Recall-Confidence Curve*, yaitu grafik yang menunjukkan hubungan antara nilai *confidence* dan *recall* dari model deteksi objek. *Recall* menggambarkan kemampuan model dalam mendeteksi seluruh objek yang seharusnya teridentifikasi. Berdasarkan garfik nilai *recall* untuk kelas gawang, suar, dan bak berada pada tingkat yang tinggi pada rentang *confidence* rendah hingga menengah. Hal ini menunjukkan bahwa model mampu mendeteksi sebagian besar objek dengan baik pada berbagai Tingkat keyakinan. Namun, seiring meningkatnya nilai *confidence*, *recall* mulai mengalami penurunan yang cukup signifikan terutama Ketika mendekati nilai di atas 0.9.

Kondisi ini menunjukkan adanya *trade-off* antara *recall* dan *precision*, Dimana semakin tinggi *confidence* yang digunakan, model menjadi lebih selektif sehingga beberapa objek tidak terdeteksi (*false negative*). Meskipun demikian, secara keseluruhan kurva ini menunjukkan bahwa model memiliki performa yang baik dan konsisten dalam mendeteksi objek, terutama pada rentang *confidence* yang optimal untuk penggunaan sistem secara *real-time*.



Gambar 4. 25 *F-Score Curve*

(Sumber : Dokumentasi Pribadi)

Gambar 4.25 menampilkan *F1-Confidence Curve*, yaitu grafik yang menunjukkan hubungan antara nilai confidence dan F1-score dari model deteksi objek. *F1-score* merupakan metrik yang menggabungkan *precision* dan *recall*, sehingga mencerminkan keseimbangan antara ketepatan dan kelengkapan dalam proses deteksi. Berdasarkan garfik, nilai *F1-score* untuk kelas gawang, suar, dan bak berada pada tingkat yang tinggi pada rentang *confidence* rendah hingga menengah. Nilai maksimum untuk seluruh kelas tercapai sebesar 0.97 pada confidence sekitar 0.580, sebagaimana ditunjukkan oleh kurva *all classes*. Hal ini menunjukkan bahwa pada tingkat *confidence* tersebut, model memiliki performa yang paling optimal dalam menyeimbangkan *precision* dan *recall*.

Seiring meningkatnya nilai *confidence*, *F1-score* mulai mengalami penurunan, terutama Ketika mendekati nilai tinggi di atas 1.00. Hal ini mengindikasikan bahwa model menjadi lebih selektif, sehingga meskipun *precision* meningkat, nilai *recall* menurun dan berdampak pada penurunan *F1-score*. Secara keseluruhan, kurva ini menunjukkan bahwa model memiliki performa yang baik dan stabil, dengan titik optimal pada confidence sekitar 0.58, yang dapat digunakan

sebagai acuan dalam menentukan ambang batas (*threshold*) pada sistem deteksi objek.

4.5 Hasil Estimasi Jarak Objek Berbasis *Stereo Vision*

Pengujian akurasi deteksi dan estimasi jarak menggunakan sensor kamera stereo bertujuan untuk memilai sejauh mana kamera ini dapat mendeteksi objek seperti gawang, suar, dan bak dengan akurasi yang tepat pada jarak tertentu. Dalam hal ini, pengujian dilakukan pada jarak tertentu.

4.5.1 Pembacaan Jarak Awal *Stereo Vision* (Sebelum Koreksi)

Pengujian tahap awal dilakukan untuk mengetahui jarak yang dihasilkan sistem *stereo vision* secara langsung dari perhitungan triangulasi, sebelum koreksi diterapkan. Jarak ini diperoleh dari nilai disparitas antara citra kamera kiri dan kanan menggunakan persamaan triangulasi dengan parameter panjang fokus dan *baseline* hasil kalibrasi.

Pengujian dilakukan dengan menempatkan objek pada beberapa jarak acuan yang diukur secara langsung menggunakan meteran, kemudian dicatat jarak yang terbaca oleh sistem. Hasil pembacaan jarak sebelum koreksi ditunjukkan pada Tabel 4.16.

Tabel 4. 16 Data Pembacaan Jarak dari Kamera Stereo (Sebelum Kalibrasi)

No	Jarak Aktual (m)	Jarak Terbaca (m)	Error (%)
1	0,4	0,332	17
2	0,6	0,565	5,833
3	0,8	0,565	29,375
4	1,0	0,655	34,5
5	1,4	0,874	37,57
6	1,6	0,991	38,062
7	1,8	1,100	38,888
8	2,0	1,197	40,15

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.15, terlihat bahwa jarak yang terbaca oleh sistem secara konsisten lebih kecil daripada jarak sebenarnya. Pada jarak 1,0m sistem membaca jarak sekitar 0,655m. Penyimpangan ini bersifat sistematis, yaitu selisihnya mengikuti pola yang teratur seiring bertambahnya jarak, bukan kesalahan acak. Penyimpangan yang teratur tersebut disebabkan oleh pembiasan cahaya (refraksi) ketika cahaya melewati antarmuka antara air dan housing kamera, yang mengubah sudut datang cahaya sehingga objek tampak berada pada jarak yang

berbeda dari posisi sebenarnya. Karena penyimpangan ini bersifat sistematis, penyimpangan dapat dimodelkan dan dikoreksi, aygn diuraikan pada sub -bab berikutnya

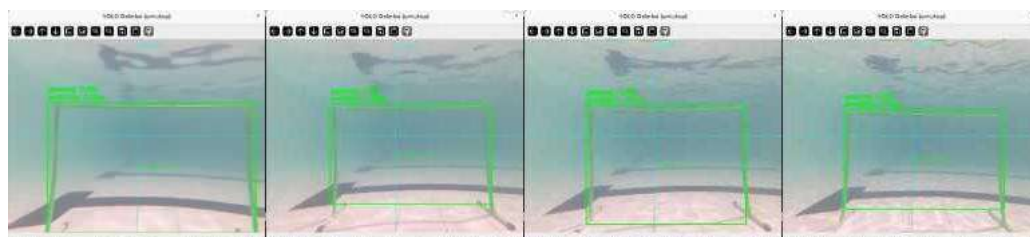
4.5.2 Koreksi Jarak Hasil Kalibrasi

Penyimpangan jarak yang bersifat sistematis pada pembacaan awal dikompensasi melalui kalibrasi koreksi jarak. Koreksi dilakukan dengan mencari hubungan antara jarak terbaca dan jarak sebenarnya menggunakan regresi linear, sehingga diperoleh persamaan koreksi:

$$Z_{koreksi} = a \times Z_{ukur} + b \quad (4.28)$$

Dengan a merupakan faktor skala dan b merupkan offset. Berdasarkan data pengukuran di dalam air, diperoleh nilai $a = 1,8174$ dan $b = -0,1919$. Penerapan persamaan koreksi tersebut terhadap pembacaan jarak menghasilkan jarak jauh lebih mendekati jarak sebenarnya yang ditampilkan pada subab berikutnya.

4.5.3 Pengujian Jarak Objek Gawang



Gambar 4. 26 Hasil Pengujian Deteksi Gawang pada Berbagai Jarak (1.2m, 1.6m, 2.0m, 2.4m)

(Sumber : Dokumentasi Pribadi)

Tabel 4. 17 Hasil Pengujian Deteksi Gawang

Pengujian	Pembacaan Aktual (m)	Jarak Terbaca (m)	Error (%)	Confidence
1	1,2	2,74	128	0,92
2	1,6	2,70	68,75	0,96
3	2,0	1,83	8,5	0,95
4	2,4	1,54	35,83	0,97
5	2,8	3,32	18,57	0,96
6	3,2	3,45	7.81	0,96
Rata – Rata			44,57	0,95

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.17, pengujian deteksi jarak pada objek gawang menghasilkan nilai *error* yang tinggi dengan rata-rata sebesar 44,57%. Nilai *error* sangat bervariasi antar pengujian, mulai dari 7,81% pada pengujian keenam hingga

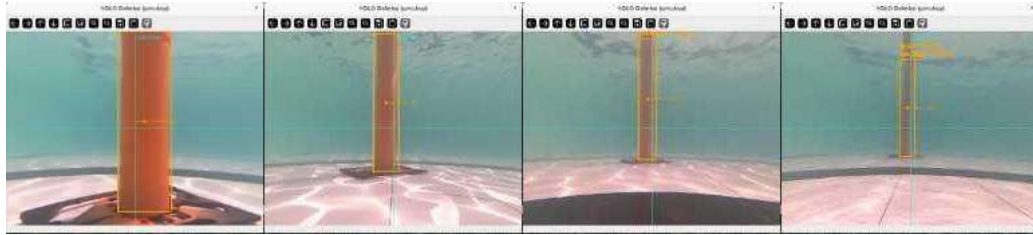
128% pada pengujian pertama. Meskipun nilai *confidence* deteksi konsisten tinggi (minimal 0,92 pada seluruh pengujian dan tertinggi di antara ketiga objek dengan rata-rata 0,95), keberhasilan deteksi objek oleh YOLO ternyata tidak menjamin keakuratan pembacaan jaraknya.

Selain nilainya yang besar, penyimpangan jarak gawang juga tidak konsisten arahnya. Pada beberapa pengujian jarak terbaca jauh lebih besar dari jarak sebenarnya (pengujian 1 dan 2), sedangkan pada pengujian lain jarak terbaca justru lebih kecil (pengujian 3 dan 4). Ketidakteraturan arah penyimpangan ini menunjukkan bahwa kesalahan pembacaan bersifat acak, bukan penyimpangan sistematis. Hal ini berbeda dengan penyimpangan akibat refraksi yang bersifat teratur dan dapat dikoreksi; kesalahan yang acak seperti pada gawang tidak dapat diperbaiki melalui kalibrasi karena tidak mengikuti pola tertentu.

Tingginya *error* pada pembacaan jarak gawang disebabkan oleh bentuk gawang yang berongga pada bagian tengahnya. Ketika sistem membaca jarak pada area deteksi gawang, sebagian besar area tersebut berupa ruang kosong (air) dan latar belakang yang terlihat menembus bukaan gawang, bukan permukaan padat gawang itu sendiri. Akibatnya, algoritma SGBM menghitung disparitas dari campuran antara rangka gawang, air, dan objek latar belakang yang posisinya berubah-ubah pada setiap sudut pandang. Nilai disparitas yang dihasilkan menjadi tidak stabil, sehingga jarak yang terbaca melompat- lompat tanpa pola yang tetap. Kondisi ini konsisten dengan hasil peta kedalaman gawang yang memperlihatkan bagian tengah gawang dipenuhi derau, sementara hanya kedua tiang gawang yang terpetakan dengan jelas.

Berdasarkan temuan tersebut, estimasi jarak gawang tidak digunakan sebagai acuan dalam proses penghindaran maupun navigasi. Sebagai gantinya, navigasi terhadap gawang dilakukan menggunakan posisi horizontal gawang pada citra (arah kiri–kanan objek terhadap pusat kamera), bukan jaraknya. Pendekatan ini dipilih karena posisi horizontal gawang tetap dapat dibaca secara andal meskipun jaraknya tidak akurat, sehingga wahana tetap dapat diarahkan menuju dan melewati gawang dengan tepat. Dengan demikian, tingginya *error* jarak gawang tidak mengganggu keberhasilan misi, karena fungsi jarak memang tidak diandalkan untuk objek ini.

4.5.4 Pengujian Jarak Objek Suar



Gambar 4. 27 Hasil Pengujian Deteksi Suar pada Berbagai Jarak (0.4m, 1.2m, 2.0m, 2.8m)

(Sumber : Dokumentasi Pribadi)

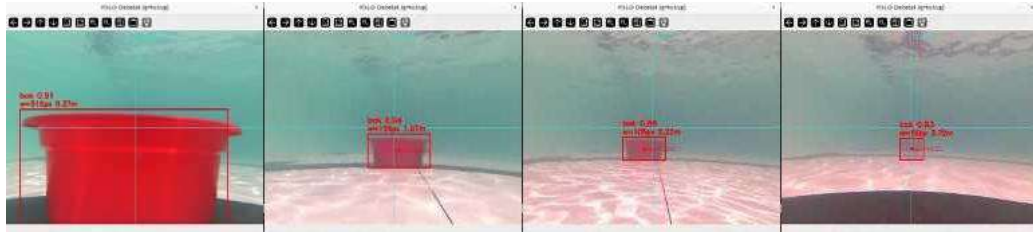
Tabel 4. 18 Hasil Pengujian Deteksi Suar

Pengujian	Jarak Aktual (m)	Pembacaan Stereo (m)	<i>Error (%)</i>	<i>Confidence</i>
1	0,4	0,41	2,5	0,89
2	0,8	0,82	5	0,84
3	1,2	1,30	8,333	0,88
4	1,6	1,72	7,5	0,87
5	2,0	2,13	6,5	0,87
6	2,4	2,45	2,0833	0,86
7	2,8	2,82	0,71	0,88
Rata – Rata			4,6609	0,87

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.18, suar berhasil terdeteksi secara konsisten pada seluruh titik pengujian dari jarak 0,4 hingga 2,8 meter. Setelah koreksi jarak diterapkan, estimasi jarak suar memiliki galat rata-rata sebesar 4,66%, dengan galat terkecil 0,71% pada jarak 2,8 meter dan galat terbesar 8,33% pada jarak 1,2 meter. Pembacaan jarak cenderung sedikit lebih besar daripada jarak sebenarnya (*overestimate*) pada hampir seluruh titik, namun selisihnya kecil dan konsisten sehingga tidak mengganggu fungsi penghindaran. Nilai *confidence* deteksi berada pada rentang 0,84 hingga 0,89 dengan rata-rata sekitar 0,87, relatif lebih rendah dibandingkan objek lain. Hal ini disebabkan oleh dimensi suar yang kecil sehingga hanya menempati area *piksel* yang terbatas dalam *frame*. Meskipun demikian, deteksi tetap stabil pada seluruh titik pengujian dan galat jaraknya tergolong rendah, sehingga estimasi jarak suar dapat diandalkan sebagai acuan penghindaran hal yang penting karena suar merupakan rintangan utama yang harus dihindari oleh algoritma APF.

4.5.5 Pengujian Jarak Objek Bak



Gambar 4. 28 Hasil Pengujian Deteksi Suar pada Berbagai Jarak (0.4m, 1.6m, 2.8m, 0.8m)

(Sumber : Dokumentasi Pribadi)

Tabel 4. 19 Hasil Pengujian Deteksi Bak

Pengujian	Jarak Aktual (m)	Pembacaan Camera (m)	Error (%)	confidence
1	0,4	0,42	5	0,92
2	0,8	0,76	5	0,96
3	1,2	1,17	2.5	0,96
4	1,6	1,62	1.25	0,94
5	2,0	1,90	5	0,93
6	2,4	2,22	7.5	0,89
7	2,8	2,65	5.3	0,85
Rata – Rata Error			4.507	0,92

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.19, bak terdeteksi secara konsisten pada seluruh titik pengujian dari jarak 0,4 hingga 2,8 meter dengan nilai *confidence* rata-rata sekitar 0,92 yang tergolong tinggi karena ukurannya besar dan warnanya kontras. Setelah koreksi jarak diterapkan, estimasi jarak bak memiliki galat rata-rata sebesar 4,51%, dengan galat terkecil 1,25% pada jarak 1,6 meter dan galat terbesar 7,5% pada jarak 2,4 meter. Pembacaan jarak sangat mendekati jarak sebenarnya pada rentang dekat hingga menengah, sedangkan pada jarak yang lebih jauh estimasi cenderung sedikit lebih pendek (*underestimate*) seiring menurunnya resolusi disparitas. Secara keseluruhan, estimasi jarak bak tergolong akurat pada rentang kerja misi sehingga dapat diandalkan sebagai acuan jarak, sementara arah menuju bak tetap dapat ditentukan dari posisi lateral objek pada citra (*cx_norm*).

4.6 Pengujian Sistem Obstacle Avoidance pada Simulasi

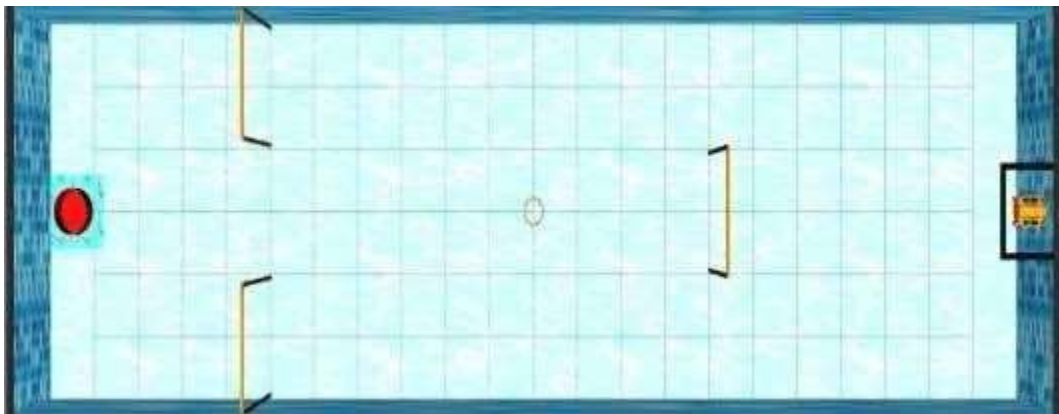
Pengujian simulasi dilakukan untuk memvalidasi implementasi metode Artificial Potential Field (APF) berbasis *stereo depth* pada AUV KRBAI sebelum diimplementasikan pada perangkat keras nyata. Simulasi menggunakan platform

Gazebo dengan model AUV KRBAI yang telah dikonfigurasi sesuai dengan spesifikasi fisik AUV nyata. Lingkungan simulasi dirancang menyerupai arena kompetisi KRBAI, yang terdiri dari kolam renang berukuran panjang 10 meter, lebar 6 meter, dan tinggi 1.5 meter. Dua gawang yang harus dilewati AUV, satu suar sebagai rintangan, dan sebuah bak merah sebagai titik akhir misi.

Pengujian simulasi terbagi menjadi dua skenario yang dirancang untuk mengevaluasi kontribusi kamera *stereo depth* dalam sistem navigasi APF. Skenario A menggunakan *waypoint* eksplisit *AVOID_SUAR* sebagai jalur referensi tambahan, sementara Skenario B menghapus *waypoint* tersebut sehingga penghindaran rintangan sepenuhnya bergantung pada pembacaan kamera *stereo depth* secara *real-time*.

4.6.1 Konfigurasi Lingkungan Simulasi

Pengujian simulasi dilakukan menggunakan platform Gazebo dengan model AUV KRBAI yang telah dikonfigurasi sesuai spesifikasi fisik AUV nyata. Lingkungan simulasi dirancang menyerupai arena kompetisi KRBAI yang terdiri dari kolam berukuran panjang 10 meter, lebar 5 meter, dan tinggi 1.5 meter, dengan dua gawang yang harus dilewati AUV, satu suar sebagai rintangan, dan sebuah bak merah sebagai titik akhir misi.

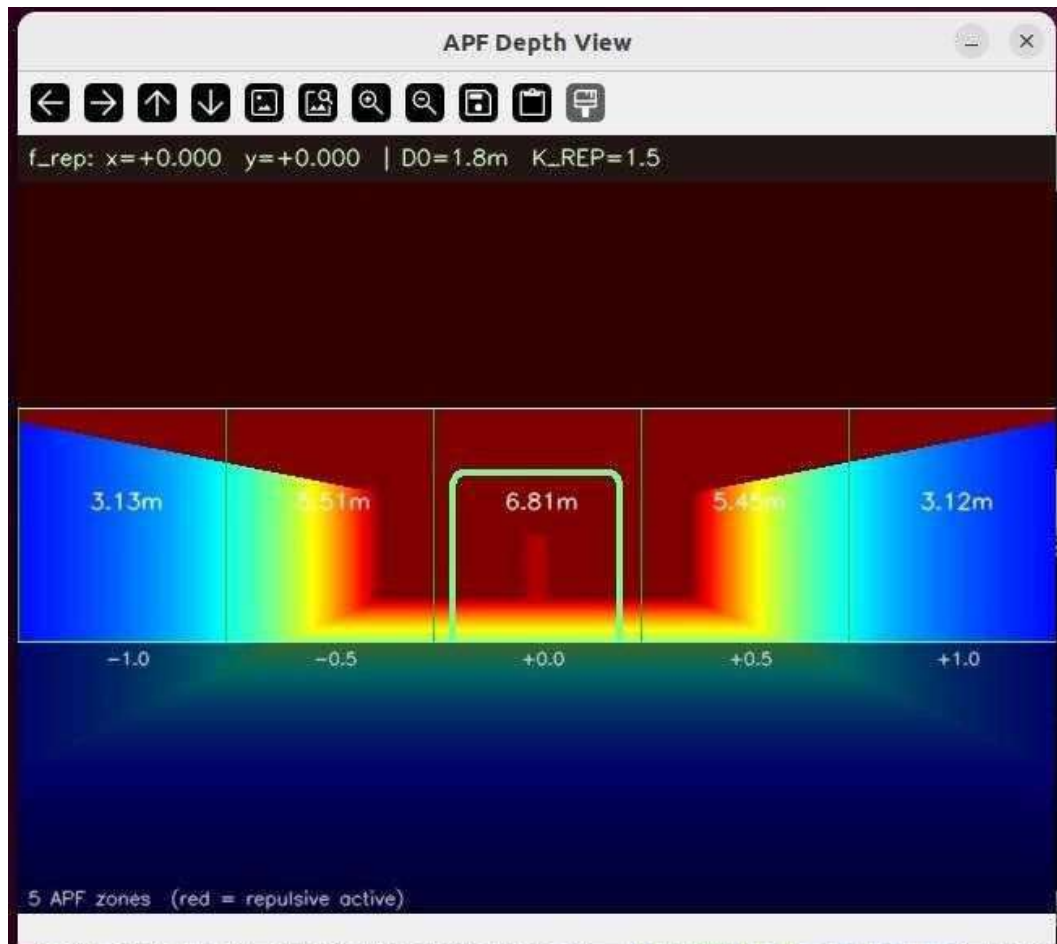


Gambar 4. 29 Tampilan *World* Gazebo Arena KRBAI

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.29, tata letak arena disusun sesuai konfigurasi kompetisi KRBAI dimana AUV ditempatkan pada titik *spawn* di sisi kanan kolam. Bak merah diletakkan pada sisi kiri sebagai titik akhir misi, sementara dua gawang dan satu suar terdistribusi di sepanjang lintasan navigasi yang harus dilalui AUV.

Untuk mendukung persepsi visual selama simulasi, AUV dilengkapi kamera stereo virtual yang menghasilkan citra RGB dan *depth map*, merepresentasikan keluaran pipeline *stereo depth* kamera DWE exploreHD pada perangkat keras nyata. Hasil pengolahan kamera stereo dalam simulasi ditampilkan pada Gambar 4.30.

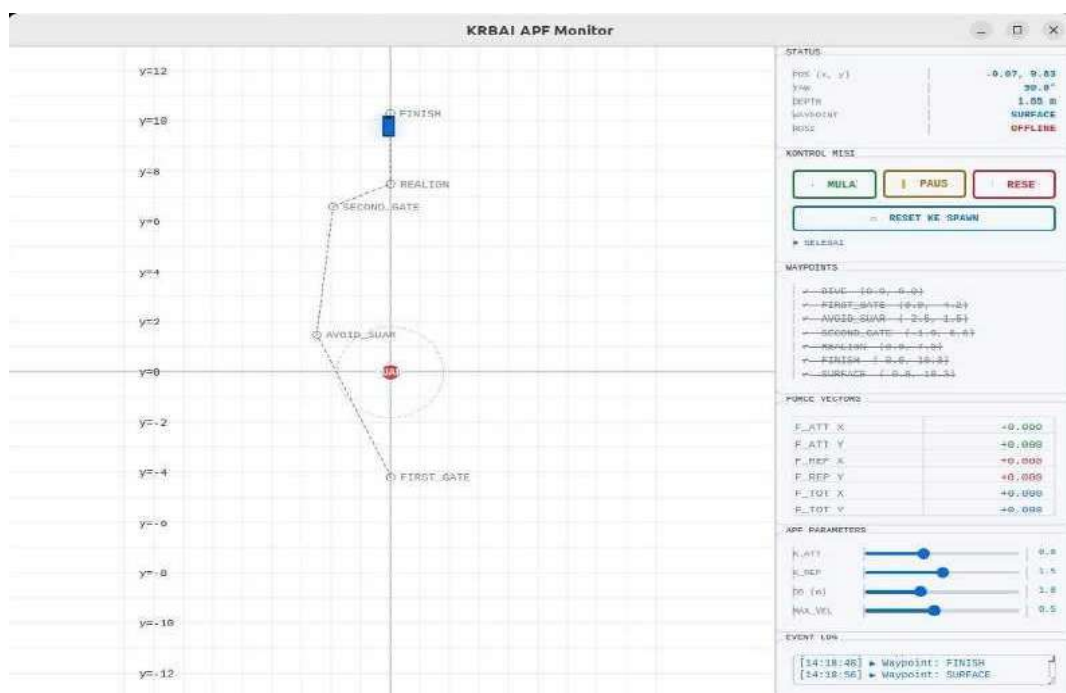


Gambar 4. 30 Tampilan APF *Depth View* dari Kamera Stereo AUV

(Sumber : Dokumentasi Pribadi)

Gambar 4.30 menampilkan visualisasi *depth image* yang dibagi menjadi lima zona lateral guna mengidentifikasi arah datangnya rintangan. Pembagian zona spasial ini digunakan sebagai acuan utama dalam perhitungan vektor perintah geser pada algoritma Artificial Potential Field (APF). Setiap zona secara dinamis menyajikan nilai jarak terdekat rintangan dalam satuan meter. Apabila terdeteksi objek yang masuk ke dalam batas aman (jarak aktual $\leq$ jarak ambang D_0), zona tersebut akan disorot dengan warna merah sebagai indikator bahwa perintah geser sedang aktif. Visualisasi ini sangat penting karena memungkinkan peneliti untuk

memvalidasi secara real-time tingkat akurasi dan kontribusi kamera *stereo depth* dalam menjalankan sistem *obstacle avoidance*. Sementara itu, untuk memantau status operasional kendaraan secara keseluruhan selama simulasi berlangsung, telah dikembangkan sebuah antarmuka grafis yang dinamakan aplikasi KRBAI APF Monitor, sebagaimana ditampilkan pada Gambar 4.31. Aplikasi ini dirancang sebagai pusat pemantauan (*dashboard*) untuk memvisualisasikan berbagai parameter sistem secara langsung, sehingga mempermudah proses pengamatan, evaluasi, dan identifikasi masalah (*debugging*) tanpa perlu membaca data mentah secara manual.



Gambar 4. 31 Tampilan Aplikasi KRBAI APF Monitor

(Sumber : Dokumentasi Pribadi)

Aplikasi KRBAI APF Monitor menampilkan posisi AUV pada bidang X-Y, *waypoint* aktif, status koneksi ROS2, komponen gaya APF (F_{ATT} , F_{REP} , F_{TOT}), serta event log yang mencatat transisi antar *waypoint* selama misi. Aplikasi ini juga menyediakan kontrol parameter APF (K_{ATT} , K_{REP} , D_0 , MAX_VEL) yang dapat disesuaikan secara *real-time* untuk keperluan tuning sistem.

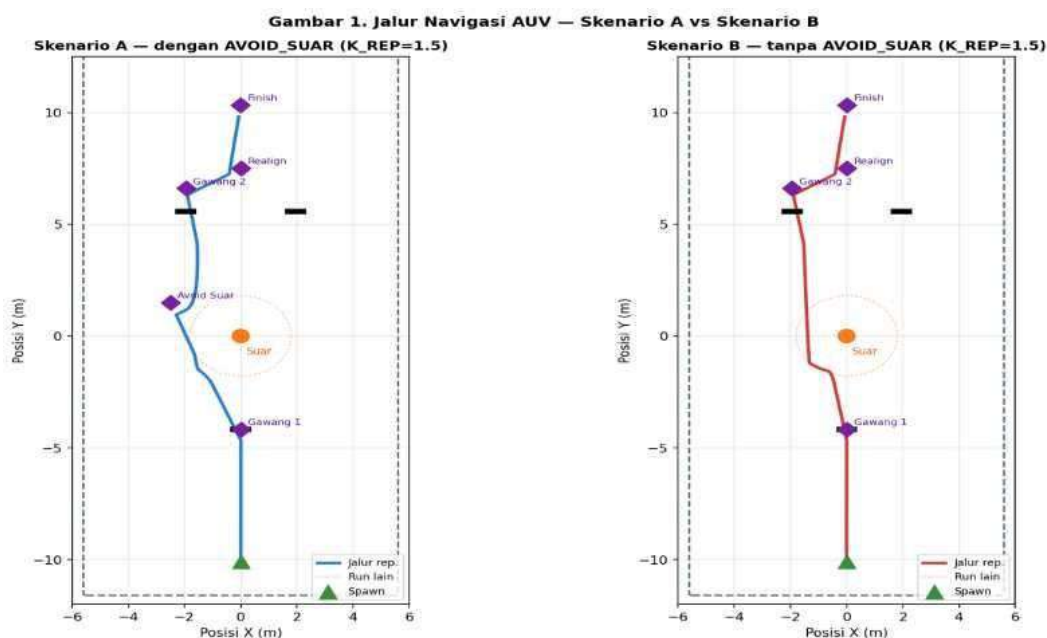
Pengujian simulasi dilakukan dengan konfigurasi parameter APF yang telah ditentukan berdasarkan karakteristik lingkungan kolam simulasi. Parameter yang

digunakan adalah sebagai berikut: gain gaya Tarik semu $K_{ATT} = 0,8$; gain perintah geser $K_{REP} = 1,5$; jarak pengaruh *obstacle* $D0 = 1,8$ meter; dan kecepatan maksimum horizontal $MAX_VEL = 0,5$ m/s. Konfigurasi *waypoint* navigasi mencakup tujuh titik tujuan yang meliputi fase penyelaman (*DIVE*), gawang pertama (*FIRST_GATE*), penghindaran suar (*AVOID_SUAR*, khusus Skenario A), gawang kedua (*SECOND_GATE*), pelurusan jalur (*REALIGN*), titik akhir (*FINISH*), dan permukaan (*SURFACE*).

Setiap skenario dijalankan sebanyak lima kali pengulangan untuk memperoleh data yang representatif. Setiap run direkam secara otomatis oleh modul data logger yang terintegrasi dalam *node apf_mission*, mencatat posisi AUV, komponen gaya APF, dan status pendeteksian *obstacle* setiap 100 milidetik.

4.6.2 Hasil Pengujian Skenario A – Dengan *Waypoint* AVOID_SUAR

Skenario A merupakan pengujian dengan konfigurasi *waypoint* lengkap termasuk titik AVOID_SUAR yang ditempatkan pada koordinat (-2,5; 1,5) meter untuk memandu AUV melewati sisi kiri suar sebelum menuju gawang kedua. Pada skenario ini, perintah geser dari kamera *stereo depth* berfungsi sebagai lapisan pengaman tambahan yang memperkuat penghindaran *obstacle* yang telah direncanakan melalui *waypoint*.



Gambar 4. 32 Jalur Navigasi AUV pada Skenario A (kiri) dan Skenario B (kanan)

(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.32, terlihat bahwa pada Skenario A AUV berhasil mengikuti jalur navigasi yang telah direncanakan. AUV melalui seluruh *waypoint* secara berurutan mulai dari titik spawn di koordinat (0; -10,1) meter, melewati gawang pertama, menjauhi suar melalui *waypoint AVOID_SUAR*, melewati gawang kedua, dan mencapai bak merah sebagai titik akhir misi. Pada seluruh lima pengulangan, jalur yang ditempuh sangat konsisten dengan deviasi posisi yang kecil antar run.

Tabel 4. 20 Hasil Pengujian Skenario A (dengan AVOID_SUAR)

Run	Waktu (s)	Jalur (m)	f_{rep} Aktif (%)	Dist. Suar (m)	Status
1	57,60	21,58	18,5	1,83	Berhasil
2	57,50	21,58	18,7	1,82	Berhasil
3	57,60	21,58	18,5	1,82	Berhasil
4	57,60	21,58	18,5	1,82	Berhasil
5	57,60	21,58	18,7	1,82	Berhasil
Rata-rata	57,58	21,58	18,58	1,82	5/5 (100%)

(Sumber : Dokumentasi Pribadi)

Berdasarkan hasil pengujian pada Tabel 4.20, seluruh lima pengulangan Skenario A berhasil diselesaikan (success rate 100%). Rata-rata waktu tempuh yang diperoleh adalah 57,58 detik dengan panjang jalur rata-rata 21,58 meter. Perintah geser dari kamera *stereo depth* (f_{rep}) aktif selama rata-rata 18,58% dari total durasi misi, menunjukkan bahwa sensor berhasil mendeteksi keberadaan suar dalam jangkauan $D0 = 1,8$ meter. Jarak minimum rata-rata AUV terhadap suar adalah 1,82 meter, tepat di tepi luar batas pengaruh $D0 = 1,8$ meter menunjukkan bahwa pada Skenario A *waypoint AVOID_SUAR* telah memandu AUV menjauh sebelum perintah geser suar aktif signifikan, sehingga aktivasi f_{rep} sebesar 18,58% didominasi oleh deteksi elemen arena lain seperti dinding kolam dan struktur gawang.

4.6.3 Hasil Pengujian Skenario B – Tanpa *Waypoint AVOID_SUAR*

Skenario B merupakan pengujian dengan konfigurasi *waypoint* yang tidak menyertakan titik *AVOID_SUAR*, sehingga penghindaran terhadap suar sepenuhnya bergantung pada perintah geser yang dihasilkan dari pembacaan

kamera *stereo depth* secara *real-time*. Pada skenario ini, AUV harus mendeteksi keberadaan suar secara mandiri melalui analisis citra kedalaman dan merespons dengan menghasilkan perintah geser yang memadai untuk menghindari tabrakan.

Pengolahan data kamera *stereo depth* dilakukan dengan membagi *frame* citra kedalaman menjadi lima zona lateral. Setiap zona dihitung nilai persentil ke-10 dari kedalaman *piksel* sebagai estimasi jarak *obstacle* terdekat. Perintah geser kemudian dihitung menggunakan persamaan APF: $f_{rep} = K_{REP} \times (1/d - 1/D_0) / d^2$, dengan nilai nol apabila jarak melebihi batas D_0 .

Berdasarkan Gambar 4.15, jalur navigasi AUV pada Skenario B memiliki pola yang berbeda dibandingkan Skenario A. Tanpa *waypoint AVOID_SUAR* sebagai panduan jalur, AUV bergerak lebih langsung menuju gawang kedua namun dengan koreksi arah yang lebih aktif saat mendekati area suar, sebagai respons terhadap perintah geser dari kamera *stereo depth*.

Tabel 4. 21 Hasil Pengujian Skenario B (tanpa AVOID_SUAR)

Run	Waktu (s)	Jalur (m)	f_{rep} Aktif (%)	Dist. Suar (m)	Status
1	54,70	21,29	25,9	1,39	Berhasil
2	54,80	21,27	25,5	1,37	Berhasil
3	54,70	21,27	25,7	1,37	Berhasil
4	54,70	21,27	25,7	1,37	Berhasil
5	54,70	21,27	25,5	1,37	Berhasil
Rata-rata	54,72	21,27	25.64	1,37	5/5 (100%)

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.21, seluruh lima pengulangan Skenario B juga berhasil diselesaikan dengan success rate 100%. Rata-rata waktu tempuh yang diperoleh adalah 54,72 detik, lebih cepat 2,86 detik dibandingkan Skenario A. Hal ini disebabkan oleh tidak adanya *waypoint AVOID_SUAR* yang menambah jarak tempuh. Panjang jalur rata-rata yang lebih pendek yaitu 21,27 meter juga mengkonfirmasi hal tersebut. Aktivasi perintah geser f_{rep} pada Skenario B mencapai rata-rata 25,64% dari total durasi misi, lebih tinggi 7,06% dibandingkan Skenario A, yang membuktikan bahwa kamera *stereo depth* lebih aktif bekerja

sebagai sensor *obstacle avoidance* ketika tidak ada *waypoint* eksplisit sebagai panduan jalur. Jarak minimum rata-rata AUV terhadap suar adalah 1,37 meter, lebih dekat dibandingkan Skenario A namun masih dalam batas aman mengingat dimensi fisik AUV.

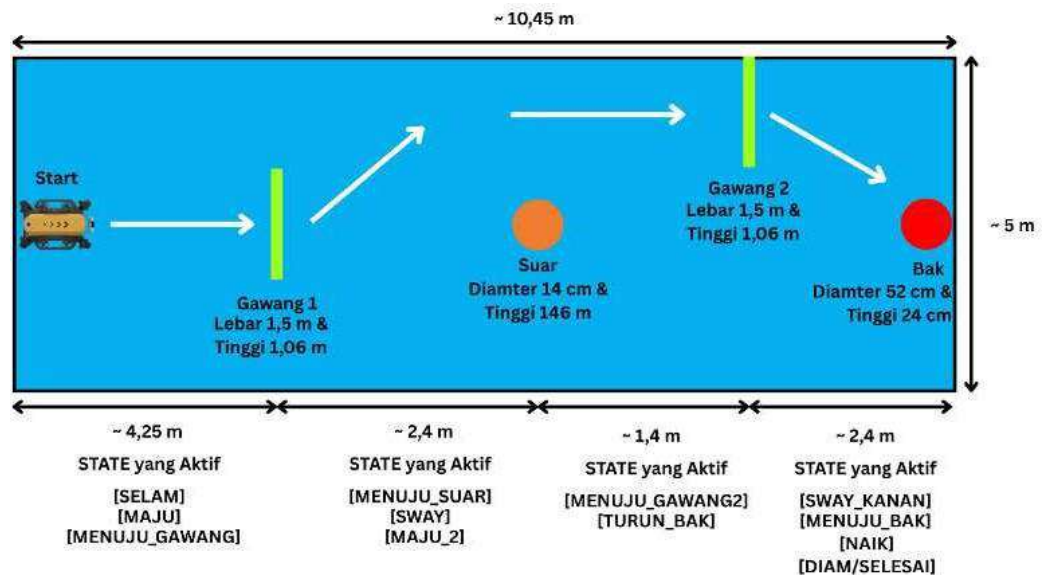
4.7 Pengujian Sistem *Obstacle Avoidance* pada Kolam Nyata

Pengujian pada kolam nyata dilakukan untuk memvalidasi kinerja sistem *obstacle avoidance* berbasis *stereo depth* dan Artificial Potential Field (APF) pada kondisi operasional sebenarnya, setelah seluruh *node* terverifikasi berjalan dan perilaku dasar tervalidasi pada simulasi. Pengujian disusun dalam dua tahap. Tahap pertama menguji ketepatan arah perintah geser yang dihasilkan algoritma APF terhadap rintangan tunggal, rintangan ganda, dan rintangan bergerak. Tahap kedua menguji keberhasilan misi secara utuh pada dua kondisi, yaitu tanpa APF sebagai pembanding dan dengan APF aktif, kemudian membandingkan kinerja keduanya.

4.7.1 Konfigurasi Pengujian Kolam

Pengujian dilakukan pada kolam renang Tirta Sagoro Brigif 2 Marinir, Gedangan, Sidoarjo, yang memiliki panjang total 50 meter dengan kedalaman bervariasi antara 1,5 hingga 2 meter. Dari keseluruhan area kolam tersebut, pengujian memanfaatkan sebagian area pada sisi berkedalaman 1,5 meter, dimana dibangun lintasan uji berukuran panjang 10,45 meter dan lebar 5 meter yang merepresentasikan konfigurasi arena KRBAI. Pemilihan area berkedalaman 1,5 meter didasarkan pada kesesuaian dengan kedalaman operasi misi AUV serta kemudahan pemasangan dan pengawasan objek lintasan selama pengujian. Kondisi pencahayaan alami di area kolam terbuka ini memberikan variasi intensitas cahaya yang ideal untuk menguji ketangguhan model deteksi objek YOLO terhadap fluktuasi iluminasi bawah air. Selain itu, tingkat kejernihan air pada area pengujian memungkinkan sistem kamera stereo untuk memetakan kedalaman rintangan secara optimal tanpa gangguan partikel tersuspensi (*turbidity*) yang berlebihan. Seluruh objek rintangan seperti gawang, suar, dan bak diletakkan secara presisi di dasar kolam menggunakan pemberat, dengan jarak antar rintangan yang diatur sedemikian rupa untuk memaksa algoritma Artificial Potential Field (APF) beroperasi secara maksimal. Lingkungan kolam yang cukup dinamis akibat pantulan riak ombak dari dinding turut dimanfaatkan untuk mengevaluasi

responsivitas sistem kendali PID dalam mempertahankan orientasi (*heading*) secara stabil. Melalui replikasi arena yang komprehensif ini, keseluruhan alur pengambilan keputusan wahana dapat diamati secara langsung guna memvalidasi keandalan integrasi sistem navigasi AUV secara utuh. Skema tata letak lintasan uji beserta *state* misi yang aktif pada setiap segmen ditunjukkan pada Gambar 4.33.



Gambar 4. 33 Skema Tata Letak Lintasan Uji dan *State* Misi pada Setiap Segme
(Sumber : Dokumentasi Pribadi)

Berdasarkan Gambar 4.33, objek lintasan ditempatkan berurutan menirukan arena KRBAI, yaitu gawang pertama, suar, gawang kedua, dan bak. Titik awal wahana berada pada jarak 4,25 meter dari gawang pertama, kemudian jarak dari gawang pertama ke suar adalah 2,4 meter, dari suar ke titik tengah gawang kedua 1,4 meter, dan dari gawang kedua ke bak 2,4 meter, sehingga total jarak yang ditempuh wahana dari titik awal hingga bak adalah 10,45 meter. Dimensi setiap objek yaitu gawang berukuran lebar 5 meter dengan tinggi 1,06 meter, suar berdiameter 14 sentimeter dengan tinggi 146 sentimeter, dan bak berdiameter 52 sentimeter dengan tinggi 24 sentimeter.

Bagian bawah Gambar 4.33 menunjukkan *state* misi yang aktif pada setiap segmen lintasan. Segmen pertama sepanjang 4,25 meter mencakup *state* SELAM, MAJU, dan MENUJU_GAWANG; segmen kedua sepanjang 2,4 meter mencakup *state* MENUJU_SUAR, SWAY, dan MAJU_2; segmen ketiga sepanjang 1,4 meter mencakup *state* MENUJU_GAWANG2 dan TURUN_BAK; sedangkan segmen

terakhir sepanjang 2,4 meter mencakup *state* SWAY_KANAN, MENUJU_BAK, NAIK, dan DIAM. Pemetaan ini diperlukan agar hubungan antara posisi fisik wahana dan *state* yang sedang dijalankan dapat ditelusuri ketika menganalisis rekaman *rosbag* hasil pengujian. Adapun kondisi nyata lintasan uji setelah seluruh objek terpasang pada kolam ditunjukkan pada Gambar 4.34.



Gambar 4. 34 Lintasan Uji yang Dibangun pada Kolam Tirto Sagoro (10,45 m × 5 m)

(Sumber: Dokumentasi Pribadi)

Gambar 4.34 memperlihatkan kondisi lintasan uji yang telah terpasang beserta keadaan air kolam saat pengujian berlangsung. Pengambilan foto dilakukan dari sisi atas permukaan air sehingga posisi dan jarak antar objek tidak dapat teramati dengan jelas akibat pembiasan cahaya pada permukaan air serta keterbatasan sudut pengambilan gambar. Oleh karena itu, skema pada Gambar 4.33 digunakan sebagai acuan tata letak dan jarak antar objek, sedangkan Gambar 4.34 berfungsi menunjukkan kondisi lingkungan pengujian yang sebenarnya, meliputi kejernihan air, pencahayaan, dan cara pemasangan objek pada lintasan.

Terdapat dua perbedaan utama antara konfigurasi kolam nyata dengan simulasi. Pertama, pembagian zona *depth* diperluas dari lima zona pada simulasi menjadi sembilan zona (grid 3×3) pada sistem nyata. Perluasan ini dilakukan karena kondisi nyata membutuhkan pemisahan vertikal, yaitu objek dapat muncul pada ketinggian berbeda seperti suar yang melayang dan bak yang berada di dasar kolam, sedangkan pada simulasi lingkungan lebih homogen. *Grid* 3×3 memungkinkan perintah geser dihitung per kolom (kiri, tengah, dan kanan) dengan mengambil jarak minimum tiap kolom, sekaligus tetap tahan terhadap gangguan lokal karena setiap zona dihitung dari persentil ke-25 *piksel* valid dengan ambang minimum 80 *piksel*. Kedua, sistem tidak menggunakan *odometri* maupun koordinat absolut; navigasi sepenuhnya berbasis persepsi, yaitu pelurusan terhadap posisi

horizontal objek pada citra (cx_norm) yang dikombinasikan dengan mesin keadaan misi. Parameter final yang digunakan pada seluruh pengujian dirangkum pada Tabel 4.22.

Tabel 4. 22 Parameter Final Sistem pada Pengujian Kolam Nyata

No.	Parameter	Nilai	Keterangan
1	K_REP	120	Konstanta perintah geser
2	D0	1,8 m	Radius pengaruh rintangan
3	Batas gaya sway	$\pm 250 \mu s$	Saturasi perintah geser lateral
4	d_min	0,30 m	Batas bawah pengukuran jarak
5	EMA α	0,4	Penghalusan pembacaan zona
6	<i>Deadband</i> perintah geser	60 μs	Ambang perintah geser
7	Pembagian zona depth	9 zona (3×3)	Persentil ke-25, min. 80 <i>piksel</i>

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.22, komponen atraktif tidak diimplementasikan sebagai konstanta atraktif eksplisit, melainkan sebagai kendali pelurusan proporsional terhadap cx_norm pada mesin keadaan misi. Seluruh data pengujian direkam otomatis oleh *mission_recorder_node* ke dalam *rosvag*, mencakup topik */mission/state*, */cmd/velocity*, */apf/debug*, */obstacle/zone_distances*, */obstacle/nearest_distance*, */perception/gate*, */perception/buoy*, */perception/bin*, */imu/yaw*, */imu/pitch*, */imu/roll*, */depth/data*, */pid/setpoint*, */thruster/pwm*, dan */serial/feedback*. Perekaman dimulai dan diakhiri mengikuti sinyal *kill switch* sehingga setiap *trial* menghasilkan satu berkas *bag* yang utuh.

4.7.2 Skenario Pengujian

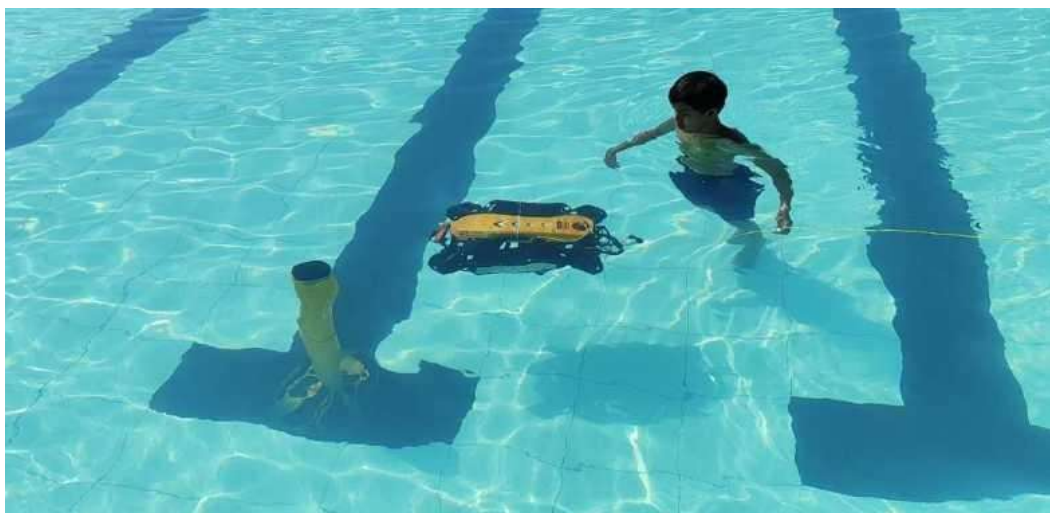
Pengujian tahap pertama menguji ketepatan arah perintah geser APF dengan menempatkan rintangan pada lintasan wahana dalam tiga kondisi, yaitu satu objek diam, dua objek diam, dan rintangan yang muncul mendadak. Besaran yang diamati adalah arah perintah geser, besar perintah geser, jarak terdekat wahana terhadap rintangan, serta faktor pengereman. Pengujian tahap kedua menguji keberhasilan misi secara utuh dalam dua skenario pada lintasan dan kondisi kolam yang sama.

Skenario pertama menonaktifkan perintah geser (*apf_enable = false*) sehingga navigasi hanya mengandalkan mesin keadaan dan pelurusan *cx_norm*; skenario ini menjadi pembanding untuk mengukur kontribusi APF secara terisolasi. Skenario kedua menjalankan sistem lengkap dengan perintah geser dihitung dari zona *stereo depth* dan dijumlahkan dengan perintah *sway* misi.

Kriteria keberhasilan adalah menyelesaikan seluruh *state* misi tanpa menabrak objek rintangan. *Trial* yang harus dihentikan operator melalui *kill switch* karena wahana menuju tabrakan dinyatakan gagal. Pengujian tahap kedua dilaksanakan dalam satu sesi dengan lintasan dan kondisi identik, masing-masing skenario diulang sepuluh kali sehingga diperoleh total dua puluh *trial*. Urutan *state* misi yang harus diselesaikan adalah SELAM → MAJU → MENUJU_GATE → MENUJU_SUAR → SWAY → MAJU_2 → MENUJU_GATE2 → TURUN_BAK → SWAY_KANAN → MENUJU_BAK → NAIK → DIAM.

4.7.3 Pengujian Respons Perintah Geser terhadap Satu Objek Diam

Pengujian ini bertujuan mengetahui arah dan besar perintah geser yang dihasilkan algoritma terhadap satu rintangan diam pada berbagai jarak. Satu buah suar ditempatkan pada lintasan wahana, kemudian wahana didekatkan secara bertahap sambil merekam keluaran perintah geser pada setiap jarak. Posisi suar divariasikan pada sisi kanan dan sisi kiri lintasan agar dapat diketahui apakah arah perintah geser mengikuti sisi tempat rintangan berada. Proses pengujian ditunjukkan pada Gambar 4.35.



Gambar 4. 35 Pengujian Respons Perintah Geser terhadap Satu Objek Diam

(Sumber : Dokumentasi Pribadi)

Gambar 4.35 memperlihatkan penempatan suar dalam keadaan diam pada lintasan serta posisi wahana saat mendekatinya. Kondisi rintangan tunggal dipilih sebagai pengujian awal karena hanya terdapat satu sumber perintah geser, sehingga hubungan antara jarak rintangan dan besar keluaran dapat diamati tanpa pengaruh rintangan lain. Hasil pengujian ditunjukkan pada Tabel 4.23.

Tabel 4. 23 Hasil Pengujian Respons Perintah Geser terhadap Satu Objek Diam

No.	Posisi Objek	Arah Sway	Jarak (m)	PWM (μs)	Keterangan
1	Kanan	-	1,2	0	Di bawah ambang <i>deadband</i>
2	Kanan	-	1,0	0	Di bawah ambang <i>deadband</i>
3	Kanan	Kiri	0,9	66	Ambang <i>deadband</i> terlampaui
4	Kanan	Kiri	0,7	85	Pergeseran menguat seiring jarak berkurang
5	Kanan	Kiri	0,5	123	Pergeseran kuat
6	Kanan	Kiri	0,35	213	Mendekati batas saturasi
7	Kanan	Kiri	0,3	250	Sway maksimum
8	Kiri	-	1,2	0	Di bawah ambang <i>deadband</i>
9	Kiri	-	1,0	0	Di bawah ambang <i>deadband</i>
10	Kiri	Kanan	0,9	68	Ambang <i>deadband</i> terlampaui
11	Kiri	Kanan	0,7	88	Pergeseran menguat seiring jarak berkurang
12	Kiri	Kanan	0,5	124	Pergeseran kuat
13	Kiri	Kanan	0,35	217	Mendekati batas saturasi
14	Kiri	Kanan	0,3	250	<i>Sway</i> maksimum

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.23, arah perintah geser selalu berlawanan dengan sisi tempat rintangan berada. Ketika suar berada di sisi kanan, perintah geser mengarah ke kiri, dan ketika suar dipindahkan ke sisi kiri, arah perintah geser berbalik ke kanan. Hasil ini membuktikan bahwa arah penghindaran dihitung dari posisi rintangan yang terdeteksi dan tidak ditetapkan pada satu sisi tertentu.

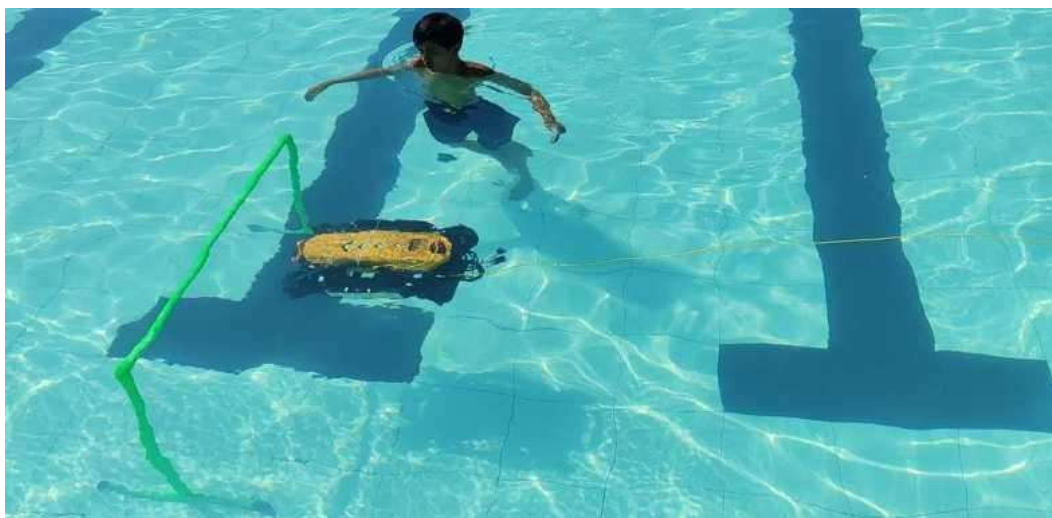
Besar keluaran meningkat seiring berkurangnya jarak rintangan. Pada jarak 1,2 dan 1,0 meter keluaran bernilai nol karena hasil perhitungan masih berada di

bawah ambang *deadband* sebesar 60 μ s, sehingga wahana tetap bergerak lurus. Keluaran mulai muncul pada jarak 0,9 meter dengan nilai 66 μ s pada kondisi objek di kanan dan 68 μ s pada kondisi objek di kiri, yaitu tepat di atas ambang *deadband*. Keluaran terus meningkat hingga mencapai batas saturasi 250 μ s pada jarak 0,3 meter. Pola ini sesuai dengan karakteristik perintah geser yang berbanding terbalik terhadap jarak, sekaligus menunjukkan bahwa mekanisme *deadband* bekerja sesuai perancangan dengan menahan reaksi hingga rintangan benar-benar mendekat.

Keluaran pada kedua sisi menunjukkan kesetaraan yang tinggi. Pada jarak yang sama, selisih keluaran antara kondisi objek di kanan dan objek di kiri hanya berkisar 0 hingga 4 μ s dengan rata-rata 2,0 μ s, yaitu di bawah 2 persen terhadap nilai keluarannya. Kesetaraan ini menunjukkan bahwa perhitungan perintah geser bersifat simetris terhadap kedua sisi, sehingga wahana memiliki kemampuan menghindar yang sama baiknya ke kiri maupun ke kanan.

4.7.4 Pengujian Respons Perintah Geser terhadap Dua Objek Diam

Pengujian dengan dua objek diam bertujuan memverifikasi bahwa algoritma mampu menjumlahkan kontribusi dari lebih dari satu rintangan menjadi satu perintah geser resultan. Rintangan yang digunakan adalah gawang, dimana tiang kiri dan tiang kanan berperan sebagai dua objek terpisah yang menghasilkan perintah geser dari sisi berlawanan. Konfigurasi ini dipilih karena merepresentasikan kondisi nyata pada misi, yaitu melintasi celah di antara dua rintangan. Proses pengujian ditunjukkan pada Gambar 4.36.



Gambar 4. 36 Pengujian Respons Perintah Geser terhadap Dua Objek Diam

(Sumber : Dokumentasi Pribadi)

Gambar 4.36 memperlihatkan gawang yang digunakan sebagai dua rintangan beserta posisi wahana saat memasuki celah di antara kedua tiang. Karena kedua tiang berada pada sisi berlawanan, perintah geser yang dihasilkan keduanya juga saling berlawanan arah, sehingga arah resultan ditentukan oleh tiang yang berada lebih dekat terhadap wahana. Hasil pengujian ditunjukkan pada Tabel 4.24.

Tabel 4. 24 Hasil Pengujian Respons Perintah Geser terhadap Dua Objek Diam

No.	Posisi Objek Terdekat	Arah Sway	Jarak (m)	PWM (μs)	Faktor Pengereman Min.
1	Kanan	-	1,4	0	Di bawah ambang <i>deadband</i>
2	Kanan	-	1,2	0	Di bawah ambang <i>deadband</i>
3	Kanan	Kiri	1,0	0	Di bawah ambang <i>deadband</i>
4	Kanan	Kiri	0,9	65	Ambang <i>deadband</i> terlampaui
5	Kanan	Kiri	0,7	78	Pergeseran menguat seiring jarak berkurang
6	Kanan	Kiri	0,5	114	Pergeseran kuat
7	Kanan	Kiri	0,35	216	Mendekati batas saturasi
8	Kanan	Kiri	0,3	250	Sway maksimum
9	Kiri	-	1,4	0	Di bawah ambang <i>deadband</i>
10	Kiri	-	1,2	0	Di bawah ambang <i>deadband</i>
11	Kiri	Kanan	1,0	0	Di bawah ambang <i>deadband</i>
12	Kiri	Kanan	0,9	66	Ambang <i>deadband</i> terlampaui
13	Kiri	Kanan	0,7	80	Pergeseran menguat seiring jarak berkurang
14	Kiri	Kanan	0,5	120	Pergeseran kuat
15	Kiri	Kanan	0,35	220	Mendekati batas saturasi
16	Kiri	Kanan	0,3	250	Sway maksimum

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.24, arah perintah geser selalu menjauhi tiang yang berada lebih dekat terhadap wahana. Ketika wahana memasuki celah dengan posisi lebih dekat ke tiang kanan, perintah geser mengarah ke kiri, dan sebaliknya. Hasil ini menunjukkan bahwa penjumlahan kontribusi kedua tiang berjalan sesuai perancangan, dimana tiang yang lebih dekat memberikan kontribusi lebih besar sehingga mendominasi arah resultan. Karakteristik tersebut merupakan konsekuensi sifat perintah geser yang berbanding terbalik terhadap jarak.

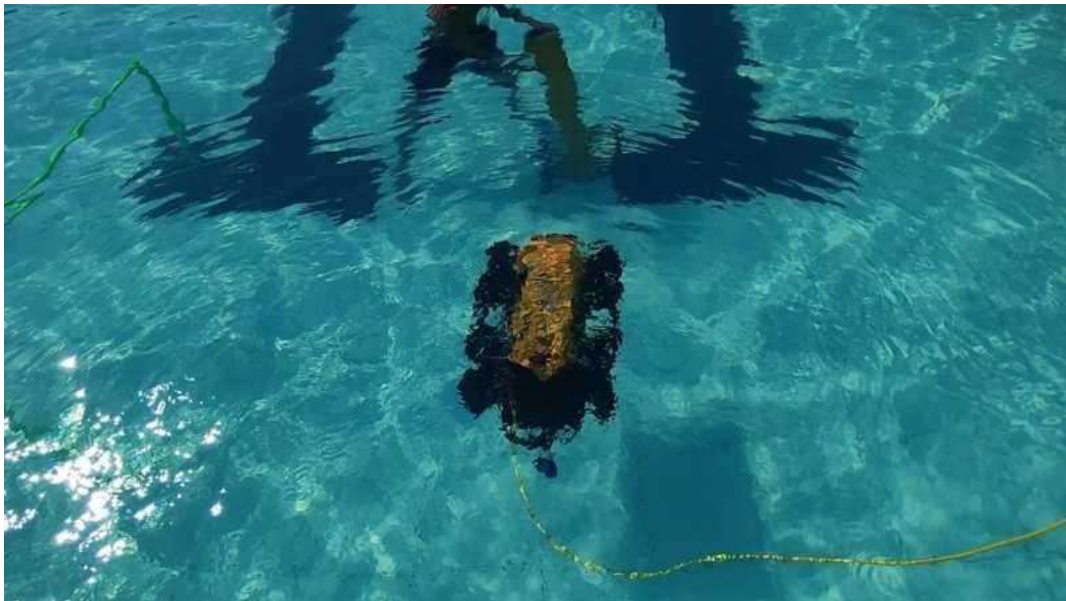
Perilaku tersebut menghasilkan efek pemusatan wahana pada celah gawang. Ketika wahana bergeser mendekati salah satu tiang, perintah geser dari tiang tersebut membesar sehingga wahana didorong kembali ke arah tengah celah, dan hal yang sama berlaku pada sisi sebaliknya. Kesetaraan keluaran pada kedua sisi memperkuat hal ini, yaitu selisih keluaran pada jarak yang sama hanya berkisar 0 hingga 6 μs dengan rata-rata 2,6 μs . Dengan demikian kedua tiang gawang secara bersama-sama membentuk lintasan aman di tengah celah tanpa memerlukan titik acuan koordinat. Mekanisme inilah yang menjadi dasar keberhasilan wahana melintasi kedua gawang pada pengujian misi dengan APF aktif, sekaligus menjelaskan mengapa kegagalan pada pengujian tanpa APF terjadi saat melintasi celah gawang, sebagaimana dibahas pada subbab 4.7.6 dan 4.7.7.

Perlu dicatat bahwa keluaran pada kondisi dua objek mulai muncul pada jarak yang sama dengan kondisi satu objek, yaitu 0,9 meter, namun dengan nilai yang sedikit lebih kecil pada jarak 0,5 dan 0,7 meter. Hal ini terjadi karena tiang pada sisi berlawanan turut menghasilkan perintah geser yang saling mengurangi, sehingga resultannya lebih kecil dibandingkan kondisi rintangan tunggal.

4.7.5 Pengujian Respons Perintah Geser terhadap Rintangan yang Muncul Mendadak

Pengujian ini bertujuan memverifikasi kemampuan algoritma dalam merespons rintangan yang tidak berada pada lintasan ketika misi dimulai. Pengujian dilaksanakan dengan bantuan seorang operator pembantu. Wahana dijalankan maju secara otonom, kemudian ketika wahana telah bergerak mendekat, suar diletakkan secara mendadak di jalur wahana pada sisi kiri. Dengan demikian wahana tidak memiliki informasi awal mengenai keberadaan rintangan dan harus bereaksi

sepenuhnya berdasarkan pembacaan sesaat dari kamera stereo. Proses pengujian ditunjukkan pada Gambar 4.37.



Gambar 4. 37 Pengujian Respons Perintah Geser terhadap Rintangan yang Muncul Mendadak

(Sumber : Dokumentasi Pribadi)

Gambar 4.37 memperlihatkan proses peletakan suar di jalur wahana oleh operator pembantu. Kondisi ini merupakan pengujian yang paling mendekati sifat perencanaan lintasan lokal, yaitu pengambilan keputusan gerak berdasarkan kondisi lingkungan terkini tanpa perencanaan lintasan global maupun peta lingkungan yang disiapkan sebelumnya. Hasil pengujian ditunjukkan pada Tabel 4.25.

Tabel 4. 25 Hasil Pengujian Arah Respons Perintah Geser terhadap Rintangan yang Muncul Mendadak

No.	Posisi Objek	Gerak Sway	Jarak (m)	PWM (μs)	Jarak Terdekat (m)
1	Objek bergerak	Kiri	1,4	0	Di bawah ambang <i>deadband</i>
2	Objek kergerak	Kiri	1,2	0	Di bawah ambang <i>deadband</i>
3	Objek bergerak	Kiri	1,0	0	Di bawah ambang <i>deadband</i>
4	Objek bergerak	Kiri	0,9	68	Ambang <i>deadband</i> terlampaui

No.	Posisi Objek	Gerak Sway	Jarak (m)	Keluaran Sway (μs)	Jarak Terdekat (m)
5	Objek bergerak	Kiri	0,7	76	Pergeseran menguat seiring jarak berkurang
6	Objek bergerak	Kiri	0,5	130	Pergeseran kuat
7	Objek bergerak	Kiri	0,35	168	Mendekati batas saturasi
8	Objek bergerak	Kiri	0,3	250	Sway maksimum

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.25, wahana menghasilkan perintah geser ke arah kanan pada seluruh pengujian ketika suar diletakkan pada sisi kiri jalur. Pola peningkatan keluaran terhadap berkurangnya jarak juga serupa dengan kedua pengujian sebelumnya, yaitu bernilai nol hingga jarak 1,0 meter karena masih di bawah ambang *deadband*, mulai muncul pada jarak 0,9 meter dengan nilai 68 μs , dan mencapai saturasi 250 μs pada jarak 0,3 meter. Pola peningkatan keluaran terhadap berkurangnya jarak juga serupa dengan kedua pengujian sebelumnya, yaitu bernilai nol hingga jarak 1,0 meter, mulai muncul pada jarak 0,9 meter, dan mencapai saturasi 250 μs pada jarak 0,3 meter. Kesamaan pola ini menunjukkan bahwa algoritma memperlakukan rintangan yang muncul mendadak dengan cara yang sama seperti rintangan yang sudah ada sejak awal, sehingga tidak diperlukan penanganan khusus untuk rintangan tak terduga.

Hasil ini membuktikan bahwa algoritma bersifat reaktif dan tidak bergantung pada pengetahuan awal mengenai posisi rintangan, sesuai karakteristik perencanaan lintasan lokal yang dibahas pada Bab 2. Secara keseluruhan, dari ketiga kondisi pengujian pada Tabel 4.22 hingga Tabel 4.24, algoritma menghasilkan arah perintah geser yang benar pada seluruh titik pengukuran dengan keluaran nol ketika rintangan berada di luar jarak reaksi efektif, sehingga tidak terjadi manuver yang tidak diperlukan.

4.7.6 Hasil Pengujian Misi dengan APF Nonaktif

Pengujian dengan APF nonaktif dilakukan sebagai pembandingan untuk mengetahui kinerja wahana ketika navigasi hanya mengandalkan mesin keadaan dan pelurusan *cx_norm* tanpa perintah geser. Hasil sepuluh *trial* ditunjukkan pada Tabel 4.26.

Tabel 4. 26 Hasil Pengujian Misi dengan APF Nonaktif

No.	Durasi (s)	Status	State Terakhir	Keterangan
1	41,8	Selesai	DIAM	Misi lengkap
2	44,7	Selesai	DIAM	Misi lengkap
3	26,9	Gagal	MENUJU_GATE2	Menabrak tiang gawang
4	42,4	Selesai	DIAM	Misi lengkap
5	34,0	Gagal	SWAY_KANAN	Menabrak tiang gawang
6	42,7	Selesai	DIAM	Misi lengkap
7	47,8	Selesai	DIAM	Misi lengkap
8	47,4	Selesai	DIAM	Misi lengkap
9	37,6	Gagal	SWAY_KANAN	Menabrak tiang gawang
10	44,5	Selesai	DIAM	Misi lengkap
Tingkat keberhasilan	7 dari 10 percobaan (70 %)			
Durasi rata-rata	44,5 s			

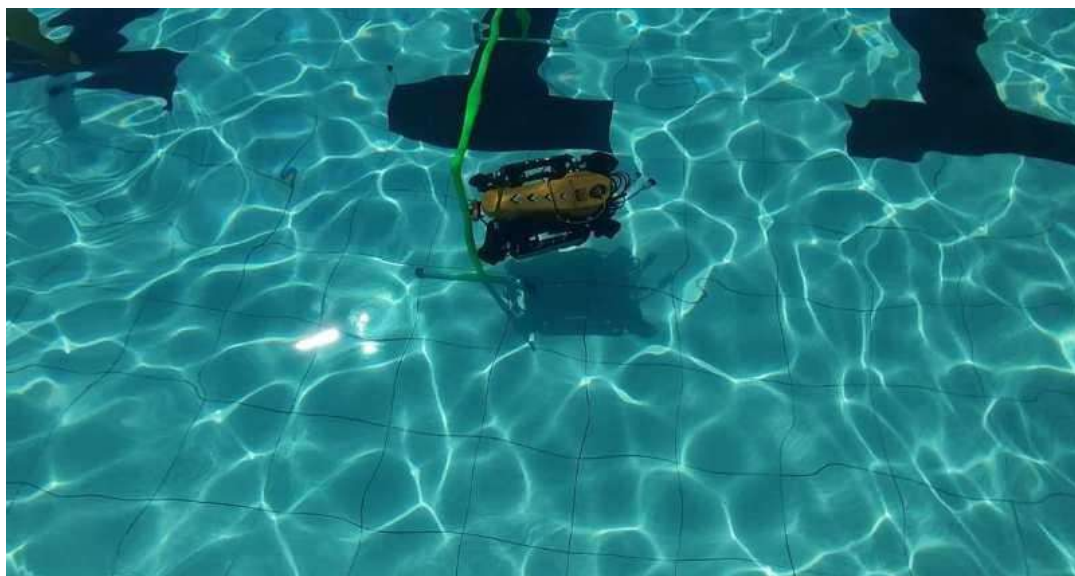
(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.26, tujuh dari sepuluh *trial* berhasil menyelesaikan seluruh urutan *state* misi dengan durasi rata-rata 44,5 detik. Tiga *trial* dinyatakan gagal karena wahana menabrak tiang gawang sehingga harus dihentikan operator melalui kill switch, yaitu *trial* ke-3, ke-5, dan ke-9 dengan durasi masing-masing 26,9 detik, 34,0 detik, dan 37,6 detik. Durasi ketiga *trial* gagal tersebut lebih singkat karena pengujian dihentikan sebelum misi selesai.

State terakhir yang terekam pada ketiga *trial* gagal adalah MENUJU_GATE2 pada satu *trial* serta SWAY_KANAN pada dua *trial*, sehingga seluruh kegagalan terjadi di sekitar gawang kedua. Perlu dicatat bahwa *state* terakhir yang terekam lebih maju daripada lokasi fisik tabrakan. Hal ini terjadi karena mesin keadaan bertransisi berdasarkan pewaktu dan status deteksi, bukan berdasarkan konfirmasi posisi, sehingga *state* tetap berjalan meskipun wahana telah tertahan oleh gawang.

Penyebab kegagalan adalah tidak adanya koreksi terhadap penyimpangan lateral wahana. Tanpa perintah geser, penyimpangan posisi wahana terhadap sumbu

lintasan tidak terkoreksi sepanjang fase melaju, sehingga wahana mendekati celah gawang dari arah yang tidak terpusat dan menabrak tiang. Kegagalan terkonsentrasi pada gawang kedua karena gawang tersebut berada setelah manuver penghindaran suar, sehingga penyimpangan lateral yang terakumulasi dari manuver sebelumnya belum terkoreksi ketika wahana memasuki celah gawang kedua. Salah satu kondisi kegagalan tersebut ditunjukkan pada Gambar 4.38.



Gambar 4. 38 Posisi Wahana saat Menabrak Tiang Gawang pada Pengujian APF Nonaktif

(Sumber : Dokumentasi Pribadi)

Gambar 4.38 memperlihatkan posisi wahana ketika mengalami tabrakan, Dimana wahana mendekati celah gawang dengan posisi yang tidak terpusat sehingga salah satu sisi lambungnya tertahan oleh tiang gawang. Kondisi tersebut menunjukkan penyebab kegagalan, yaitu tidak adanya koreksi terhadap penyimpangan lateral wahana. Tanpa perintah geser, penyimpangan posisi wahana terhadap sumbu lintasan tidak terkoreksi sepanjang fase melaju, sehingga wahana memasuki celah gawang dari arah yang menyudut dan menabrak tiang. Kegagalan terkonsentrasi pada gawang kedua karena gawang tersebut berada setelah manuver penghindaran suar, sehingga penyimpangan lateral yang terakumulasi dari manuver sebelumnya belum terkoreksi ketika wahana memasuki celah gawang kedua.

4.7.7 Hasil Pengujian Misi dengan APF Aktif

Pengujian dengan APF aktif dilakukan pada lintasan dan kondisi kolam yang identik dengan pengujian sebelumnya, dengan perintah geser dihitung dari

zona *stereo depth* dan dijumlahkan terhadap perintah sway misi. Hasil sepuluh *trial* ditunjukkan pada Tabel 4.27.

Tabel 4. 27 Hasil Pengujian Misi dengan APF Aktif

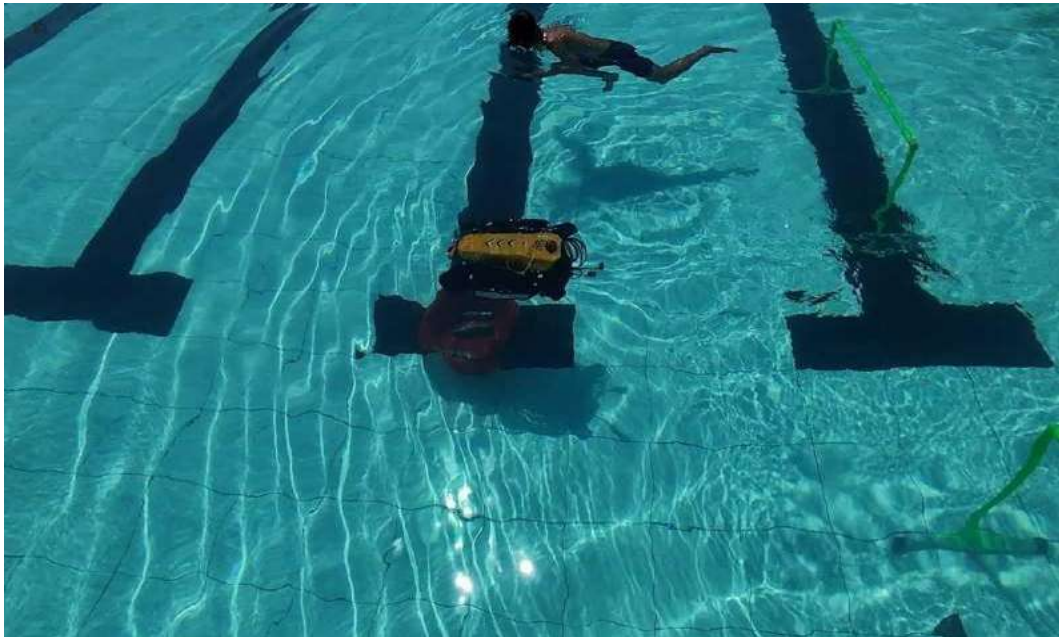
No.	Durasi (s)	Status	State Terakhir	Keterangan
1	46,6	Selesai	DIAM	Misi lengkap
2	45,5	Selesai	DIAM	Misi lengkap
3	42,8	Selesai	DIAM	Misi lengkap
4	42,3	Selesai	DIAM	Misi lengkap
5	50,9	Selesai	DIAM	Misi lengkap
6	43,8	Selesai	DIAM	Misi lengkap
7	42,9	Selesai	DIAM	Misi lengkap
8	43,8	Selesai	DIAM	Misi lengkap
9	40,9	Selesai	DIAM	Misi lengkap
10	44,9	Selesai	DIAM	Misi lengkap
Tingkat keberhasilan	10 dari 10 percobaan (100 %)			
Durasi rata - rata	44,4 s			

(Sumber : Dokumentasi Pribadi)

Berdasarkan Tabel 4.27, seluruh sepuluh *trial* berhasil menyelesaikan urutan *state* misi lengkap hingga *state* DIAM tanpa satu pun tumbukan, dengan durasi rata-rata 44,4 detik. Tidak terdapat *trial* yang harus dihentikan operator. Keberhasilan pada seluruh *trial* menunjukkan bahwa perintah geser yang dihasilkan dari pembacaan zona *stereo depth* mampu menjaga wahana tetap terpusat pada celah gawang. Ketika salah satu tiang gawang terbaca dekat pada sisi tertentu, perintah geser menghasilkan dorongan ke arah berlawanan sehingga penyimpangan lateral wahana terkoreksi sebelum wahana memasuki celah gawang.

Durasi terpanjang sebesar 50,9 detik terjadi pada *trial* kelima, sedangkan durasi terpendek sebesar 40,9 detik terjadi pada *trial* kesembilan. Selisih tersebut disebabkan perbedaan besarnya manuver penghindaran yang dilakukan wahana pada setiap *trial*, dimana penyimpangan awal yang lebih besar menuntut manuver menyamping yang lebih panjang beserta pengereman yang lebih kuat sehingga waktu tempuh bertambah. Variasi durasi antar *trial* tersebut tetap berada pada

rentang yang wajar dan tidak menunjukkan adanya perilaku menyimpang pada satu *trial* tertentu. Kondisi wahana pada akhir misi ditunjukkan pada Gambar 4.v.



Gambar 4. 39 Kondisi Wahana saat Misi Selesai pada Pengujian APF Aktif

(Sumber : Dokumentasi Pribadi)

Gambar 4.39 memperlihatkan wahana yang telah muncul kembali ke permukaan air setelah menyelesaikan state NAIK, yang menandakan seluruh urutan state misi terselesaikan hingga state DIAM. Kondisi ini tercapai pada seluruh sepuluh *trial* tanpa intervensi operator. Durasi terpanjang sebesar 50,9 detik terjadi pada *trial* kelima, sedangkan durasi terpendek sebesar 40,9 detik terjadi pada *trial* kesembilan. Selisih tersebut disebabkan perbedaan besarnya manuver penghindaran pada setiap *trial*, dimana penyimpangan awal yang lebih besar menuntut manuver menyamping yang lebih panjang beserta pengereman yang lebih kuat sehingga waktu tempuh bertambah. Variasi durasi antar *trial* tersebut tetap berada pada rentang yang wajar dan tidak menunjukkan adanya perilaku menyimpang pada satu *trial* tertentu.

4.7.8 Perbandingan Kinerja APF Aktif dan Nonaktif

Perbandingan kinerja kedua skenario dirangkum pada Tabel 4.27. Kedua skenario dijalankan pada lintasan dan kondisi kolam yang identik sehingga perbedaan hasil yang diperoleh murni disebabkan oleh diaktifkan atau tidaknya perintah geser APF.

Tabel 4. 28 Rangkuman perbandingan APF aktif dan nonaktif

No.	Metrik	APF Nonaktif	APF Aktif	Keterangan
1	Tingkat keberhasilan misi	7/10 (70%)	10/10 (100%)	Metrik utama
2	Jumlah tumbukan	3	0	Seluruhnya pada gawang kedua
3	Durasi rata - rata	44,5 s	44,4 s	Selesai 0,1 % tanpa penambahan waktu tempuh

(Sumber : Dokumentasi Pribadi)

Metrik utama pada Tabel 4.28 adalah keberhasilan menyelesaikan misi. Pada skenario tanpa perintah geser APF, wahana mengalami tumbukan dengan tiang gawang pada tiga dari sepuluh percobaan sehingga hanya tujuh percobaan yang berhasil menyelesaikan seluruh urutan *state* misi. Pada skenario dengan perintah geser APF aktif, seluruh sepuluh percobaan berhasil menyelesaikan misi tanpa satu pun tumbukan. Dengan demikian, pengaktifan perintah geser APF menghilangkan ketiga kegagalan yang terjadi pada skenario pembandingan. Ketiga kegagalan tersebut berupa tumbukan dengan tiang gawang akibat penyimpangan lateral wahana yang tidak terkoreksi, dan perintah geser APF bekerja tepat pada penyebab tersebut dengan mendorong wahana menjauhi tiang yang terbaca dekat sehingga wahana tetap terpusat pada celah gawang.

Perbaikan tersebut diperoleh tanpa penambahan waktu tempuh misi. Durasi rata-rata percobaan yang berhasil pada skenario tanpa APF adalah 44,5 detik, sedangkan pada skenario dengan APF aktif adalah 44,4 detik, sehingga selisihnya hanya 0,1 persen. Hal ini menunjukkan bahwa manuver penghindaran yang dilakukan wahana berlangsung singkat dan tidak membuat wahana berhenti atau berputar berulang kali, sehingga penambahan kemampuan menghindar tidak mengorbankan efisiensi waktu misi.

Berdasarkan kedua metrik tersebut, sistem *obstacle avoidance* berbasis *Stereo Depth* dan Artificial Potential Field terbukti mampu menghilangkan seluruh tumbukan yang terjadi pada skenario tanpa APF, sehingga wahana dapat menyelesaikan misi pada seluruh percobaan tanpa menambah waktu tempuh.

(Halaman ini sengaja dikosongkan)

BAB 5

KESIMPULAN

5.1 Kesimpulan

Berdasarkan perancangan, implementasi, dan pengujian sistem obstacle avoidance pada Autonomous Underwater Vehicle (AUV) berbasis deteksi objek YOLO, informasi kedalaman kamera stereo, dan perencanaan gerak Artificial Potential Field (APF) yang diintegrasikan dalam ROS2, diperoleh kesimpulan sebagai berikut:

1. Deteksi YOLO dan informasi kedalaman kamera stereo diimplementasikan sebagai dua simpul ROS2 terpisah yang digabungkan pada tahap akhir. Simpul kedalaman menghitung peta disparitas dari pasangan citra terektifikasi menggunakan algoritma *Semi-Global Block Matching*, lalu mengubahnya menjadi jarak melalui hubungan triangulasi dengan panjang fokus 350,99 *piksel* dan jarak antar kamera 98,51 mm. Simpul YOLO menjalankan inferensi pada citra kiri dan menghasilkan kotak pembatas objek. Penggabungan dilakukan dengan mengambil nilai kedalaman hanya di dalam kotak pembatas, sehingga pembacaan jarak tidak tercemar *piksel* latar yang banyak mengandung derau. Nilai jarak tersebut kemudian dikoreksi terhadap refraksi air menggunakan fungsi linear yang diperoleh dari regresi data terendam. Melalui mekanisme ini, model mengenali ketiga objek lintasan dengan mAP@50 sebesar 0,990 dan menghasilkan galat estimasi jarak rata-rata di bawah 5% pada rentang kerja 0,4–2,8 m, yaitu 4,66% pada suar dan 4,51% pada bak. Rasio faktor skala koreksi air terhadap darat sebesar 1,362 mendekati indeks bias air tawar ($n \approx 1,333$), sehingga penyimpangan estimasi jarak terbukti berasal dari pembiasan cahaya dan bukan dari kesalahan kalibrasi. Objek gawang dikecualikan dari acuan jarak karena bentuknya yang berongga membuat sebagian besar kotak pembatas terisi air, sehingga navigasinya mengandalkan simpangan pusat kotak pembatas terhadap pusat citra.
2. *Local planner* berbasis APF diterapkan tanpa memerlukan peta global maupun koordinat absolut. Peta kedalaman dibagi menjadi sembilan zona berbentuk kisi tiga kali tiga; setiap zona menghasilkan gaya tolak semu yang besarnya berbanding terbalik terhadap jarak rintangan pada zona tersebut, aktif hanya

ketika jarak berada di bawah ambang 1,0 meter. Resultan gaya tolak dari kesembilan zona dijumlahkan dengan gaya tarik semu berupa pelurusan visual terhadap objek target, lalu diubah menjadi perintah geser lateral dalam satuan mikrodetik PWM, dihaluskan dengan penapis rata-rata bergerak eksponensial agar keluaran tidak berosilasi. Perintah tersebut dieksekusi melalui tujuh mode gerak dasar wahana dengan kecepatan maju 0,455 m/s, mundur 0,417 m/s, menyamping 0,369 m/s, naik 0,261 m/s, turun 0,229 m/s, dan berputar 60,0 °/s; perbedaan kecepatan maju dan menyamping sesuai rasio luas penampang lambung dengan selisih 2,1 % terhadap perhitungan, sehingga merupakan konsekuensi geometri dan bukan ketidakseimbangan *thruster*. Pengujian terhadap satu rintangan diam, celah gawang, dan rintangan yang muncul mendadak menunjukkan arah perintah geser selalu menjauhi sisi tempat rintangan terdeteksi, dengan keluaran bernilai nol pada jarak di atas 1,0 meter dan mencapai saturasi 250 μ s pada jarak 0,3 meter. Selisih keluaran sisi kiri dan kanan rata-rata hanya 2,0 μ s pada rintangan tunggal dan 3,2 μ s pada celah gawang, sehingga kemampuan menghindar setara ke kedua arah.

3. Kinerja sistem diuji pada lintasan yang menyerupai arena KRBAI dengan membandingkan dua konfigurasi pada kondisi identik. Sistem tanpa APF mengalami tumbukan dengan tiang gawang pada tiga dari sepuluh percobaan, sedangkan sistem dengan APF aktif menyelesaikan seluruh sepuluh percobaan tanpa satu pun tumbukan, sehingga tingkat keberhasilan navigasi meningkat dari 70% menjadi 100%. Peningkatan tersebut diperoleh tanpa penambahan waktu tempuh, yaitu durasi rata-rata 44,4 detik dengan APF dibandingkan 44,5 detik tanpa APF. Hal ini menunjukkan bahwa gaya tolak semu bekerja sebagai koreksi lateral bertahap yang menjaga jarak aman terhadap rintangan tanpa membelokkan wahana secara berlebihan dari lintasan menuju target, sehingga keandalan navigasi meningkat sekaligus efisiensi lintasan tetap terjaga.

5.2 Saran

Berdasarkan temuan dan keterbatasan selama penelitian, beberapa saran untuk pengembangan lanjutan adalah sebagai berikut:

1. Estimasi kedalaman berbasis stereo pasif kurang andal pada permukaan bertekstur rendah maupun pada objek berongga seperti gawang, karena

algoritma pencocokan disparitas kehilangan acuan tekstur. Penelitian lanjutan dapat memperbaiki hal ini melalui strategi pengambilan sampel kedalaman yang lebih selektif, penambahan sensor jarak pelengkap seperti sonar, atau penggunaan metode estimasi kedalaman berbasis pembelajaran mesin.

2. Kalibrasi koreksi jarak bawah air pada penelitian ini divalidasi pada rentang 0,4–2,0 m, sehingga estimasi jarak cenderung underestimate ketika objek berada di luar rentang tersebut. Pengembangan selanjutnya dapat memperluas rentang dan menambah titik data kalibrasi agar akurasi jarak tetap terjaga pada jangkauan deteksi yang lebih jauh.
3. Gaya tolak semu APF berbasis kedalaman saat ini belum dapat membedakan rintangan yang harus dihindari dari objek yang sebenarnya tidak menghalangi (misalnya tali pemberat di dekat bak). Pengembangan selanjutnya dapat memadukan informasi kelas objek dari YOLO ke dalam perhitungan gaya tolak semu, sehingga sistem mampu mengabaikan objek yang bukan rintangan secara lebih cerdas.
4. Penghindaran rintangan saat ini aktif pada fase melaju, namun belum aktif pada fase manuver menyamping (sway) yang berdurasi tetap. Penelitian lanjutan dapat mengembangkan mekanisme penghindaran yang tetap aktif selama manuver menyamping berlangsung, misalnya dengan memantau ulang peta kedalaman di tengah manuver sehingga durasi geser dapat dipotong atau diperpanjang secara adaptif sesuai kondisi rintangan terkini.

(Halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA

- Adi Rahmad Ramadhan, Khumaidi, A., Mustika Kurnia Mayangsari, Mat Syai'in, Imam Sutrisno, & Aulia Rahma Annisa. (2025). Penerapan Harris Corner Detection dan YOLOv5 pada Kamera Stereo Vision untuk Estimasi Jarak Robot Sepak Bola Beroda KRSBI-B. *Jurnal Elektronika Dan Otomasi Industri*, 12(1), 111–122. <https://doi.org/10.33795/elkolind.v12i1.7254>
- Adryady, D. F., Prasetyo, A., Shatyaziamawan, H. P., & Priambodo, A. S. (2022). Implementasi Algoritma Path Planning dan Mapping Arena pada Mobile Robot. *Jurnal Teknik Elektro Dan Komputer TRIAC*, 9(2), 41–45. <https://journal.trunojoyo.ac.id/triac/article/view/13215>
- Al-Batati, A. S., Koubaa, A., & Abdelkader, M. (2024). *ROS 2 Key Challenges and Advances: A Survey of ROS 2 Research, Libraries, and Applications*. <https://doi.org/10.20944/preprints202410.1204.v1>
- Amini, S., Informatika, T., Informasi, F. T., Luhur, U. B., Utara, P., Lama, K., Switch, M. D., Ruangan, K., Keamanan, P., Dengan, R., Pir, S., Magnetic, D., Switch, D., & Web, B. (2021). Perancangan Keamanan Ruangan Dengan Sensor Pir. *Juli*, 4(2), 50–56.
- Chen, C., Liu, T., He, L., Zhu, Y., & Yuan, F. (2025). Real-Time Underwater Vision Sensing System for AUV Tracking. *IET Image Processing*, 19(1). <https://doi.org/10.1049/ipr2.70183>
- Chen, R., & Chen, Y. (2023). Improved Convolutional Neural Network YOLOv5 for Underwater Target Detection Based on Autonomous Underwater Helicopter. *Journal of Marine Science and Engineering*, 11(5), 989. <https://doi.org/10.3390/jmse11050989>
- Cong, Y., Gu, C., Zhang, T., & Gao, Y. (2021). Underwater robot sensing technology: A survey. *Fundamental Research*, 1(3), 337–345. <https://doi.org/10.1016/j.fmre.2021.03.002>
- Dai, Z., & Jing, L. (2021). Lightweight extended kalman filter for marg sensors attitude estimation. *IEEE Sensors Journal*, 21(13), 14749–14758. <https://doi.org/10.1109/JSEN.2021.3072887>
- Dijaya, R., & Setiawan, H. (2023). *BUKU AJAR PENGOLAHAN CITRA DIGITAL*.

- UMSIDA PRESS. <https://doi.org/https://doi.org/10.21070/2023/978-623-464-075-5>
- Exploration, D. (2023). *DWE exploreHD*. DeepWater Exploration. <https://dwe.ai/products/explorehd>
- Fan, X., Guo, Y., Liu, H., Wei, B., & Lyu, W. (2020). Improved Artificial Potential Field Method Applied for AUV Path Planning. *Mathematical Problems in Engineering*, 2020. <https://doi.org/10.1155/2020/6523158>
- Firmansyah, F. I., Zuhrie, M. S., Rohman, M., Endryansyah, E., & Aulia, M. (2023). Sistem Obstacle Avoidance Pada Omnidirectional Mobile Robot Dengan Metode Artificial Potentional Field (APF) Berbasis Fuzzy Logic Controller (FLC). *Jurnal Teknik Elektro*, 12(2), 6–13. <https://doi.org/10.26740/jte.v12n2.p6-13>
- Ginting, R., Patmasari, R., & Aulia, S. (2019). Sistem Orientasi Objek Dengan Metode Stereo Vision Berbasis Raspberry Pi. *It Journal Research and Development*, 4(1), 72–85. [https://doi.org/10.25299/itjrd.2019.vol4\(1\).3562](https://doi.org/10.25299/itjrd.2019.vol4(1).3562)
- Jalal, F., & Nasir, F. (2021). Underwater Navigation, Localization and Path Planning for Autonomous Vehicles: A Review. *Proceedings of 18th International Bhurban Conference on Applied Sciences and Technologies, IBCAST* 2021, 817–828. <https://doi.org/10.1109/IBCAST51254.2021.9393315>
- Kurnia AR, H. (2025). Implementasi Iot Pada Sistem Monitoring Suhu Dan Kelembaban Menggunakan Esp32, Firebase Dan Kodular. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(1), 1781–1787. <https://doi.org/10.36040/jati.v9i1.12874>
- Liu, Y., & Wang, X. (2021). Research on obstacle avoidance technology of fixed wing formation based on improved artificial potential field method with stereo vision. *MATEC Web of Conferences*, 336, 07007. <https://doi.org/10.1051/mateconf/202133607007>
- Lu, P., Dong, B., Gao, X., Zhang, F., Song, Y., Liu, Z., & Zhang, Z. (2025). Variable-Stiffness Underwater Robotic Systems: A Review. *Journal of Marine Science and Engineering*, 13(9), 1–26. <https://doi.org/10.3390/jmse13091805>

- Nurhalim, K., Siradjuddin, I., & Putri, R. I. (2022). Implementasi Algoritma Potential Field Obstacle Avoidance Pada Robot Beroda 4 Omni Wheels Simetris. In *Journal of Applied Smart Electrical Network and Systems* (Vol. 3, Issue 01, pp. 16–24). <https://doi.org/10.52158/jasens.v3i01.417>
- Nursusanto, U., Khairunnisa', K., & Hartoyo, H. (2022). Real Time Battery Monitoring Control in Mini Generating System. *Jurnal Edukasi Elektro*, 6(2), 96–104. <https://doi.org/10.21831/jee.v6i2.54299>
- NVIDIA, & Kit, L. (2023). *Jetson Orin NX Module Datasheet (DS-10712-001, v0.5)*. NVIDIA Corporation.
- Rovmaker. (2023a). *High resolution 300m depth pressure sensor (MS5837)*. <https://rovmaker.org/product/high-resolution-300m-depth-pressure-sensor/>
- Rovmaker. (2023b). *New T200 Brushless ROV Thruster*. Rovmaker. <https://rovmaker.org/product/new-t200-brushless-rov-thruster/>
- RovMaker. (2023). *Thruster ESC*. RovMaker. <https://rovmaker.org/product/thruster-esc/>
- Sai Prasad Reddy, M., Pranav, K., Rahman, W., Vijay Kumar, B., Jain, A., & Gurulakshmi, A. B. (2025). Speed Control of BLDC Motor using PWM and Arduino Uno. *E3S Web of Conferences*, 619, 1–13. <https://doi.org/10.1051/e3sconf/202561902007>
- Saleh, M. S., Al Mashhadany, Y., Alshaibi, M., Ameen, F. M., & Algburi, S. (2025). Optimal Mobile Robot Navigation for Obstacle Avoidance Based on ANFIS Controller. *Journal of Robotics and Control (JRC)*, 6(1), 484–492. <https://doi.org/10.18196/jrc.v6i1.24882>
- Suparno, W., & Jalil, A. (2023). Implementasi Robot Arm Sebagai Peningkat Jaga Jarak Berbasis Vision Menggunakan Ros 2 Dan Raspberry Pi Implementation of Arm Robot for Keep Distance Reminder Based on Vision Using Ros 2 and Raspberry Pi. *Jurnal Elektro Luceat*, 8(2). <https://doi.org/https://doi.org/10.32531/jelekn.v8i2.512>
- Wang, P., Liu, X., & Song, A. (2025). Actuation and Locomotion of Miniature Underwater Robots: A Survey. *Engineering*, 51, 195–214. <https://doi.org/10.1016/j.eng.2024.10.022>
- Widodo, A., & Sumaedi, A. (2024). Pengembangan Sistem Embedded pada Robot

- Penghindar Objek Berbasis Sensor HC-SR04. *Jurnal Ilmu Komputer Dan Bisnis*, 15(2a), 174–184. <https://doi.org/10.47927/jikb.v15i2a.909>
- WitMotion. (2023). *Accelerometer with Self-Recovering Magnetic Field Chip Inclinator HWT905 TTL (MPU-9250)*. WitMotion. <https://witmotion-sensor.com/products/accelerometer-with-self-recovering-magnetic-field-chip-inclinometer-hwt905-ttl-mpu-9250>
- Xing, T., Wang, X., Ding, K., Ni, K., & Zhou, Q. (2023). Improved Artificial Potential Field Algorithm Assisted by Multisource Data for AUV Path Planning. *Sensors*, 23(15). <https://doi.org/10.3390/s23156680>
- Yang, Y., Xiao, Y., & Li, T. (2021). A Survey of Autonomous Underwater Vehicle Formation: Performance, Formation Control, and Communication Capability. *IEEE Communications Surveys and Tutorials*, 23(2), 815–841. <https://doi.org/10.1109/COMST.2021.3059998>
- Yu, H., & Ning, L. (2023). Coordinated Obstacle Avoidance of Multi-AUV Based on Improved Artificial Potential Field Method and Consistency Protocol. *Journal of Marine Science and Engineering*, 11(6). <https://doi.org/10.3390/jmse11061157>
- Zalewski, J., & Hożyń, S. (2024). Computer Vision-Based Position Estimation for an Autonomous Underwater Vehicle. *Remote Sensing*, 16(5). <https://doi.org/10.3390/rs16050741>
- Zhang, J., Han, F., Han, D., Yang, J., Zhao, W., & Li, H. (2024). Advanced Underwater Measurement System for ROVs : Integrating Sonar and Stereo Vision for Enhanced Subsea Infrastructure Maintenance. *Journal of Marine Science and Engineering*, 12(2), 306. <https://doi.org/10.3390/jmse12020306>

- Adi Rahmad Ramadhan, Khumaidi, A., Mustika Kurnia Mayangsari, Mat Syai'in, Imam Sutrisno, & Aulia Rahma Annisa. (2025). Penerapan Harris Corner Detection dan YOLOv5 pada Kamera Stereo Vision untuk Estimasi Jarak Robot Sepak Bola Beroda KRSBI-B. *Jurnal Elektronika Dan Otomasi Industri*, 12(1), 111–122. <https://doi.org/10.33795/elkolind.v12i1.7254>
- Adryady, D. F., Prasetyo, A., Shatyaziamawan, H. P., & Priambodo, A. S. (2022). Implementasi Algoritma Path Planning dan Mapping Arena pada Mobile Robot. *Jurnal Teknik Elektro Dan Komputer TRIAC*, 9(2), 41–45. <https://journal.trunojoyo.ac.id/triac/article/view/13215>
- Al-Batati, A. S., Koubaa, A., & Abdelkader, M. (2024). *ROS 2 Key Challenges and Advances: A Survey of ROS 2 Research, Libraries, and Applications*. <https://doi.org/10.20944/preprints202410.1204.v1>
- Amini, S., Informatika, T., Informasi, F. T., Luhur, U. B., Utara, P., Lama, K., Switch, M. D., Ruangan, K., Keamanan, P., Dengan, R., Pir, S., Magnetic, D., Switch, D., & Web, B. (2021). Perancangan Keamanan Ruangan Dengan Sensor Pir. *Juli*, 4(2), 50–56.
- Chen, C., Liu, T., He, L., Zhu, Y., & Yuan, F. (2025). Real-Time Underwater Vision Sensing System for AUV Tracking. *IET Image Processing*, 19(1). <https://doi.org/10.1049/ipr2.70183>
- Chen, R., & Chen, Y. (2023). Improved Convolutional Neural Network YOLOv5 for Underwater Target Detection Based on Autonomous Underwater Helicopter. *Journal of Marine Science and Engineering*, 11(5), 989. <https://doi.org/10.3390/jmse11050989>
- Cong, Y., Gu, C., Zhang, T., & Gao, Y. (2021). Underwater robot sensing technology: A survey. *Fundamental Research*, 1(3), 337–345. <https://doi.org/10.1016/j.fmre.2021.03.002>
- Dai, Z., & Jing, L. (2021). Lightweight extended kalman filter for marg sensors attitude estimation. *IEEE Sensors Journal*, 21(13), 14749–14758. <https://doi.org/10.1109/JSEN.2021.3072887>
- Dijaya, R., & Setiawan, H. (2023). *BUKU AJAR PENGOLAHAN CITRA DIGITAL*. UMSIDA PRESS. <https://doi.org/https://doi.org/10.21070/2023/978-623-464-075-5>

- Exploration, D. (2023). *DWE exploreHD*. DeepWater Exploration. <https://dwe.ai/products/explorehd>
- Fan, X., Guo, Y., Liu, H., Wei, B., & Lyu, W. (2020). Improved Artificial Potential Field Method Applied for AUV Path Planning. *Mathematical Problems in Engineering*, 2020. <https://doi.org/10.1155/2020/6523158>
- Firmansyah, F. I., Zuhrie, M. S., Rohman, M., Endryansyah, E., & Aulia, M. (2023). Sistem Obstacle Avoidance Pada Omnidirectional Mobile Robot Dengan Metode Artificial Potentional Field (APF) Berbasis Fuzzy Logic Controller (FLC). *Jurnal Teknik Elektro*, 12(2), 6–13. <https://doi.org/10.26740/jte.v12n2.p6-13>
- Ginting, R., Patmasari, R., & Aulia, S. (2019). Sistem Orientasi Objek Dengan Metode Stereo Vision Berbasis Raspberry Pi. *It Journal Research and Development*, 4(1), 72–85. [https://doi.org/10.25299/itjrd.2019.vol4\(1\).3562](https://doi.org/10.25299/itjrd.2019.vol4(1).3562)
- Jalal, F., & Nasir, F. (2021). Underwater Navigation, Localization and Path Planning for Autonomous Vehicles: A Review. *Proceedings of 18th International Bhurban Conference on Applied Sciences and Technologies, IBCAST* 2021, 817–828. <https://doi.org/10.1109/IBCAST51254.2021.9393315>
- Kurnia AR, H. (2025). Implementasi Iot Pada Sistem Monitoring Suhu Dan Kelembaban Menggunakan Esp32, Firebase Dan Kodular. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(1), 1781–1787. <https://doi.org/10.36040/jati.v9i1.12874>
- Liu, Y., & Wang, X. (2021). Research on obstacle avoidance technology of fixed wing formation based on improved artificial potential field method with stereo vision. *MATEC Web of Conferences*, 336, 07007. <https://doi.org/10.1051/mateconf/202133607007>
- Lu, P., Dong, B., Gao, X., Zhang, F., Song, Y., Liu, Z., & Zhang, Z. (2025). Variable-Stiffness Underwater Robotic Systems: A Review. *Journal of Marine Science and Engineering*, 13(9), 1–26. <https://doi.org/10.3390/jmse13091805>
- Nurhalim, K., Siradjuddin, I., & Putri, R. I. (2022). Implementasi Algoritma Potential Field Obstacle Avoidance Pada Robot Beroda 4 Omni Wheels

- Simetris. In *Journal of Applied Smart Electrical Network and Systems* (Vol. 3, Issue 01, pp. 16–24). <https://doi.org/10.52158/jasens.v3i01.417>
- Nursusanto, U., Khairunnisa', K., & Hartoyo, H. (2022). Real Time Battery Monitoring Control in Mini Generating System. *Jurnal Edukasi Elektro*, 6(2), 96–104. <https://doi.org/10.21831/jee.v6i2.54299>
- NVIDIA, & Kit, L. (2023). *Jetson Orin NX Module Datasheet (DS-10712-001, v0.5)*. NVIDIA Corporation.
- Rey, L., Bernardos, A. M., Dobrzycki, A. D., Carramiñana, D., Bergesio, L., Besada, J. A., & Casar, J. R. (2025). A Performance Analysis of You Only Look Once Models for Deployment on Constrained Computational Edge Devices in Drone Applications. *Electronics (Switzerland)*, 14(3), 1–25. <https://doi.org/10.3390/electronics14030638>
- Rovmaker. (2023a). *High resolution 300m depth pressure sensor (MS5837)*. <https://rovmaker.org/product/high-resolution-300m-depth-pressure-sensor/>
- Rovmaker. (2023b). *New T200 Brushless ROV Thruster*. Rovmaker. <https://rovmaker.org/product/new-t200-brushless-rov-thruster/>
- RovMaker. (2023). *Thruster ESC*. RovMaker. <https://rovmaker.org/product/thruster-esc/>
- Sai Prasad Reddy, M., Pranav, K., Rahman, W., Vijay Kumar, B., Jain, A., & Gurulakshmi, A. B. (2025). Speed Control of BLDC Motor using PWM and Arduino Uno. *E3S Web of Conferences*, 619, 1–13. <https://doi.org/10.1051/e3sconf/202561902007>
- Saleh, M. S., Al Mashhadany, Y., Alshaibi, M., Ameen, F. M., & Algburi, S. (2025). Optimal Mobile Robot Navigation for Obstacle Avoidance Based on ANFIS Controller. *Journal of Robotics and Control (JRC)*, 6(1), 484–492. <https://doi.org/10.18196/jrc.v6i1.24882>
- Suparno, W., & Jalil, A. (2023). Implementasi Robot Arm Sebagai Peningat Jaga Jarak Berbasis Vision Menggunakan Ros 2 Dan Raspberry Pi Implementation of Arm Robot for Keep Distance Reminder Based on Vision Using Ros 2 and Raspberry Pi. *Jurnal Elektro Luceat*, 8(2). <https://doi.org/https://doi.org/10.32531/jelekn.v8i2.512>
- Wang, P., Liu, X., & Song, A. (2025). Actuation and Locomotion of Miniature

- Underwater Robots: A Survey. *Engineering*, 51, 195–214.
<https://doi.org/10.1016/j.eng.2024.10.022>
- Widodo, A., & Sumaedi, A. (2024). Pengembangan Sistem Embedded pada Robot Penghindar Objek Berbasis Sensor HC-SR04. *Jurnal Ilmu Komputer Dan Bisnis*, 15(2a), 174–184. <https://doi.org/10.47927/jikb.v15i2a.909>
- WitMotion. (2023). *Accelerometer with Self-Recovering Magnetic Field Chip Inclinator HWT905 TTL (MPU-9250)*. WitMotion. <https://witmotion-sensor.com/products/accelerometer-with-self-recovering-magnetic-ield-chip-inclinometer-hwt905-ttl-mpu-9250>
- Xing, T., Wang, X., Ding, K., Ni, K., & Zhou, Q. (2023). Improved Artificial Potential Field Algorithm Assisted by Multisource Data for AUV Path Planning. *Sensors*, 23(15). <https://doi.org/10.3390/s23156680>
- Yang, M., Sha, Z., & Zhang, F. (2025). Learning-Based Leader Localization for Underwater Vehicles With Optical-Acoustic-Pressure Sensor Fusion. *IEEE/ASME Transactions on Mechatronics*, 1–11.
<https://doi.org/10.1109/TMECH.2025.3603065>
- Yang, Y., Xiao, Y., & Li, T. (2021). A Survey of Autonomous Underwater Vehicle Formation: Performance, Formation Control, and Communication Capability. *IEEE Communications Surveys and Tutorials*, 23(2), 815–841.
<https://doi.org/10.1109/COMST.2021.3059998>
- Yu, H., & Ning, L. (2023). Coordinated Obstacle Avoidance of Multi-AUV Based on Improved Artificial Potential Field Method and Consistency Protocol. *Journal of Marine Science and Engineering*, 11(6).
<https://doi.org/10.3390/jmse11061157>
- Zalewski, J., & Hożyń, S. (2024). Computer Vision-Based Position Estimation for an Autonomous Underwater Vehicle. *Remote Sensing*, 16(5).
<https://doi.org/10.3390/rs16050741>

LAMPIRAN

Lampiran 1. Biodata Mahasiswa

1. Nama : Syahrizal Aditya Kuncoro
2. NRP : 0922040096
3. Progam Studi : D4 – Teknik Otomasi
4. Agama : Islam
5. Status : Belum Menikah
6. Alamat Asal : Griya Permata Gedangan Blok H2 No. 20
7. Nomor Telepon : 081357914853
8. Jenis Kelamin : Laki-Laki
9. Email : syahrizaladitya@student.ppns.ac.id
10. Tempat, Tanggal Lahir : Jombang, 08 Maret 2004
11. Nama Orang Tua / Wali : Rudy Kuncoro
12. Alamat Orang Tua / Wali : Griya Permata Gedangan Blok H2 No. 20



PENDIDIKAN FORMAL			
Pendidikan	Tahun	Tempat Pendidikan	Jurusan
Diploma IV	2022 - 2026	Politeknik Perkapalan Negeri Surabaya	Teknik Otomasi
SMK	2019 - 2022	SMKN 3 Buduran	Teknik Komputer dan Jaringan
SMP	2015 - 2018	SMPN 1 Gedangan	-
SD	2010 - 2016	SDI Raudatul Jannah	-

(Halaman ini sengaja dikosongkan)



(Halaman ini sengaja dikosongkan)