

**IMPLEMENTASI TEKNOLOGI *IMAGE PROCESSING*
UNTUK ALAT PENGHITUNG BENIH IKAN NILA**

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Sarjana Terapan/Diploma IV Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh:

Fadillah Aslammy Pratama Sayti NIM : 1052237

Muhammad Putra Deswalki NIM : 1052251

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2025**

LEMBAR PENGESAHAN

IMPLEMENTASI TEKNOLOGI IMAGE PROCESSING UNTUK ALAT PENGHITUNG BENIH IKAN NILA

Oleh:

Fadillah Aslammy Pratama Sayti/1052237

Muhammad putra Deswalki/1052251

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Sarjana Terapan/Diploma IV Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1

Indra Dwisaputra, S.ST., M.T.

Pembimbing 2

Sirlus Andreanto Jasman Duli, S.Pd., M.T

Penguji 1

Zanu Saputra, S.ST., M.Tr.T.

Penguji 2

Ade Putra Maulana, S.Tr.T., M.Tr.T.

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa 1: Fadillah Aslammy Pratama Sayti NIM: 1052237

Nama Mahasiswa 2: Muhammad Putra Deswalki NIM: 1052260

Dengan Judul : Implementasi Teknologi Image Processing untuk
Alat Penghitung Benih Ikan Nila

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 08 Desember 2025

Nama Mahasiswa

Tanda Tangan

1. Fadillah Aslammy Pratama Sayti



2. Muhammad Putra Deswalki



ABSTRAK

Ikan nila merupakan salah satu jenis ikan yang banyak dikonsumsi oleh masyarakat di Indonesia terutama di wilayah Bangka. Hal ini dikarenakan mudahnya perawatan ikan tersebut yang membuat harganya tidak terlalu tinggi dan mudah dibudidayakan. Namun dalam proses perhitungan benih ikan secara manual masih sering dilakukan oleh pembudidaya ikan, terutama pada tempat yang kami survei yaitu Balai Benih Ikan (BBI) Sungailiat Kabupaten Bangka. Metode ini cenderung kurang akurat, memerlukan waktu lama, dan berisiko terhadap stres benih ikan. Oleh karena itu, penelitian ini bertujuan untuk merancang dan mengembangkan alat penghitung benih ikan nila secara otomatis menggunakan teknologi image processing dengan metode YOLOv5. Kami menggunakan metode YOLOv5 dikarenakan metode ini merupakan metode YOLO yang mudah dipakai terutama untuk proses deteksi objek. Sistem ini memanfaatkan kamera sebagai input, Raspberry Pi 5 sebagai perangkat pemroses, dan algoritma YOLOv5 untuk mendeteksi serta menghitung jumlah benih ikan nila. Dataset citra benih ikan nila dikumpulkan dan dianotasi untuk melatih model deteksi objek. Metode YOLOv5 ini mempunyai performa yang tinggi yaitu nilai Accuracy diatas 28,33%, nilai Precision sebesar 1. Recall diatas 28,33%, dan F1 score bernilai diatas 44,16%. Hasil pengujian menunjukkan bahwa sistem mampu mendeteksi dan menghitung benih ikan dengan tingkat akurasi yang tinggi dengan proses perhitungan yang lebih cepat 7 detik untuk setiap 10 ekor benih ikan nila dibandingkan metode manual. Implementasi alat ini diharapkan dapat membantu petani ikan dalam proses grading dan pendataan benih secara efisien, akurat, dan berkelanjutan.

Kata kunci: Penghitung benih ikan nila, deteksi objek, image processing, Raspberry Pi, YOLOv5.

ABSTRACT

Tilapia is a type of fish widely consumed by people in Indonesia, especially in the Bangka region. This is because the fish is easy to care for, which makes the price not too high and easy to cultivate. However, the process of counting fish seeds manually is still often done by fish farmers, especially in the location we surveyed, namely the Balai Benih Ikan (BBI), Bangka Regency. This method tends to be inaccurate, takes a long time, and risks stressing the fish seeds. Therefore, this study aims to design and develop an automatic tilapia seed counting tool using image processing technology with the YOLOv5 method. We used the YOLOv5 method because this method is easy to use, especially for the object detection process. This system uses a camera as input, a Raspberry Pi 5 as a processing device, and the YOLOv5 algorithm to detect and count the number of tilapia seeds. The tilapia seed image dataset was collected and annotated to train the object detection model. The YOLOv5 method has high performance, with an accuracy value was above 28.33% and a precision value of 1. The recall rate was above 28.33%, and the F1 score was above 44.16%. Test results showed that the system was able to detect and count fish fry with a high level of accuracy, with the calculation process being 7 seconds faster for every 10 tilapia fry compared to manual methods. Implementation of this tool is expected to assist fish farmers in grading and recording fry efficiently, accurately, and sustainably.

Keywords: Counter tilapia fish seed, object detection, image processing, Raspberry Pi, YOLOv5.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh. Segala puji bagi Allah SWT yang telah melimpahkan taufiq dan hidayah-Nya kepada penulis, sehingga penulis dapat menyusun laporan proyek akhir ini dengan Judul "Implementasi Teknologi Image Processing untuk Alat Penghitung Benih Ikan Nila". Sholawat serta salam semoga selalu tercurahkan kepada Nabi Muhammad SAW beserta keluarga dan sahabat-sahabatnya, serta semoga semua umatnya senantiasa dapat menjalankan syari'at-syari'atnya, Aamiin.

Penulis menyadari sepenuhnya, bahwa dalam penyusunan laporan Proyek Akhir ini banyak terdapat kekurangan mengingat terbatasnya kemampuan penulis, namun berkat rahmat Allah SWT, serta pengarahan dari berbagai pihak, akhirnya laporan proyek akhir ini dapat diselesaikan. Harapan penulis semoga laporan proyek akhir ini dapat bermanfaat untuk kepentingan bersama.

Sehubungan dengan itu, penulis menyampaikan ucapan terima kasih kepada:

1. Ibunda dan Ayahanda tercinta serta seluruh keluarga yang dengan penuh keikhlasan dan kesungguhan hati memberikan bantuan moral dan spiritual yang tak ternilai harganya.
2. Bapak I Made Andik Setiawan, M.Eng, Ph.D. Selaku Direktur di Politeknik Manufaktur Negeri Bangka Belitung yang telah banyak memberikan kemudahan dalam menyelesaikan pendidikan.
3. Bapak Aan Febriansyah, S.ST., M.T. selaku Ka. Prodi D4 Elektronika Politeknik Manufaktur Negeri Bangka Belitung dan Bapak Surojo, S.T., M.T. selaku wali kelas III STEB.
4. Bapak Indra Dwisaputra, S.ST., M.T. dan Bapak Sirlus Andreanto Jasman Duli, S.Pd., M.T. Selaku dosen pembimbing yang telah membimbing, mengarahkan dan memberi saran-saran dalam pembuatan dan penyusunan laporan proyek akhir ini.

5. Dosen dan Staf Pengajar di Politeknik Manufaktur Negeri Bangka Belitung yang telah mendidik, membina dan mengantarkan penulis untuk menempuh kematangan dalam berfikir dan berperilaku.
6. Teman–teman seperjuangan dan semua pihak yang telah memberikan bantuannya serta sahabat-sahabat yang selalu memberikan support selama ini dan mitra kerja penulis selama mengerjakan proyek akhir ini yang selalu berjuang bersama-sama.

Setelah melalui proses yang panjang dan penuh tantangan, akhirnya penulis dapat menyelesaikan pembuatan alat dan laporan proyek akhir ini yang tentunya masih banyak kekurangan dan jauh dari kesempurnaan. Walaupun demikian, penulis berharap laporan proyek akhir ini dapat bermanfaat bagi kita semua dan penulis khususnya. Semoga Allah senatiasa melimpahkan taufiq dan hidayah-Nya kepada penulis dan semua pihak yang telah membantu dalam pembuatan alat dan penulisan laporan proyek ini, Wassalamua ’laikum Warahmatullahi Wabarakatuh.

DAFTAR ISI

LEMBAR PENGESAHAN.....	Error! Bookmark not defined.
PERNYATAAN BUKAN PLAGIAT	Error! Bookmark not defined.
ABSTRAK	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
BAB I	12
PENDAHULUAN	12
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan Proyek Akhir.....	3
1.4 Batasan Masalah.....	3
BAB II.....	3
DASAR TEORI.....	3
2.1 Ikan Nila.....	7
2.2 Pengolahan Citra	8
2.3 Raspberry Pi	10
2.4 Webcam.....	12
2.5 Python	13
BAB III.....	16
METODE PENELITIAN.....	16
3.1 Identifikasi Masalah dan Studi Literatur	17

3.2 Perancangan Kontruksi	18
3.3 Pembuatan Kontruksi Alat.....	19
3.4 Pembuatan Program	21
3.5 Perancangan Sistem	22
3.6 Pengujian Sistem Kerja.....	25
3.7 Pengambilan Data dan Analisa.....	26
3.8 Pembuatan Laporan Proyek Akhir	27
BAB IV	27
PEMBAHASAN.....	27
4.1 Pengumpulan Data Citra.....	28
4.2 Pelabelan Data Citra.....	29
4.3 <i>Training/Pelatihan Data.....</i>	33
4.4 Proses Deteksi YOLOv5.....	37
4.5 Pengujian Performa Alat.....	43
BAB V.....	51
KESIMPULAN	51
3.1 Kesimpulan	51
3.2 Saran.....	52
DAFTAR PUSTAKA	53
LAMPIRAN	55

DAFTAR TABEL

Table 2.1. <i>State of The Art</i>	4
Table 3.1. Peralatan untuk Kontruksi.....	20
Table 4.1. Hasil Pengujian Secara <i>Realtime</i>	44
Tabel 4.2. Hasil <i>Confusion Matrix</i>	48
Tabel 4.3. Nilai <i>Confusion Matrix</i>	49



DAFTAR GAMBAR

Gambar 2.1. Ikan Nila.....	8
Gambar 2.2. Raspberry Pi 5	12
Gambar 2.3. <i>Webcam</i>	13
Gambar 2.4. <i>Python</i>	15
Gambar 3.1. <i>Flowchart</i> Metode Pelaksanaan	16
Gambar 3.2. Desain Kontruksi Alat.....	19
Gambar 3.3. Hasil Kontruksi Alat	21
Gambar 3.4. Skema Pembuatan Program	22
Gambar 3.5. <i>Flowchart</i> metode sistem	23
Gambar 3.6. <i>Flowchart</i> Sistem Kerja Alat.....	25
Gambar 4.1. Dataset Benih Ikan Nila	29
Gambar 4.2. Tampilan Setelah di- <i>Labelling</i>	30
Gambar 4.3. Tampilan File “best.pt”	30
Gambar 4.4. Gambar <i>Upload Dataset</i> Citra.....	31
Gambar 4.5. Pelabelan <i>Dataset</i> Citra Benih Ikan Nila	31
Gambar 4.6. Tampilan Augmentasi Dataset Citra	32
Gambar 4.7. Tampilan Setelah di- <i>Labelling</i>	33
Gambar 4.8. Tampilan <i>Link</i> Roboflow.....	33
Gambar 4.9. Tampilan Kode Program Penginstalan YOLOv5	34
Gambar 4.10. Tampilan Kode Program Untuk Library Dependensi	34
Gambar 4.11. Tampilan Penginstalan Roboflow Dan Dataset Citra Roboflow	34
Gambar 4.12. Proses Pelatihan Model (Training YOLOv5)	35
Gambar 4.13. Proses <i>Training</i>	36
Gambar 4.14. Proses Pencarian <i>Best.pt</i>	36
Gambar 4.15. File <i>Best.pt</i>	36
Gambar 4.16. Inisialisasi Perangkat Lunak dan Pemanggilan Model YOLOv5 ...	38
Gambar 4.17. Tampilan Webcam Aktif.....	39
Gambar 4.18. Proses Deteksi dan Perhitungan Jumlah Ikan.....	39
Gambar 4.19. Dataset Citra Per-frame	40
Gambar 4.20. Menganalisis YOLOv5	40

Gambar 4.21. <i>Bounding Box</i> Dan Label “Benih Ikan Nila”	41
Gambar 4.22. Jumlah Ikan dan <i>Bounding Box</i>	41
Gambar 4.23. Integrasi Servo Otomatis.....	42
Gambar 4.24. Proses <i>End</i> Layar Monitor.....	42
Gambar 4.25. <i>Confusion Matrix</i>	48



BAB I

PENDAHULUAN

1.1 Latar Belakang

Ikan merupakan salah satu bahan pangan hewani yang banyak dikonsumsi masyarakat karena kaya akan protein dan nutrisi penting lainnya yang bermanfaat bagi pertumbuhan dan kesehatan tubuh, termasuk dalam pembentukan tulang (Kementerian Kelautan dan Perikanan, 2021). Selain itu, harga ikan relatif terjangkau dan mudah didapat, menjadikannya sumber pangan utama bagi sebagian besar masyarakat di Indonesia. Indonesia sendiri merupakan negara kepulauan dengan potensi perikanan yang besar, baik dari sektor tangkap maupun budidaya. Setiap tahunnya, produksi ikan mengalami peningkatan seiring dengan meningkatnya permintaan pasar (BPS, 2023).

Untuk mendukung produksi perikanan nasional yang berkelanjutan, ketersediaan benih ikan yang unggul menjadi hal yang penting. Salah satu sektor penting dalam perikanan budidaya adalah kegiatan pemberian, yang berperan sebagai hulu produksi dalam rantai budidaya ikan. Keberhasilan usaha pemberian tidak hanya bergantung pada daya dukung lingkungan, tetapi juga kemampuan pembudidaya dalam mengelola benih, termasuk dalam hal akurasi penghitungan benih (Mulyani & Suryandari, 2018).

Selama ini, penghitungan benih ikan dilakukan secara manual maupun volumetrik. Metode manual dilakukan dengan menghitung satu per satu benih secara langsung, yang meskipun akurat, namun memakan banyak waktu dan tenaga. Sedangkan metode volumetrik menghitung benih berdasarkan volume wadah tertentu yang diasumsikan berisi sejumlah benih ikan. Metode ini lebih efisien secara waktu dan tenaga, namun rentan terhadap ketidakpastian hasil (Handoyo & Purbowaskito, 2017). Selain itu, metode estimasi berdasarkan berat juga telah digunakan, yakni dengan menimbang sejumlah benih ikan dan memperkirakan jumlah keseluruhan berdasarkan rata-rata berat per ekor.

Namun, semua metode konvensional tersebut memiliki keterbatasan dari sisi efisiensi, akurasi, dan risiko terhadap stres benih ikan. Oleh karena itu, penggunaan

teknologi seperti *image processing* menjadi solusi yang menjanjikan. Teknologi ini memungkinkan proses penghitungan benih dilakukan secara otomatis dan *realtime* dengan memanfaatkan *webcam* sebagai perangkat pemrosesan gambar.

Salah satu metode modern dalam pengolahan citra adalah YOLO (*You Only Look Once*), yaitu algoritma deteksi objek berbasis *deep learning* yang mampu mendeteksi dan menghitung objek secara cepat dan akurat dari foto atau video (Redmon et al., 2016). Beberapa penelitian telah menerapkan teknologi ini untuk keperluan di bidang perikanan, termasuk untuk mendeteksi dan menghitung benih ikan secara otomatis (Romero et al., 2023).

Pada tempat yang kami survei yaitu Balai Benih Ikan (BBI) Sungailiat Kabupaten Bangka masih menggunakan metode manual untuk menghitung benih ikan nila dalam jumlah sedikit dan volumetrik untuk menghitung benih ikan nila dalam jumlah yang banyak. Oleh karena itu, dalam penelitian ini dirancang sebuah sistem “Alat Penghitung Benih Ikan Nila Menggunakan *Image Processing* Metode YOLOv5” yang diharapkan dapat menjadi solusi efektif, efisien, dan aman dalam mendukung usaha pemberian. Sistem ini diimplementasikan secara *realtime* menggunakan Python dan *webcam*, serta hasil deteksi dapat langsung ditampilkan melalui monitor atau laptop, sehingga lebih praktis dan minim risiko terhadap benih ikan nila.

1.2 Rumusan Masalah

Adapun rumusan masalah yang diangkat pada proyek akhir ini yaitu sebagai berikut:

1. Bagaimana membuat alat penghitung benih ikan nila menggunakan teknologi *image processing*?
2. Bagaimana cara mengidentifikasi dan mendeteksi benih ikan nila dalam gambar menggunakan metode *image processing* berbasis YOLOv5?

3. Bagaimana cara merancang kontruksi untuk alat penghitung benih ikan nila agar dapat melakukan perhitungan secara optimal?

1.3 Tujuan Proyek Akhir

Adapun tujuan penelitian dalam penyusunan proyek akhir ini diantaranya adalah:

1. Membuat alat penghitung benih ikan nila menggunakan teknologi *image processing*.
2. Mengidentifikasi dan mendeteksi benih ikan nila dalam gambar menggunakan metode *image processing* berbasis YOLO.
3. Merancang kontruksi untuk alat penghitung benih ikan nila agar dapat melakukan perhitungan secara optimal.

1.4 Batasan Masalah

Adapun batasan masalah dalam proyek ahir ini adalah:

1. Menggunakan teknologi image processing dengan metode YOLOv5
2. Toleransi *error* dalam perhitungan benih ikan nila yaitu 5%.
3. Jumlah benih ikan nila untuk pengujian alat ini maksimal 50 ekor per sesi.

BAB II

DASAR TEORI

Penelitian ini menggunakan metode *Research and Development* (R&D), yaitu metode penelitian yang bertujuan untuk menyempurnakan produk yang telah ada melalui serangkaian tahapan yang sistematis. Dalam konteks penelitian ini, produk yang dikembangkan adalah alat penghitung benih ikan nila berbasis teknologi *image processing* menggunakan metode deteksi objek YOLOv5. Pada setiap tahap, penting untuk mengacu pada *state of the art*, yaitu studi atau pemahaman terkini terhadap teknologi, metode, dan hasil penelitian terbaru yang relevan. Hal ini bertujuan agar pengembangan alat dilakukan dengan pendekatan yang paling *modern*, efektif, dan terbukti secara ilmiah.

Tabel 2.1 *State of The Art*

NO	JUDUL PENELITIAN	NAMA PENELITI DAN INSTANSI	HASIL PEMBAHASAN
1.	Alat Penghitung Benih Ikan Lele Menggunakan Pengolahan Citra.	Dendi, A. S. & Sunardi, S. (2021), Politeknik Negeri Bangka Belitung	Merancang alat dengan Raspberry Pi dan kamera untuk menghitung benih lele secara otomatis. Menggunakan Python dan <i>image processing</i> untuk mendeteksi dan menghitung jumlah benih dalam sebuah wadah. Sistem berhasil mengotomatisasi penghitungan benih dengan tingkat efisiensi tinggi, namun perlu penyesuaian pada kondisi pencahayaan yang berubah-ubah.
2.	<i>Nile Tilapia Size Estimation and Tracking Using DeepSORT</i>	Romero, A. A., Cruz, K. D., Ocampo, D., dkk (2023), University	Menggabungkan YOLOv5 dan <i>DeepSORT</i> untuk mendeteksi, menghitung, dan mengestimasi ukuran ikan nila secara real-

	<i>YOLOv5 and of the East, time. DeepSORT. Philippines</i>	Digunakan dalam lingkungan perairan dengan hasil akurat. Teknologi YOLOv5 dapat digunakan secara <i>realtime</i> untuk pendekslsian dan pelacakan ikan dengan akurasi tinggi. Sistem bekerja baik dalam kondisi perairan akuakultur.
3.	Pengembangan Sistem Monitoring Kualitas Air dan Jumlah Benih Ikan Menggunakan Kamera dan <i>IoT</i> . Rahmawati, I., & Nugroho, (2020), Universitas Gadjah Mada	A. kamera dan sensor <i>IoT</i> untuk mendekksi benih dan kondisi air secara bersamaan. Penghitungan benih dilakukan dengan metode <i>image processing</i> . Sistem mampu meningkatkan efisiensi dalam monitoring dan penghitungan benih, tetapi akurasi sistem masih dipengaruhi oleh pergerakan ikan dan kondisi pencahayaan.
4.	Penerapan YOLO untuk Deteksi Objek dalam Sektor Pertanian Diponegoro dan Perikanan. Setiawan, R. & Lestari, M. (2022), Universitas	Membandingkan performa YOLOv3, YOLOv4, dan YOLOv5 dalam deteksi objek kecil seperti benih ikan dan hama tanaman. YOLOv5 menunjukkan performa terbaik dari segi kecepatan dan akurasi. YOLOv5 direkomendasikan

untuk implementasi di sektor perikanan karena mendukung pendektsian *realtime* dengan pengaturan yang relatif mudah dan fleksibel.

5.	Estimasi Jumlah Benih Ikan Berdasarkan Berat Rata-Rata.	Handoyo, R., & Purbowaskito, O.	Menganalisis Universitas Brawijaya	efektivitas metode estimasi berat dalam menghitung benih ikan dibandingkan metode manual dan volumetrik. Penggunaan timbangan tanpa air sebagai acuan jumlah benih. Metode estimasi berat lebih cepat daripada metode manual, namun memiliki kelemahan dalam hal akurasi terutama jika variasi ukuran benih terlalu besar.
----	---	---------------------------------	------------------------------------	--

Berdasarkan penelitian sebelumnya, berbagai pendekatan telah dikembangkan dalam proses penghitungan benih ikan, mulai dari metode manual, volumetrik, estimasi berat, hingga otomatisasi menggunakan pengolahan citra digital. Penelitian seperti yang dilakukan oleh Dendi & Sunardi (2021) dan Romero et al. (2023) menunjukkan bahwa penggunaan teknologi *image processing* dan algoritma YOLO efektif untuk mendekksi dan menghitung ikan secara otomatis.

Namun, sebagian besar penelitian sebelumnya masih terbatas pada spesies ikan tertentu (seperti lele) atau berfokus pada pelacakan dan estimasi ukuran ikan secara umum, bukan secara spesifik pada penghitungan benih ikan nila secara *realtime*. Pada penelitian sebelumnya menggunakan ikan lele sebagai objek penelitian. Karakteristik ikan lele yang cenderung bergerak lebih lambat dan memiliki warna

tubuh gelap membuat proses deteksi objek relatif lebih mudah, terutama pada latar belakang cerah. Namun, ukuran tubuh lele yang lebih panjang dan gerakan yang sering berada di dasar wadah menyebabkan pendektsian dalam jumlah besar menjadi kurang optimal. Selain itu, belum banyak yang mengintegrasikan YOLOv5 sebagai algoritma deteksi utama dalam sistem sederhana yang dirancang agar mudah digunakan oleh masyarakat pembudidaya ikan skala kecil.

1.1 Ikan Nila

Ikan nila (*oreochromis niloticus*) merupakan salah satu komoditas perikanan air tawar unggulan di Indonesia karena memiliki nilai ekonomi tinggi, pertumbuhan cepat, dan mudah dibudidayakan. Ikan ini bersifat eurihaline, sehingga mampu beradaptasi pada berbagai tingkat salinitas, mulai dari air tawar hingga payau, dengan kisaran ideal suhu 25–30°C dan pH 6,5–8,5. Ikan nila bersifat omnivora, memakan tumbuhan air, plankton, dan pakan buatan yang kaya nutrisi, sehingga mudah dipelihara dan biaya budayanya relatif rendah. Keunggulan lain ikan nila adalah daya tahan tubuh yang baik terhadap perubahan lingkungan, menjadikannya salah satu pilihan utama dalam budidaya skala besar.

Penelitian kami ini menggunakan ikan nila yang memiliki warna tubuh lebih cerah dan bergerak lebih aktif di kolom air, sehingga proses deteksi menjadi lebih kompleks, terutama ketika benih bergerak cepat atau saling bertumpuk. Pemilihan ikan nila juga lebih relevan dengan kebutuhan petani ikan air tawar di Bangka Belitung karena nila merupakan salah satu komoditas budidaya utama. Dengan demikian, penelitian ini tidak hanya berfokus pada pengembangan teknologi, tetapi juga lebih aplikatif pada kondisi budidaya ikan nila yang banyak dilakukan di lapangan. Adapun gambar ikan nila dapat dilihat pada gambar 2.1.



Gambar 2.1. Ikan Nila

1.2 Pengolahan Citra

Pengolahan citra (*image processing*) adalah suatu proses pengolah piksel- piksel di dalam citra digital untuk tujuan tertentu. Pengolahan citra dulunya hanya digunakan untuk memperbaiki kualitas citra, tetapi dengan adanya perkembangan dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer serta munculnya ilmu-ilmu komputasi yang memungkinkan manusia dalam mengambil informasi dari suatu citra .

Ada beberapa cara dalam menyimpan citra digital di dalam memori. Cara penyimpanan menentukan jenis citra digital yang akan terbentuk. Jenis citra digital yang paling sering digunakan adalah citra biner, citra *grayscale* dan citra warna.

- Citra biner (monokrom).

Banyaknya dua warna, yaitu hitam dan putih. Dibutuhkan 1 bit di sebuah memori untuk dapat menyimpan kedua warna ini.

- Citra *grayscale* (skala keabuan).

Banyaknya warna tergantung pada jumlah bit yang telah disediakan di dalam memori untuk menampung kebutuhan warna. Citra 2 bit mewakili beberapa warna yaitu 4 warna yang memiliki citra 3 bit, mewakili 8 warna, dan seterusnya. Jika nilai bit warna yang disediakan di memori besar, maka semakin halus juga gradasi warna yang terbentuk.

- Citra warna (*color image*).

Setiap piksel pada citra warna mewakili beberapa warna yang merupakan kombinasi dari warna dasar (RG8 = Red, Green, Blue). Setiap warna dasar digunakan dalam menyimpan 8 bit = 1 byte, yang berarti setiap warna mempunyai gradasi warna sebanyak 255 warna. Setiap piksel mempunyai kombinasi warna yang unik yaitu sebanyak 28 x 28 x 28 warna lebih. Oleh sebab itu, dinamakan color image sebab mempunyai jumlah warna yang cukup besar sehingga dapat dikatakan hampir mencakup semua warna di alam.

Monitoring dengan metode berbagai YOLOv1, dengan 24 lapisan konvolusi (*convolutional layer*) yang diikuti oleh 2 lapisan yang terhubung penuh (*fully connected layer*), sistem sudah dapat mendeteksi objek namun beberapa objek kecil tidak dapat terdeteksi oleh sistem. Lalu untuk meningkatkan kemampuan deteksinya, maka dibuatlah YOLOv2 yang merupakan hasil peningkatan dari YOLOv1 dengan meningkatkan *batch normalization*, menaikkan resolusi klasifikasi, serta menggunakan 19 *convolution layer* dengan 5 *max pooling layer*. YOLOv3 merupakan peningkatan dari YOLOv2 dengan menggunakan 53 *convolutional layer* dengan filters 3×3 dan 1×1 sehingga menghasilkan kemampuan deteksi yang jauh lebih akurat dibandingkan dengan versi sebelumnya.

Namun pada YOLOv3 memerlukan *device* yang canggih agar mendapatkan kecepatan *frame* yang sesuai. Berdasarkan masalah itu, dibuatlah YOLOv3-*Tiny* yang merupakan algoritma sederhana dari YOLOv3. Algoritma ini bekerja dengan mengurangi *depth* pada *convolutional layer* yang menyebabkan kecepatan deteksi pada algoritma ini jauh lebih cepat dibandingkan dengan YOLOv3 namun akurasi deteksi pada algoritma ini akan berkurang. Metode ini sudah banyak diaplikasikan untuk tujuan tertentu, contohnya seperti deteksi penggunaan masker, deteksi bagasi yang terbengkalai, deteksi jenis-jenis spesies ikan, deteksi kendaraan, serta masih banyak lagi pengaplikasian metode ini.

YOLOv4 memperkenalkan berbagai teknik peningkatan akurasi seperti *bag of freebies* dan *bag of specials*, namun cenderung lebih kompleks dan berat dalam implementasi. Sementara itu, YOLOv5 hadir dengan berbagai keunggulan, antara lain ukuran model yang lebih ringan, kecepatan inferensi yang lebih tinggi, serta

kemudahan dalam pelatihan dan implementasi karena ditulis sepenuhnya dalam bahasa Python menggunakan pustaka PyTorch.

Dalam penelitian ini, YOLOv5 dipilih karena menawarkan keseimbangan optimal antara akurasi dan efisiensi. YOLOv5 sangat cocok untuk digunakan pada perangkat seperti Raspberry Pi atau komputer mini lainnya, karena model ini dapat disesuaikan untuk kebutuhan ringan namun tetap memberikan hasil deteksi yang presisi. Selain itu, dukungan dokumentasi dan komunitas yang luas juga menjadi alasan pemilihan YOLOv5, sehingga mempermudah proses pengembangan alat penghitung benih ikan nila. Oleh karena itu, penggunaan metode YOLOv5 diharapkan mampu mendeteksi benih ikan secara akurat meskipun dalam kondisi bergerak cepat di dalam air sehingga memungkinkan penghitungan secara otomatis dan *realtime*.

1.3 Raspberry Pi

Raspberry Pi, sering disingkat dengan nama Raspi, adalah komputer papan tunggal (*single-board circuit*) yang seukuran dengan kartu kredit yang dapat digunakan untuk menjalankan program perkantoran, permainan komputer, dan sebagai pemutar media hingga video beresolusi tinggi. Raspberry Pi dikembangkan oleh yayasan nirlaba, Raspberry Pi Foundation, yang digawangi sejumlah pengembang dan ahli komputer dari Universitas Cambridge, Inggris.

Ide dibalik Raspberry Pi diawali dari keinginan untuk mencetak pemrogram generasi baru. Seperti disebutkan dalam situs resmi Raspberry Pi Foundation, waktu itu Eben Upton, Rob Mullins, Jack Lang, dan Alan Mycroft, dari Laboratorium Komputer Universitas Cambridge memiliki kekhawatiran melihat kian turunnya keahlian dan jumlah siswa yang hendak belajar ilmu komputer. Mereka lantas mendirikan yayasan Raspberry Pi bersama dengan Pete Lomas dan David Braben pada 2009. Tiga tahun kemudian, Raspberry Pi Model B memasuki produksi massal. Dalam peluncuran pertamanya pada akhir Februari 2012 dalam beberapa jam saja sudah terjual 100.000 unit. Pada bulan Februari 2016, Raspberry

Pi Foundation mengumumkan bahwa mereka telah menjual 8 juta perangkat Raspi, sehingga menjadikannya sebagai perangkat paling laris di Inggris.

Dalam penelitian ini, Raspberry Pi 5 dipilih sebagai perangkat utama untuk memproses data penghitungan benih ikan nila karena memiliki peningkatan performa yang signifikan dibandingkan dengan seri-seri sebelumnya. Raspberry Pi 5 dibekali dengan prosesor quad-core ARM Cortex-A76 64-bit dengan kecepatan hingga 2.4 GHz, yang merupakan lompatan besar dibandingkan Raspberry Pi 4 yang hanya menggunakan Cortex-A72 1.5 GHz. Peningkatan ini memberikan kemampuan komputasi yang jauh lebih tinggi, sehingga sangat cocok untuk menjalankan model deteksi objek seperti YOLOv5 yang membutuhkan pemrosesan intensif, terutama saat memproses citra secara *realtime*.

Selain itu, Raspberry Pi 5 memiliki dukungan RAM yang lebih besar dan lebih cepat (LPDDR4X hingga 8GB), yang penting untuk menjaga stabilitas dan efisiensi selama inferensi model *deep learning*. Raspberry Pi 5 juga telah dilengkapi dengan PCIe lane, USB 3.0 yang lebih cepat, serta sistem pendingin yang lebih baik, sehingga dapat bekerja dalam jangka waktu lebih lama tanpa mengalami *thermal throttling* seperti pada Raspberry Pi 3 atau 4.

Jika dibandingkan dengan Raspberry Pi 3 dan 4, seri 5 tidak hanya unggul dari segi kecepatan pemrosesan, tetapi juga dalam hal efisiensi daya, kestabilan, dan kompatibilitas dengan perangkat keras pendukung seperti kamera dan penyimpanan eksternal. Oleh karena itu, Raspberry Pi 5 menjadi pilihan ideal untuk mendukung sistem penghitungan benih ikan nila berbasis *image processing* yang membutuhkan pemrosesan citra secara cepat dan akurat.



Gambar 2.1. Raspberry Pi 5

1.4 Webcam

Webcam adalah perangkat keras yang digunakan untuk menangkap gambar dan video secara *realtime* melalui komputer atau perangkat lain yang kompatibel. *Webcam* berfungsi sebagai alat input visual yang sangat penting dalam sistem pengolahan citra, seperti untuk keperluan komunikasi video, pemantauan, hingga pendekripsi objek secara otomatis.

Dalam sistem penghitungan benih ikan berbasis pengolahan citra, kualitas gambar sangat berperan penting dalam menentukan akurasi deteksi objek. Oleh karena itu, pada penelitian ini digunakan *webcam* dengan resolusi 2K (2560×1440 piksel), karena mampu menghasilkan citra yang lebih tajam dan detail dibandingkan dengan *webcam* beresolusi standar seperti 720p (HD) atau 1080p (Full HD).

Dengan resolusi 2K, setiap *frame* yang diambil dari video memiliki lebih banyak piksel, sehingga model deteksi objek seperti YOLOv5 dapat mengenali bentuk, kontur, dan posisi benih ikan dengan lebih presisi, bahkan ketika benih ikan berada dalam kondisi bergerak cepat atau tumpang tindih. Detail yang tinggi juga membantu mengurangi kesalahan dalam proses anotasi data saat pelatihan model serta meminimalisir deteksi ganda (*false positives*) dan kehilangan objek (*false negatives*).

Selain itu, resolusi tinggi memungkinkan sistem untuk tetap mempertahankan akurasi walaupun dilakukan proses pemotongan (*cropping*) atau penyusutan (*resizing*) citra sebelum diproses oleh model. Ini sangat penting dalam sistem *realtime* yang tetap membutuhkan efisiensi pemrosesan tanpa mengorbankan kualitas deteksi.



Gambar 2.2. *Webcam*

1.5 Python

Python adalah bahasa pemrograman tingkat tinggi yang dirancang dengan filosofi kemudahan dalam penulisan dan keterbacaan kode. Bahasa ini dikembangkan oleh Guido van Rossum dan pertama kali dirilis pada tahun 1991. Salah satu keunggulan utama Python adalah sintaksnya yang sederhana dan mirip dengan bahasa manusia, sehingga sangat cocok digunakan oleh pemula yang baru belajar pemrograman, sekaligus tetap kuat dan fleksibel untuk digunakan oleh pengembang profesional dalam proyek-proyek besar. Python mendukung berbagai paradigma pemrograman, termasuk pemrograman berorientasi objek, fungsional, dan prosedural.

Python juga dikenal sebagai bahasa pemrograman serbaguna yang dapat digunakan di berbagai bidang, seperti pengembangan web, analisis data, kecerdasan buatan, pembelajaran mesin (*machine learning*), automasi sistem, hingga pengembangan aplikasi desktop. Hal ini dimungkinkan berkat ekosistem pustaka

(*library*) dan kerangka kerja (*framework*) yang sangat luas, seperti Django dan Flask untuk web, NumPy dan pandas untuk analisis data, serta TensorFlow dan PyTorch untuk machine learning. Selain itu, Python bersifat *open-source* dan memiliki komunitas pengguna yang sangat besar dan aktif, sehingga pengguna dapat dengan mudah menemukan dokumentasi, tutorial, maupun dukungan ketika mengalami kesulitan.

Pemilihan bahasa pemrograman Python dalam penelitian ini didasarkan pada sejumlah pertimbangan teknis dan praktis yang mendukung keberhasilan implementasi sistem pengolahan citra digital untuk penghitungan benih ikan nila. Python merupakan bahasa pemrograman yang sangat populer di bidang kecerdasan buatan, *computer vision*, dan *machine learning* karena memiliki sintaks yang sederhana, mudah dipahami, serta komunitas pengguna yang sangat luas. Python juga didukung oleh banyak pustaka (*library*) yang sangat kuat dan relevan untuk penelitian ini, seperti OpenCV untuk pengolahan citra, PyTorch untuk implementasi model *deep learning* seperti YOLOv5, serta berbagai pustaka tambahan untuk manipulasi data dan visualisasi hasil.

Keunggulan lain dari Python adalah fleksibilitasnya dalam pengembangan sistem berbasis *realtime*, kemudahan integrasi dengan perangkat keras seperti kamera atau mikrokontroler, serta dukungan ekosistem yang memungkinkan pengujian dan implementasi secara langsung pada berbagai *platform*, termasuk laptop dan Raspberry Pi. Selain itu, dokumentasi Python yang lengkap dan tersedianya banyak sumber belajar menjadikan proses pengembangan lebih efisien, khususnya bagi tim peneliti yang ingin membangun sistem secara cepat dan dapat disesuaikan. Oleh karena itu, Python dipilih dalam penelitian ini karena merupakan pilihan paling ideal dan efisien untuk membangun sistem deteksi benih ikan berbasis *image processing* dan YOLOv5 secara *realtime*, mudah digunakan, serta dapat dikembangkan lebih lanjut di masa mendatang.



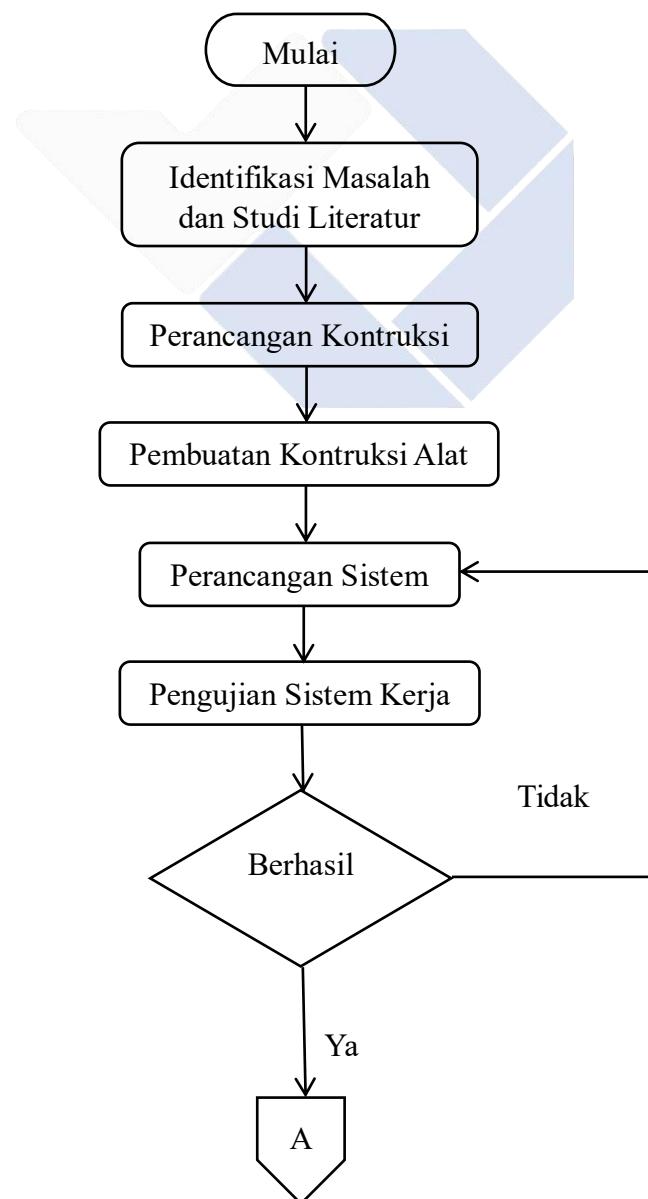
Gambar 2.3. Python

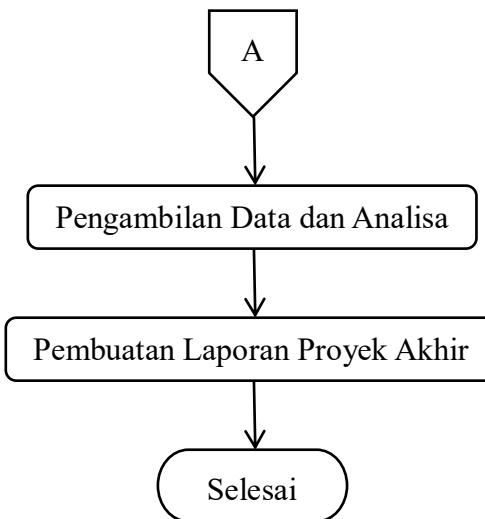


BAB III

METODE PENELITIAN

Dalam pelaksanaan proyek akhir yang berjudul "Implementasi Teknologi *Image Processing* untuk Alat Penghitung Benih Ikan Nila", dilakukan beberapa tahapan penelitian yang bertujuan untuk memudahkan penulis dalam proses pembuatan proyek akhir. Dibawah ini adalah tahapan-tahapan penelitian yang telah dirangkum dalam bentuk *flowchart* dan dapat dilihat pada gambar 3.1.





Gambar 3.1. *Flowchart* Metode Pelaksanaan

3.1 Identifikasi Masalah dan Studi Literatur

Pengumpulan data dan informasi dalam penelitian ini dilakukan untuk mendukung proses perancangan dan pengujian alat penghitung benih ikan nila berbasis teknologi *image processing* dengan metode YOLOv5. Data yang digunakan terdiri dari data primer dan data sekunder. Data primer diperoleh secara langsung melalui pengambilan gambar atau video benih ikan nila di kolam atau wadah penampungan dengan menggunakan kamera digital. Data visual ini nantinya digunakan sebagai bahan utama dalam proses pelatihan dan pengujian sistem deteksi objek.

Sementara itu, data sekunder diperoleh dari berbagai sumber literatur seperti jurnal ilmiah, artikel penelitian, dan dokumen teknis yang relevan dengan teknologi pengolahan citra, penerapan YOLOv5, serta sistem pendekripsi otomatis pada sektor perikanan. Selain itu, pengumpulan informasi juga dilakukan melalui observasi lapangan dan wawancara informal dengan pihak-pihak terkait, seperti petani ikan atau praktisi di bidang budidaya perikanan, guna mendapatkan gambaran kebutuhan di lapangan. Pada penelitian ini kami melakukan survei di Balai Benih Ikan (BBI) Sungailiat Kabupaten Bangka dan kami mendapatkan hasil bahwa mereka masih menggunakan metode manual dan volumetrik. Seluruh

data dan informasi yang diperoleh kemudian dianalisis dan dijadikan dasar dalam proses implementasi dan evaluasi sistem yang dikembangkan.

3.2 Perancangan Kontruksi

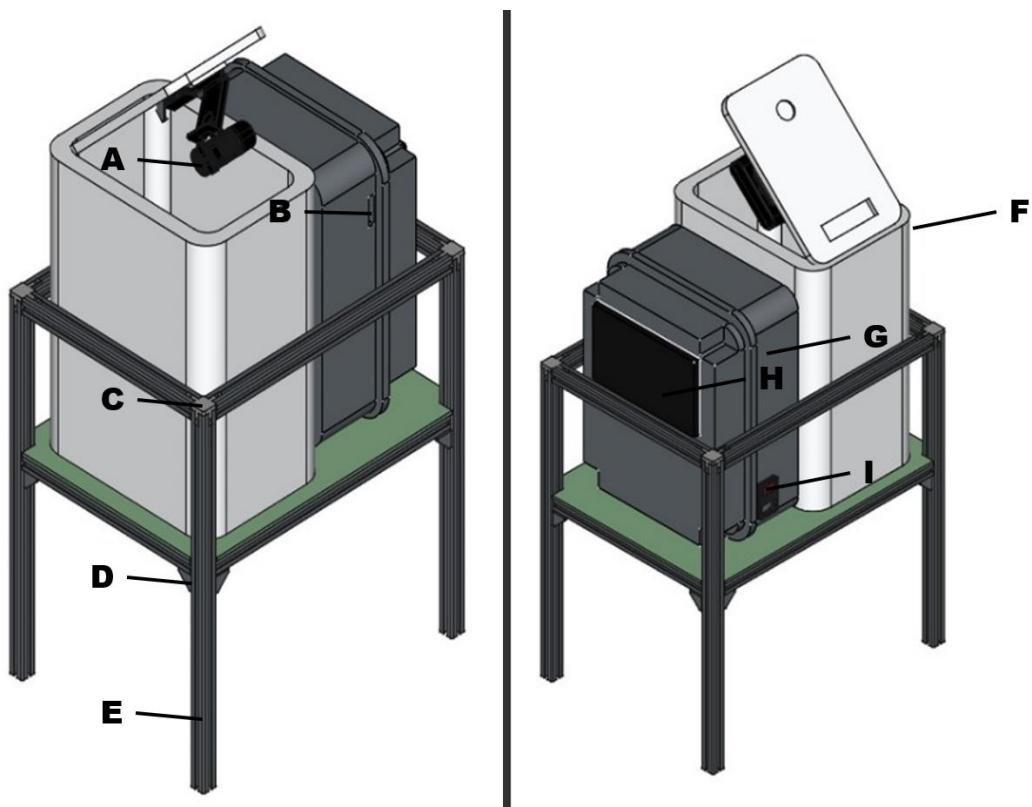
Pada tahap perancangan kontruksi alat penghitung benih ikan nila dengan menggunakan pengolahan citra ini proses yang akan dilakukan yaitu memasang *webcam* ditempat yang telah ditentukan berdasarkan perhitungan antara *webcam* dengan ukuran wadah ikan nila sehingga hasil perhitungan dapat ditampilkan pada laptop atau monitor. Sebelum membuat kontruksi alat penghitung benih ikan nila dengan menggunakan pengolahan citra siapkan terlebih dahulu rancangan alat penghitung benih ikan nila yang akan dibuat agar tidak terjadi kesalahan dalam menentukan letak dan ukuran dari bahan yang akan digunakan. Perancangan kontruksi alat pertama-tama secara manual menggunakan kertas gambar kemudian menggunakan aplikasi FreeCAD. Pembuatan desain ini disesuaikan dengan kebutuhan.

FreeCAD adalah perangkat lunak pemodelan 3D *open-source* yang dirancang untuk memenuhi berbagai kebutuhan desain teknik, arsitektur, dan manufaktur, yang memungkinkan pengguna membuat dan mengelola model parametrik dengan presisi tinggi, di mana setiap elemen desain dapat disesuaikan secara dinamis melalui parameter geometris yang telah ditentukan, memberikan fleksibilitas luar biasa dalam proses iterasi desain dan modifikasi komponen. Dikembangkan sejak tahun 2002 dengan menggunakan kombinasi bahasa pemrograman Python dan C++, FreeCAD terus berkembang berkat kontribusi komunitas global, menjadikannya solusi yang terus diperbarui dan disesuaikan dengan kebutuhan industri *modern*.

Aplikasi ini mendukung berbagai format file standar industri, seperti STEP, IGES, STL, dan banyak lagi, memudahkan integrasi dengan perangkat lunak lain dalam ekosistem desain dan manufaktur. FreeCAD dilengkapi dengan fitur-fitur canggih yang mendukung proses desain kompleks, termasuk *workbenches* modular yang dapat diaktifkan sesuai kebutuhan, seperti *Part Design* untuk pembuatan komponen mekanik, *Sketcher* untuk menggambar sketsa 2D, dan *Arch* untuk

permodelan bangunan, memungkinkan pengguna untuk bekerja secara terstruktur dan fokus pada aspek spesifik dari proyek yang sedang dikerjakan.

Dengan ketinggian 40 cm, kamera dapat menangkap keseluruhan area dalam box (panjang 28 cm × lebar 16 cm) tanpa terhalang dinding box. Ini penting agar seluruh permukaan air terlihat, memungkinkan pengambilan gambar atau video yang menyeluruh. Dengan tinggi 40 cm antara jarak yang cukup untuk melihat seluruh box dan jarak yang tidak terlalu jauh sehingga detail objek benih ikan nila masih dapat terlihat cukup jelas. Adapun desain gambar dari alat penghitung benih ikan nila dengan menggunakan pengolahan citra dapat dilihat pada gambar 3.2.



Gambar 3.2. Desain Kontruksi Alat

3.3 Pembuatan Kontruksi Alat

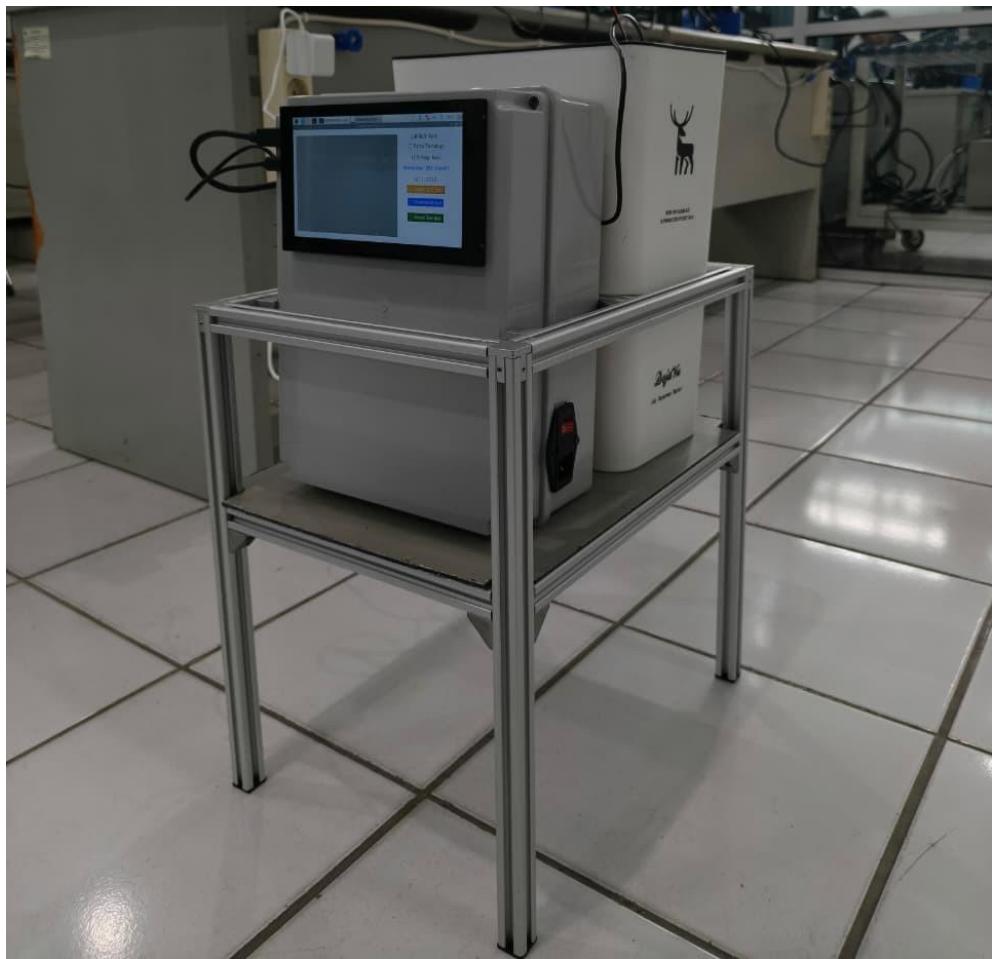
Hal yang pertama yaitu menyiapkan peralatan yang akan digunakan untuk pembuatan kontruksi alat penghitung benih ikan nila dengan menggunakan pengolahan citra. Setelah membuat desain kontruksi alat penghitung benih ikan nila

dengan menggunakan pengolahan citra selanjutnya siapkan seluruh peralatan yang digunakan untuk membuat alat penghitung benih ikan nila dan terlebih dahulu dirancang alat penghitung benih ikan nila yang akan dibuat agar tidak terjadi kesalahan agar terhindar dari pengeluaran biaya tambahan untuk membeli komponen dan peralatan. Adapun tabel peralatan yang akan digunakan untuk membuat alat penghitung benih ikan nila dapat dilihat pada tabel 3.1.

Tabel 3.1. Peralatan untuk Kontruksi

No	Alat	Jumlah	Keterangan
1.	Raspberry Pi	1	Raspberry Pi 5
2.	Webcam	1	Webcam NYK
3.	Box benih ikan	1	Nemesis A96
4.	Box panel	1	Severus
5.	Alumunium profil	12	-
6.	3 way bracket	4	-
7.	L bracket	8	-
8.	LCD	1	-
9.	Sakelar soket AC 3 in 1	1	-

Setelah menyiapkan peralatan kontruksi maka selanjutnya yaitu merakit seluruh peralatan tersebut. Dimulai dengan proses melubangi bagian atas box benih ikan sebagai tempat meletakkan webcam. Lubang tersebut digunakan untuk meletakkan skrup webcam agar dapat menempel. Kemudian pada box panel juga diberikan lubang sebagai tempat meletakkan LCD, jalur wiring, dan juga sakelar soket AC 3 in 1. Setelah semua lubang siap maka semua komponen yang diperlukan dipasang di tempatnya masing-masing. Akhirnya rangkaian kontruksi alat penghitung benih ikan nila sudah siap digunakan. Gambar hasil kontruksi dapat dilihat pada gambar 3.3.



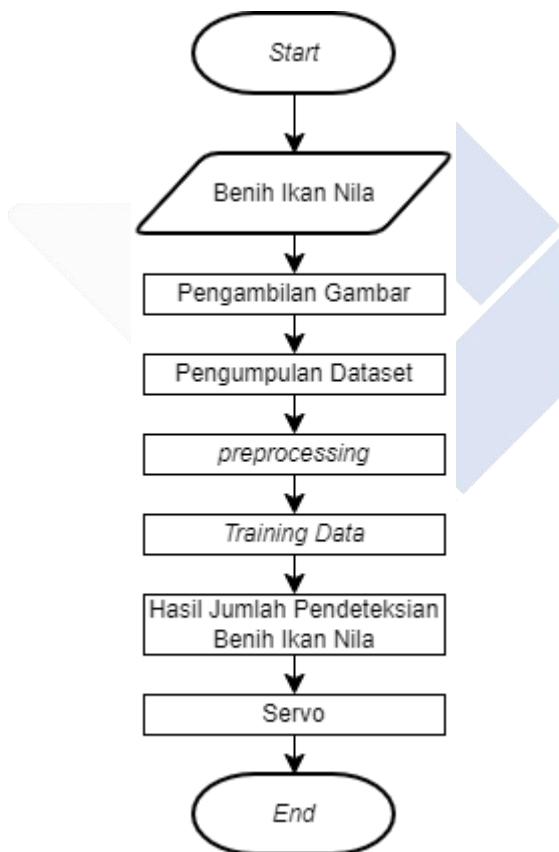
Gambar 3.3. Hasil Kontruksi Alat

3.4 Pembuatan Program

Pada tahapan ini pengumpulan data citra terlebih dahulu yaitu benih ikan nila pada proyek akhir ini diambil secara manual menggunakan kamera android. Citra yang diambil akan dikumpulkan dalam satu folder untuk *dataset*. Setelah pengumpulan data, tahap selanjutnya dilakukan Praproses data citra yang terdiri dari perubahan ukuran citra dan pelabelan. Citra yang diambil menggunakan kamera android akan dilakukan *resize* (perubahan ukuran ke ukuran yang lebih kecil) dengan 5 kali pengambilan gambar. Ukuran citra pada *dataset* sangat mempengaruhi proses *training* nanti. Selanjutnya akan dilakukan pelabelan, pelabelan citra adalah tahap awal dimana setiap citra pada *dataset* diberikan label

dengan tujuan menyampaikan informasi citra. Proses label dilakukan dengan cara memberikan *bounding box* beserta nama kelas pada setiap objek citra. Setelah pelabelan selesai, tahap selanjutnya adalah pembuatan program atau pelatihan data dengan menggunakan Google Colab dan aplikasi Visual Studio Code, selanjutnya program tersebut akan di uji coba, bisa atau tidaknya mendeteksi benih ikan nila.

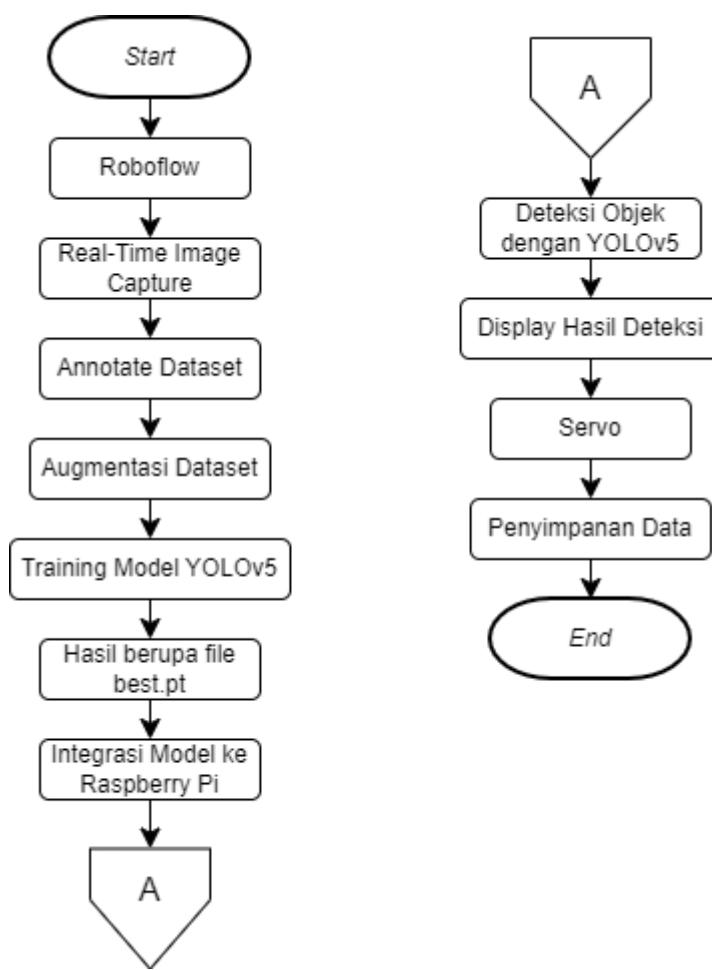
Proses pembuatan program ditunjukkan pada flowchart pada gambar 3.4 sebagai berikut :



Gambar 3. 4 Skema Pembuatan Program

3.5 Perancangan Sistem

Pada pengerjaan proyek akhir ini memiliki rancangan sistem yang digunakan. Berikut rancangan sistem akan ditunjukan pada gambar 3.5 sebagai berikut :



Gambar 3.5. Flowchart metode sistem

Alur kerja sistem ini pada tahap pertama yaitu pengumpulan *dataset* citra benih ikan nila yang dilakukan menggunakan *webcam* beresolusi tinggi. Pengambilan citra dilakukan dalam berbagai kondisi pencahayaan, sudut pandang, dan posisi ikan, baik ketika ikan diam maupun bergerak, dengan tujuan memperoleh variasi data yang kaya sehingga model dapat mendeteksi objek secara optimal. *Dataset* citra yang telah terkumpul kemudian melalui proses anotasi (pelabelan objek) menggunakan aplikasi seperti Roboflow. Setiap benih ikan diberi kotak pembatas (*bounding box*) dan dilabeli sebagai kelas “Benih Ikan Nila”. Tahap ini sangat penting karena kualitas pelabelan akan memengaruhi tingkat akurasi model deteksi.

Selanjutnya dilakukan proses augmentasi *dataset*, yaitu menambah variasi gambar dengan teknik rotasi, perubahan kecerahan, kontras, pemotongan sebagian gambar (*cropping*), hingga *flipping*. Augmentasi bertujuan agar model YOLOv5 mampu mendeteksi objek ikan secara lebih baik meskipun dalam kondisi pencahayaan rendah, ikan bergerak cepat, atau tumpang tindih satu sama lain. *Dataset* yang telah siap kemudian digunakan pada tahap pelatihan (*training*) model YOLOv5. Proses pelatihan dilakukan pada platform Google Colab yang mendukung komputasi berbasis GPU sehingga mempercepat proses pelatihan. File konfigurasi *dataset* disusun dalam format *.yaml* yang berisi jalur direktori data, jumlah kelas, dan nama kelas. Proses pelatihan dijalankan menggunakan *skrip train.py* dengan parameter tertentu seperti ukuran gambar, *batch*, *epoch*, dan bobot awal (*weights*). Setelah proses *training* selesai, diperoleh file bobot terbaik dalam format *best.pt* yang telah mempelajari pola visual benih ikan nila secara optimal.

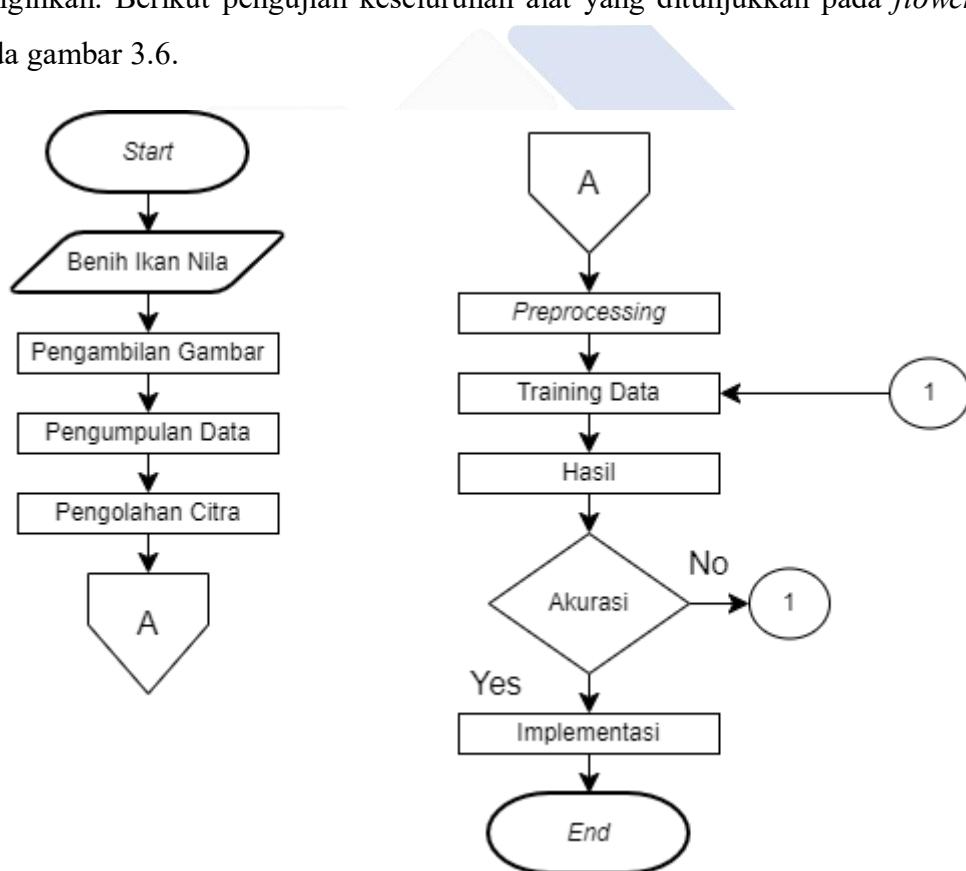
Tahap berikutnya adalah implementasi model pada perangkat Raspberry Pi sebagai pusat pemrosesan sistem. File *best.pt* dijalankan menggunakan bahasa pemrograman Python yang telah terintegrasi dengan pustaka OpenCV dan PyTorch. Raspberry Pi menerima input secara *realtime* dari *webcam* yang dipasang di atas wadah benih ikan. Gambar yang tertangkap oleh *webcam* kemudian diproses oleh model YOLOv5, sehingga sistem mampu mendeteksi objek benih ikan secara otomatis dan memberikan output berupa kotak pembatas (*bounding box*) di sekitar ikan serta label “Benih Ikan Nila” pada monitor. Jumlah benih ikan nila yang terdeteksi juga ditampilkan secara *realtime* pada sudut layar monitor.

Selain menghitung, sistem ini dapat diintegrasikan dengan servo untuk proses pengeluaran benih ikan nila ke dalam wadah yang lain sebagai tempat penampungan. Servo akan dikendalikan secara otomatis oleh Raspberry Pi berdasarkan jumlah ikan yang telah dihitung, misalnya membuka katup setiap selesai menghitung benih ikan nila yang kemudian dialiri ke wadah selanjutnya dan menutup aliran saat sedang dalam proses perhitungan benih ikan nila. Proses pendekripsi berjalan secara terus-menerus hingga semua benih ikan selesai dihitung. Setelah proses selesai, sistem secara otomatis menghasilkan laporan

berupa total jumlah ikan yang terhitung, yang selanjutnya dapat digunakan untuk analisis dan validasi tingkat akurasi sistem. Dengan alur kerja ini, sistem penghitung benih ikan nilai mampu bekerja secara otomatis, cepat, dan akurat dibandingkan metode manual, serta diharapkan dapat membantu petani ikan dalam proses grading maupun pendataan benih secara efisien.

3.6 Pengujian Sistem Kerja

Pada tahap ini dilakukan pengujian keseluruhan alat untuk mengetahui bekerja atau tidaknya sistem kerja alat yang telah dibuat dan apakah berfungsi sesuai yang diinginkan. Berikut pengujian keseluruhan alat yang ditunjukkan pada *flowchart* pada gambar 3.6.



Gambar 3.6. *Flowchart* Sistem Kerja Alat

Pengujian alat dilakukan untuk mengetahui apakah sistem penghitung benih ikan nilai yang telah dirancang dapat berjalan dengan baik dan memberikan hasil yang akurat. Pengujian ini mencakup beberapa aspek, seperti keakuratan pendekripsi ikan, pengaruh intensitas cahaya, serta tampilan hasil pada monitor. Pengujian

dilakukan dengan meletakkan sejumlah benih ikan nila di dalam wadah, kemudian alat dijalankan untuk mendeteksi dan menghitung jumlah ikan yang melewati area tangkapan *webcam*. *Webcam* akan merekam gambar secara langsung, lalu citra tersebut diproses oleh Raspberry Pi menggunakan model deteksi YOLOv5 yang telah dilatih sebelumnya. Hasil pendekripsi berupa jumlah ikan yang terdeteksi akan muncul di layar monitor.

Selama pengujian, jumlah ikan yang sebenarnya juga dihitung secara manual untuk dibandingkan dengan hasil dari alat. Perbandingan ini bertujuan untuk mengetahui tingkat akurasi sistem. Selain itu, diuji juga seberapa besar pengaruh intensitas cahaya terhadap hasil perhitungan. Melalui proses pengujian ini, dapat diketahui apakah alat dapat bekerja secara optimal dalam kondisi nyata, serta apakah ada kendala teknis seperti keterhambatan deteksi akibat intensitas cahaya, kesalahan hitung, atau masalah tampilan. Hasil dari pengujian ini akan menjadi dasar untuk evaluasi dan perbaikan sistem agar alat dapat digunakan secara efektif.

3.7 Pengambilan Data dan Analisa

Tahapan ini merupakan salah satu bagian penting dalam proses evaluasi sistem yang telah dikembangkan. Pada tahap ini, data yang berupa gambar dan video diambil selama proses pengujian alat, baik melalui pengambilan citra secara statis (gambar diam) maupun secara *realtime* (video langsung). Gambar-gambar tersebut direkam ketika benih ikan nila melewati area tangkapan kamera, dan sistem melakukan proses deteksi serta perhitungan secara otomatis menggunakan algoritma YOLOv5.

Proses pengujian dilakukan dalam beberapa skenario berbeda untuk memperoleh data yang bervariasi, perubahan intensitas pencahayaan, serta perbedaan posisi dan pergerakan ikan di dalam wadah. Semua data hasil pengujian tersebut kemudian dikumpulkan secara sistematis dan dianalisis menggunakan parameter evaluasi kinerja, seperti tingkat akurasi deteksi, jumlah objek yang berhasil dan tidak berhasil dikenali, serta perbandingan hasil otomatis dengan hasil perhitungan manual.

Tujuan utama dari tahapan ini adalah untuk mengidentifikasi sejauh mana alat yang telah dirancang dapat bekerja secara optimal dan efisien. Selain itu, melalui proses analisis ini dapat diketahui berbagai kekurangan dan kelebihan dari alat, baik dari segi konstruksi fisik (seperti posisi dan ketinggian kamera), kinerja sistem deteksi (akurasi model YOLOv5 terhadap objek kecil), maupun pengimplementasiannya dalam kondisi nyata di lapangan. Hasil dari analisis ini akan menjadi dasar untuk menyusun rekomendasi pengembangan alat lebih lanjut agar lebih optimal, presisi, dan mudah digunakan oleh pengguna di sektor perikanan budidaya.

3.8 Pembuatan Laporan Proyek Akhir

Tahapan ini merupakan tahapan akhir dari penggerjaan proyek akhir. Tahapan ini bertujuan untuk menyimpulkan keseluruhan dari proses penggerjaan proyek akhir yang dimana tahap ini akan memberikan informasi yang telah didapatkan dari keseluruhan proses pembuatan alat yang dikerjakan.

BAB IV

PEMBAHASAN

Pada Tahap pembahasan ini dengan judul “Implementasi Teknologi *Image Processing* untuk Alat Penghitung Benih Ikan Nila”. Proses dimulai dari benih ikan nila yang terekam oleh *webcam*. Gambar yang ditangkap kemudian dikirim ke Raspberry Pi 3 untuk diproses menggunakan teknologi deteksi objek seperti YOLOv5. Setelah jumlah ikan terdeteksi, hasilnya ditampilkan pada laptop atau monitor. Berikut tahapan-tahapan yang dilakukan pada pembahasan kali ini antara lain:

1. Pengumpulan Data Citra
2. Pelabelan Data Citra
3. *Training/Pelatihan Data*
4. Proses Deteksi YOLOv5
5. Pengujian Performa Alat

4.1 Pengumpulan Data Citra

Langkah awal yang akan dikerjakan adalah proses pengambilan citra menggunakan kamera android/HP. Dengan pengambilan gambar sebanyak 80 gambar dan setiap gambar terdapat beberapa benih ikan nila dengan jumlah 10 – 50 benih ikan nila. Setelah di augmentasi di Roboflow maka didapatkan 225 citra dengan format “*jpg*” yang semuanya merupakan citra benih ikan nila. Citra ini akan dikumpulkan dalam satu folder untuk dijadikan data atau *dataset* pelatihan jaringan YOLOv5. Hasil pengujian yang didapatkan cukup bagus. Namun ada pula objek benih ikan nila yang tidak terdeteksi saat pengujian, tetapi tidak sebanyak dari hasil yang berhasil dideteksi. Jika *dataset* yang dilatih banyak, maka hasil pendekripsi pun semakin baik. Untuk pelatihan dengan *dataset* 225 dengan ukuran input gambar sebesar 1000 x 750, waktu pelatihan yang telah kami lakukan yaitu sekitar 3 jam. Semakin banyak *dataset* maka semakin lama pula waktu pelatihan yang dilakukan. Hal ini sama dengan ukuran input gambar *dataset*. Semakin besar ukuran *dataset*

maka semakin lama pula pelatihan data yang dilakukan. Berikut contoh *dataset* yang telah kami ambil menggunakan kamera android/HP yang terdapat pada gambar dibawah ini.



Gambar 4.1. *Dataset* Benih Ikan Nila

4.2 Pelabelan Data Citra

Pelabelan data citra merupakan tahapan penting dalam proses pelatihan model deteksi objek berbasis YOLOv5. Pada tahap ini, setiap citra yang telah dikumpulkan harus diberi informasi tentang lokasi dan kelas objek benih ikan nila yang ingin dideteksi oleh sistem. Proses pelabelan dilakukan menggunakan perangkat lunak Roboflow, yaitu *software open-source* yang secara khusus digunakan untuk membuat anotasi berupa kotak pembatas (*bounding box*) pada objek di dalam gambar. Setiap benih ikan yang tampak pada gambar diberi kotak pembatas dan dilabeli dengan kelas objek, yaitu “Benih Ikan Nila”.

Pelabelan dilakukan secara manual dan teliti agar kualitas data tinggi dan akurasi deteksi model meningkat. Jumlah dan kualitas anotasi sangat berpengaruh terhadap performa akhir model YOLOv5, karena model akan belajar langsung dari informasi yang diberikan pada tahap ini. Oleh karena itu, dilakukan validasi secara visual terhadap hasil pelabelan untuk memastikan bahwa seluruh benih dalam gambar telah diberi label dengan tepat dan konsisten. berikut langkah-langkah melakukan *labeling* menggunakan Roboflow sebagai berikut :

1. *Search* di *google* ketikan roboflow.com, dimana tampilan awal nya seperti gambar di bawah ini:

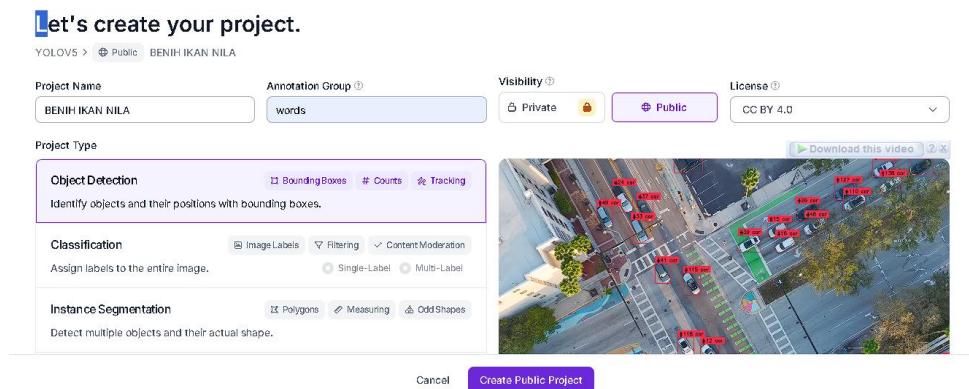
roboflow



Loading...

Gambar 4.2 Tampilan Roboflow

2. Setelah *sign up* ke Roboflow maka ada logo *new project* di kanan atas untuk memulai *project* yang dimana kami mencatatumkan nama benih ikan nila. Berikut tampilan new project yang akan di buat seperti gambar 4.3.



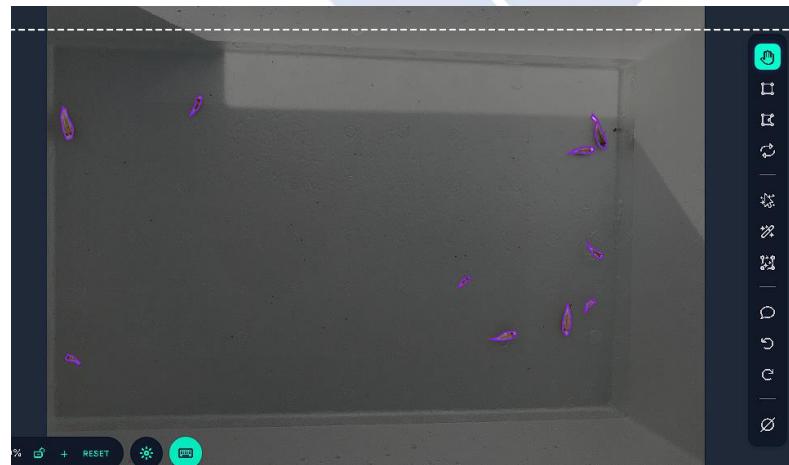
Gambar 4.3 Tampilan New Project

3. Kemudian *dataset* citra yang telah dikumpulkan dari hasil pengambilan gambar menggunakan kamera *handphone* kemudian diunggah ke Roboflow. Gambar-gambar ini memperlihatkan benih ikan nila dalam berbagai kondisi, seperti jumlah ikan yang berbeda, orientasi ikan yang bervariasi, serta pencahayaan yang tidak seragam. File citra yang diunggah disimpan dalam format *JPEG* atau *PNG*. Berikut gambar *upload dataset* citra ke Roboflow pada gambar 4.4.



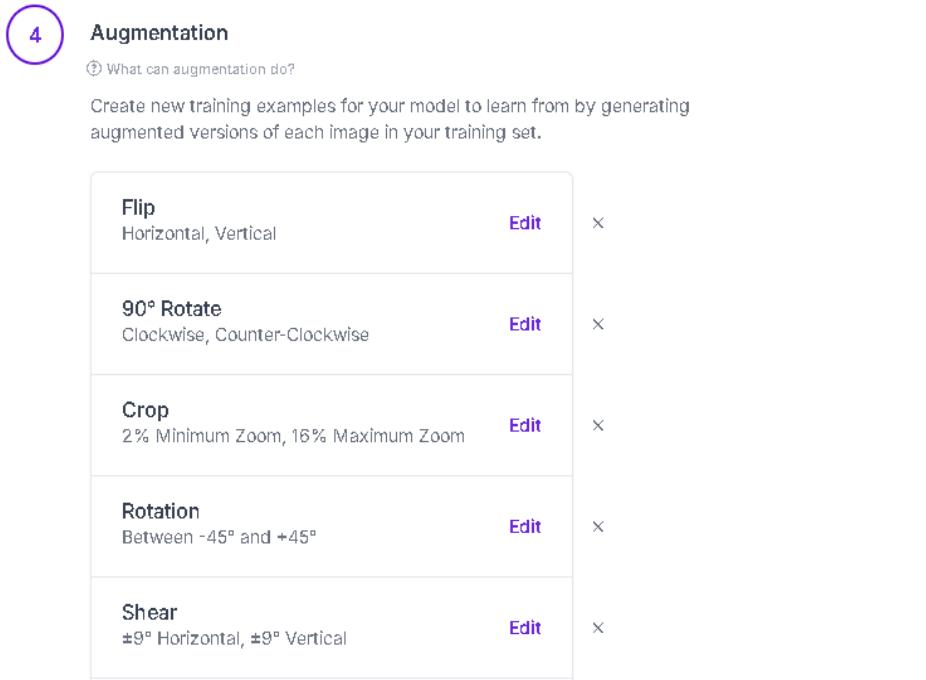
Gambar 4.4 Gambar Upload Dataset Citra

4. Setelah seluruh *dataset* berhasil diunggah, dilakukan proses anotasi manual pada setiap citra menggunakan *tool bounding box* bawaan Roboflow. Setiap ikan dalam gambar dilingkari secara teliti dengan kotak pembatas untuk menentukan posisi objek secara presisi. Label yang digunakan pada setiap *bounding box* adalah "Benih Ikan Nila" sebagai penanda bahwa objek tersebut adalah benih ikan nila. Proses ini penting agar model deteksi objek nantinya dapat belajar mengenali pola visual ikan nila secara akurat. Berikut gambar proses pelabelan benih ikan nila pada gambar 4.5



Gambar 4.5 Pelabelan *Dataset* Citra Benih Ikan Nila

5. Setelah anotasi selesai, Roboflow menyediakan fitur augmentasi *dataset* untuk memperkaya variasi data tanpa perlu mengambil gambar tambahan secara fisik. Pada proyek ini, augmentasi yang diterapkan meliputi. Berikut tampilan augmentasi pada gambar 4.6.



Gambar 4.6 Tampilan Augmentasi Dataset Citra

Disini kami menggunakan 5 augmentasi yang berbeda yaitu *flip* (horizontal dan vertikal) untuk menghasilkan gambar yang diputar arah, *rotate* 90° searah jarum jam dan berlawanan arah untuk melatih model mengenali objek dari berbagai sudut, serta *crop* dengan *zoom* minimum 2% dan maksimum 16% guna melatih model mengenali objek yang sebagian dipotong. Selain itu, digunakan juga *rotation* dengan sudut acak antara -45° hingga +45° untuk meningkatkan toleransi model terhadap kemiringan, dan *shear* hingga ±9° secara horizontal dan vertikal untuk mensimulasikan perubahan perspektif. Semua teknik ini bertujuan agar model mampu mendeteksi benih ikan dengan lebih akurat meskipun dalam kondisi atau posisi gambar yang bervariasi.

6. Setelah di augmentasi maka lanjut ke proses selanjutnya yang dimana pada proses ini semua benih ikan yang telah di label sebelumnya, akan terotomatis menampilkan hasil pelabelan *dataset* citra. Berikut gambar hasil pelabelan pada gambar 4.7.



Gambar 4.7. Tampilan Setelah di-*Labelling*

7. Setelah *dataset* selesai diberi label dan diaugmentasi, Roboflow menghasilkan *dataset* versi *final* yang siap untuk digunakan pada pelatihan model. *Dataset* dieksport dalam format YOLOv5 PyTorch. File eksport berupa ZIP atau *link*. Berikut tampilan link roboflow yang akan di *training* di google colab pada gambar 4.8.

```
!pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="REDACTED")
project = rf.workspace("yolov5-zoisy5").project("benih-ikan-nila-fix")
version = project.version(4)
dataset = version.download("yolov5")
```

Gambar 4.8 Tampilan *Link* Roboflow

4.3 Training/Pelatihan Data

Setelah tahap pra-proses data citra pada *dataset*, tahap selanjutnya yaitu melakukan *training* atau pelatihan data menggunakan metode YOLOv5. Pelatihan data akan dilakukan menggunakan Google Colab. Pada pelatihan menggunakan platform Google Colab ada beberapa tahap yang harus dilakukan antara lain :

1. Persiapan Lingkungan Kerja di Google Colab atau *Install Requirements*:

Pelatihan model deteksi objek YOLOv5 dilakukan pada platform Google Colab karena Colab menyediakan fasilitas GPU gratis yang sangat membantu dalam mempercepat proses komputasi. Penggunaan GPU diperlukan karena algoritma YOLOv5 memiliki banyak perhitungan *matriks* yang membutuhkan

kemampuan komputasi tinggi. Pada tahap ini, peneliti terlebih dahulu mengubah pengaturan *Runtime* menjadi GPU melalui menu *Runtime → Change Runtime Type → Hardware Accelerator: GPU*. Pemilihan GPU dibanding CPU biasa dilakukan karena pelatihan yang menggunakan CPU dapat memakan waktu berjam-jam bahkan berhari-hari, sedangkan GPU mampu memangkas waktu pelatihan hingga 5–10 kali lebih cepat. Setelah GPU diaktifkan, repository resmi YOLOv5 dari GitHub di-*clone* dengan perintah berikut:

```
#clone YOLOv5 and  
!git clone https://github.com/ultralytics/yolov5 # clone repo  
%cd yolov5
```

Gambar 4.9 Tampilan Kode Program Penginstalan YOLOv5

Perintah ini akan mengunduh seluruh file YOLOv5 ke direktori kerja Colab, termasuk skrip pelatihan, pengujian, dan file pendukung lainnya. Setelah proses *cloning* selesai, semua *library* dependensi yang dibutuhkan YOLOv5 diinstall menggunakan perintah pada gambar 4.10.

```
%pip install -qr requirements.txt # install dependencies
```

Gambar 4.10 Tampilan Kode Program Untuk *Library* Dependensi

2. Setelah itu melakukan *dataset* yang digunakan dalam penelitian ini adalah *dataset* dari proyek “benih-ikan-nila-fix” yang dibuat di Roboflow. *Dataset* ini telah melalui tahap anotasi (pelabelan objek) dan augmentasi (peningkatan variasi data) agar lebih optimal saat digunakan untuk melatih model YOLOv5. Untuk mempermudah, *dataset* diunduh langsung ke Google Colab menggunakan API Roboflow sehingga tidak perlu diunggah manual. Perintah Python yang digunakan adalah pada gambar 4.11.

```
!pip install roboflow  
  
from roboflow import Roboflow  
rf = Roboflow(api_key="AVgHPNDiivUnPwO2vVDR")  
project = rf.workspace("yolov5-zoliy5").project("benih-ikan-nila-fix")  
version = project.version(4)  
dataset = version.download("yolov5")
```

Gambar 4.11 Tampilan Penginstalan Roboflow Dan *Dataset* Citra Roboflow

3. Pada tahapan inti dari penelitian ini adalah melatih model deteksi objek YOLOv5 agar mampu mengenali benih ikan nila secara akurat. Pelatihan dilakukan dengan menggunakan perintah pada gambar 4.12.

```
[ ] !python train.py --img 1080 --batch 16 --epochs 50 --data {dataset.location}/data.yaml --weights yolov5s.pt --cache
```

Gambar 4.12 Proses Pelatihan Model (*Training YOLOv5*)

Berikut penjelasan kenapa kami menggunakan parameter ini:

- a. *Img 1080* → Mengatur ukuran gambar input menjadi 1080x1080 piksel, yang merupakan ukuran standar untuk YOLOv5 agar keseimbangan antara kecepatan dan akurasi tetap terjaga.
- b. *Batch 16* → Mengatur ukuran *batch* menjadi 16 gambar per iterasi pelatihan. *Batch* kecil menjaga agar pelatihan tetap stabil, terutama saat menggunakan GPU Colab.
- c. *Epochs 50* → Proses pelatihan dilakukan selama 50 *epoch* atau pengulangan penuh *dataset*, agar model cukup belajar mengenali pola visual ikan.
- d. Data *data.yaml* → Mengarahkan YOLOv5 untuk menggunakan file konfigurasi *dataset*.
- e. *Weights yolov5s.pt* → Menggunakan model YOLOv5s (*small*) yang telah dilatih sebelumnya (*pretrained model*) untuk mempercepat proses pelatihan.
- f. *Cache* → Mempercepat proses dengan menyimpan *dataset* ke memori (RAM).

Selama proses pelatihan, Colab akan menampilkan perkembangan akurasi model melalui grafik *loss* (tingkat kesalahan model) dan mAP (*mean Average Precision*). Nilai mAP yang semakin tinggi menunjukkan bahwa model semakin akurat dalam mendeteksi objek pada gambar 4.13 di bawah ini.

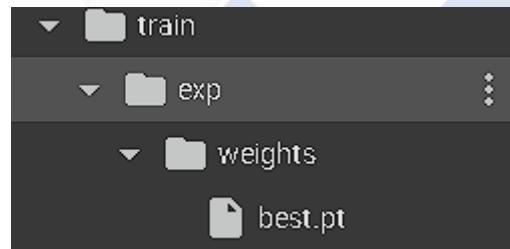
```

    Epoch   GPU_mem  box_loss  obj_loss  cls_loss Instances      Size
0% 0/13 [00:00<?, ?it/s]/content/yolov5/train.py:414: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated
...
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1336    0.1673     0    639    1088:  0% 0/13 [00:01<?, ?it/s]WARNING ▲
    0/49     9.96    0.1336    0.1673     0    639    1088:  0% 1/13 [00:02<00:35,  2.92s/it]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1326    0.1698     0    624    1088: 15% 2/13 [00:03<00:17,  1.60s/it]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1321    0.1647     0    522    1088: 23% 3/13 [00:04<00:12,  1.21s/it]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1321    0.1608     0    564    1088: 31% 4/13 [00:05<00:09,  1.01s/it]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1317    0.1549     0    417    1088: 38% 5/13 [00:05<00:06,  1.15it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1314    0.153     0    525    1088: 46% 6/13 [00:06<00:05,  1.24it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1313    0.1548     0    726    1088: 54% 7/13 [00:07<00:04,  1.28it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1312    0.1531     0    526    1088: 62% 8/13 [00:07<00:04,  1.22it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1311    0.1526     0    627    1088: 69% 9/13 [00:08<00:02,  1.34it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1309    0.1517     0    562    1088: 77% 10/13 [00:09<00:02,  1.30it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1309    0.1518     0    750    1088: 85% 11/13 [00:10<00:01,  1.29it/s]/co
with torch.cuda.amp.autocast(amp):
    0/49     9.96    0.1308    0.1506     0    607    1088: 92% 12/13 [00:10<00:00,  1.35it/s]/co
with torch.cuda.amp.autocast(amp):

```

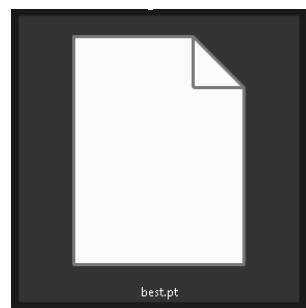
Gambar 4.13 Proses *Training*

- Setelah pelatihan selesai, YOLOv5 secara otomatis menyimpan dua file model, yaitu *last.pt* (model hasil pelatihan terakhir) dan *best.pt* (model dengan akurasi terbaik pada data validasi) file *best.pt* disimpan di direktori pada gambar 4.14.



Gambar 4.14 Proses Pencarian *Best.pt*

File *best.pt* dipilih karena secara otomatis dipilih berdasarkan nilai mAP tertinggi dan tingkat kesalahan yang paling rendah selama pelatihan. Model ini selanjutnya digunakan pada tahap pengujian (*testing*) dan implementasi. Gambar file *best.pt* yang telah di download pada gambar 4.15.



Gambar 4.15 File *Best.pt*

4.4 Proses Deteksi YOLOv5

Setelah model YOLOv5 selesai melalui tahap *training/pelatihan*, proses selanjutnya adalah melakukan deteksi objek menggunakan model terlatih tersebut. Pada penelitian ini, deteksi dilakukan terhadap objek benih ikan nila dalam gambar secara *realtime*. Proses deteksi menggunakan file model hasil pelatihan (*best.pt*) yang berisi bobot parameter hasil belajar dari *dataset* citra yang telah diberi label sebelumnya. Citra input yang berasal dari kamera atau file gambar akan dianalisis oleh model YOLOv5 untuk menemukan dan mengklasifikasikan objek benih ikan dengan menghasilkan kotak pembatas (*bounding box*) dan nilai *confidence* atau tingkat keyakinan deteksi.

Deteksi pada YOLOv5 bekerja dengan membagi gambar menjadi *grid* dan melakukan prediksi terhadap posisi, ukuran, serta kelas objek di setiap *grid*. Untuk setiap gambar, sistem akan mengeluarkan hasil berupa jumlah objek yang terdeteksi, posisi setiap objek, dan tingkat kepercayaan model. YOLOv5 sangat cocok digunakan dalam deteksi objek kecil seperti benih ikan karena memiliki kemampuan prediksi cepat dan akurat. Hasil deteksi divisualisasikan dengan kotak yang mengelilingi objek ikan dan label nama kelas yang muncul di atas kotak tersebut.

Dalam implementasinya, deteksi dijalankan dengan perintah “*detect.py*” dari repositori YOLOv5, di mana model, ukuran gambar, sumber input, dan ambang batas kepercayaan (*confidence threshold*) ditentukan oleh pengguna. Proses ini berjalan secara *realtime* dan mampu mendeteksi puluhan benih ikan dalam waktu kurang dari satu detik per gambar, tergantung pada spesifikasi perangkat keras yang digunakan. Deteksi ini juga memungkinkan perhitungan otomatis jumlah benih ikan yang tertangkap kamera, sehingga sistem dapat digunakan sebagai alat bantu akurat dalam kegiatan pemberian dan distribusi benih. Secara keseluruhan, proses deteksi menggunakan YOLOv5 terbukti efektif, cepat, dan praktis untuk diimplementasikan pada sistem penghitungan benih ikan nila secara otomatis. Berikut adalah tahapan rinci dari proses deteksi menggunakan YOLOv5:

1. Persiapan Perangkat Keras (*Hardware Setup*)

Tahap pertama dalam proses deteksi adalah menyiapkan perangkat keras yang digunakan. Raspberry Pi dipilih sebagai pusat pemrosesan karena memiliki ukuran yang kecil, konsumsi daya yang rendah, dan kompatibel untuk menjalankan model deteksi objek ringan seperti YOLOv5s. Perangkat keras lain yang digunakan adalah *webcam* sebagai sensor pengambilan citra dan monitor untuk menampilkan hasil deteksi secara langsung.

Webcam diposisikan tegak lurus di atas wadah ikan dengan tinggi 50cm dan 15cm (Gambar 3.3 Konstruksi Alat) sehingga dapat menangkap citra benih ikan nila dengan jelas dan meminimalkan distorsi perspektif. Pencahayaan diatur secukupnya agar ikan terlihat jelas tanpa bayangan berlebih. Jika sistem juga digunakan untuk proses sortir otomatis, maka servo motor dipasang pada saluran air sebagai aktuator pembuka atau penutup jalur aliran ikan.

Raspberry Pi telah dikonfigurasi sebelumnya dengan sistem operasi Raspberry Pi OS dan telah terinstal library Python, termasuk PyTorch, OpenCV, dan YOLOv5, serta telah diunggah model terlatih hasil pelatihan di Google Colab yaitu file *best.pt*.

2. Persiapan Perangkat Lunak

Setelah perangkat keras siap, tahap selanjutnya adalah menyiapkan perangkat lunak pada Raspberry Pi. Pustaka-pustaka Python yang diperlukan diimporkan terlebih dahulu, yaitu OpenCV (cv2) untuk pengolahan citra *realtime*, torch untuk memanggil model YOLOv5, dan numpy untuk pengolahan data numerik hasil deteksi. Model YOLOv5 yang sudah dilatih pada *dataset* khusus benih ikan nila kemudian dimuat ke dalam Raspberry Pi. Model dimuat dalam bentuk file *best.pt* menggunakan perintah Python sebagai gambar berikut pada gambar 4.16.

```
import torch

model = torch.hub.load('ultralytics/yolov5', 'custom', path='best.pt', force_reload=True)
```

Gambar 4.16 Inisialisasi Perangkat Lunak dan Pemanggilan Model YOLOv5

Perintah ini mengarahkan Raspberry Pi untuk menggunakan model hasil pelatihan khusus (*custom model*) yang hanya mengenali satu kelas objek, yaitu benih ikan nila, sehingga deteksi menjadi lebih spesifik dan akurat.

3. Pengambilan Data Citra secara *Realtime*

Pada Proses ini mendeteksi dimulai dengan menangkap citra secara *realtime* menggunakan *webcam*. *Webcam* diaktifkan melalui perintah pada gambar 4.17.

```
import cv2

cap = cv2.VideoCapture(0)
```

Gambar 4.17 Tampilan Webcam Aktif

Citra diambil secara berkelanjutan (*streaming*) dengan resolusi yang disesuaikan, misalnya 1920x1080 piksel, agar kualitas gambar cukup tajam namun tetap ringan untuk diproses secara *realtime* oleh Raspberry Pi. Kamera dipastikan dalam posisi stabil dan tidak bergeser selama proses penghitungan ikan berlangsung.

4. Proses Deteksi dan Perhitungan Jumlah Ikan

Pada tahap ini Citra yang ditangkap secara *realtime* kemudian dikirim ke model YOLOv5 untuk dianalisis. Model akan memproses citra *frame* per *frame*, mendeteksi objek ikan nila, memberikan *bounding box* pada setiap objek yang terdeteksi, dan menghitung jumlahnya secara otomatis. Proses ini dilakukan dalam sebuah *loop* berulang menggunakan skrip Python pada gambar 4.18 berikut ini.

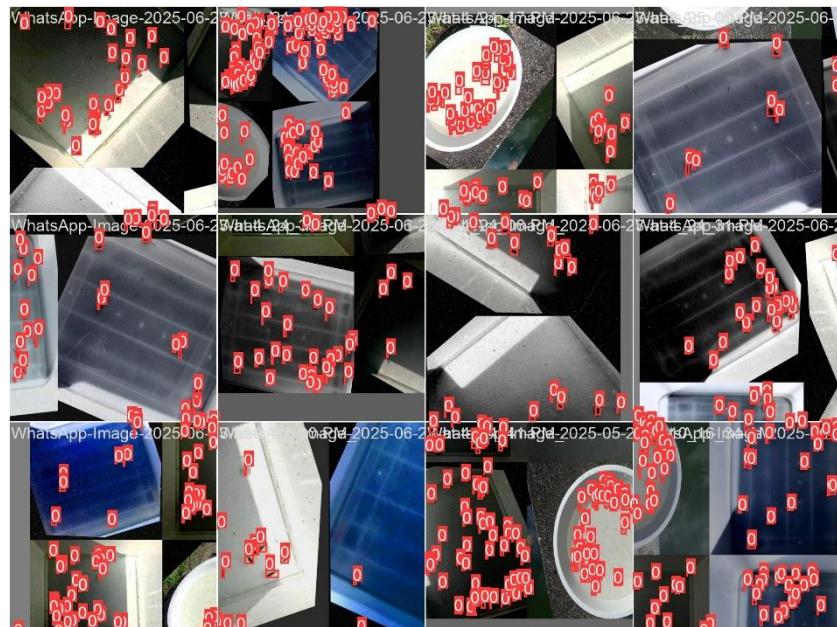
```
while True:
    ret, frame = cap.read()
    results = model(frame)
    annotated_frame = results.render()[0]
    jumlah_ikan = len(results.xyxy[0])
```

Gambar 4.18 Proses Deteksi dan Perhitungan Jumlah Ikan

Penjelasan dari gambar di atas untuk proses pendekripsi objek YOLOv5 dengan parameter jumlah ikan. Sebagai berikut contoh objek deteksi YOLOv5 dan jumlah ikan:

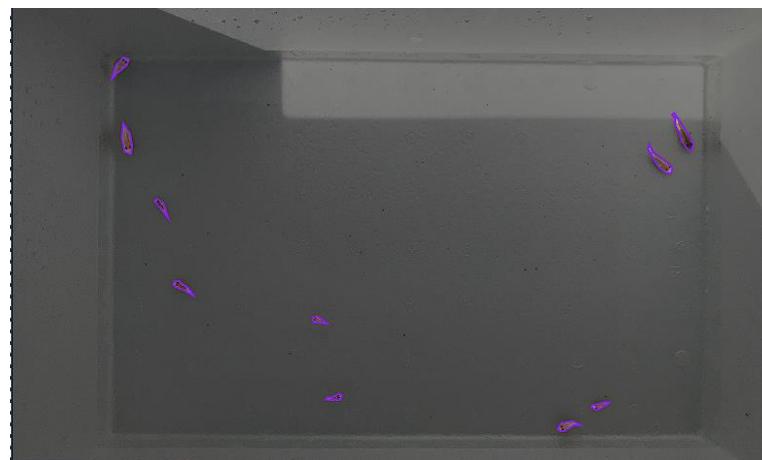
- Citra dibaca dari kamera secara berulang (per *frame*) menggunakan *cap.read()*.

Pada gambar di bawah ini.



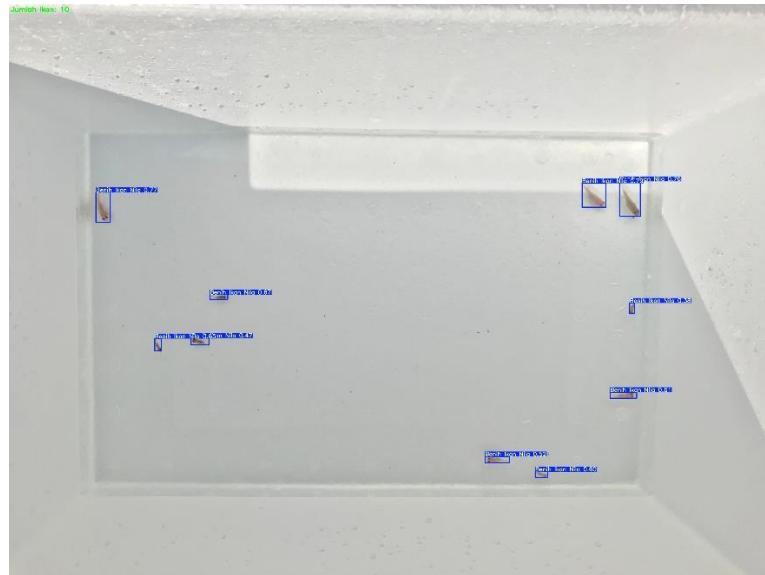
Gambar 4.19 Dataset Citra Per-frame

- Model YOLOv5 menganalisis citra dan memberikan prediksi lokasi objek ikan nila dalam bentuk koordinat *bounding box*. Pada gambar di bawah ini.



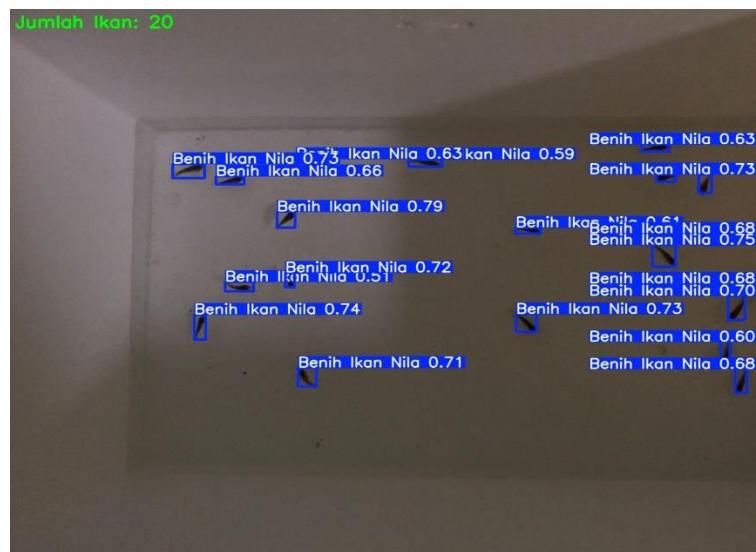
Gambar 4.20 Menganalisis YOLOv5

- c. *Bounding box* dan label ditampilkan secara otomatis pada citra hasil deteksi (`results.render()`). Pada gambar di bawah ini.



Gambar 4.21 *Bounding Box* Dan Label “Benih Ikan Nila”

- d. Jumlah ikan yang terdeteksi dihitung dengan menghitung banyaknya *bounding box* yang muncul pada setiap *frame* (`len(results.xyxy[0])`). Pada gambar di bawah ini.



Gambar 4.22 Jumlah Ikan dan *Bounding Box*

5. Integrasi Servo untuk Sortir Otomatis

Selain mendeteksi jumlah ikan, sistem juga dapat diintegrasikan dengan servo motor untuk melakukan sortir otomatis. Misalnya, ketika jumlah ikan yang terdeteksi dalam satu *batch* telah mencapai angka tertentu, servo akan membuka atau menutup pintu air dengan sampai dengan 50 benih ikan. Contoh skrip sederhana integrasi servo pada gambar 4.23 di bawah ini.

```
from gpiozero import Servo

servo = Servo(18)

if jumlah_ikan >= 50:
    servo.max() # Membuka pintu aliran ikan
else:
    servo.min() # Menutup pintu aliran ikan
```

Gambar 4.23 Integrasi Servo Otomatis

6. Penghentian Proses Deteksi

Proses deteksi *realtime* dapat dihentikan secara manual dengan menekan tombol tertentu (misalnya tombol ‘b’). Setelah proses selesai, kamera dilepaskan dan semua jendela tampilan ditutup untuk menghemat sumber daya Raspberry Pi

5. Gambar perintah kode program pada gambar 4.24.

```
cap.release()
cv2.destroyAllWindows()
```

Gambar 4.24 Proses *End* Layar Monitor

Setelah proses deteksi benih ikan nilai selesai dijalankan, program Python menggunakan dua fungsi penting untuk menghentikan seluruh aktivitas deteksi secara bersih. Fungsi `cap.release()` digunakan untuk melepaskan kembali akses kamera atau perangkat video capture yang sebelumnya digunakan oleh sistem. Hal ini bertujuan agar kamera tidak terus aktif dan dapat digunakan kembali oleh program lain tanpa terkunci. Sementara itu, fungsi `cv2.destroyAllWindows()`

berfungsi untuk menutup semua jendela tampilan yang dibuka oleh OpenCV selama proses deteksi *realtime*. Pemanggilan kedua fungsi ini sangat penting agar sumber daya perangkat keras dan tampilan grafis kembali normal setelah program dihentikan, sehingga Raspberry Pi atau komputer dapat menjalankan proses lain tanpa gangguan.

4.5 Pengujian Performa Alat

Pengujian performa alat, nilai akan diambil dengan menghitung nilai *confusion matrix*, *recall*, *precision*, serta *F1 score* pada pengujian deteksi objek benih ikan nila yang telah dilatih. Pengujian performa akan dilakukan menggunakan 115 gambar dari data hasil uji coba secara *real-time* menggunakan *Webcam* yang dimana deteksi dilakukan pada penampang berlatar putih dengan ukuran 28 x 16 cm yang merupakan tempat meletakkan objek benih ikan nila yang akan dideteksi. Pada pengujian ini kami menggunakan jarak yang telah ditentukan yaitu 40 cm dari penampang dimana sesuai dengan ukuran luas penampang. Pada jarak ini lah pendektsian mendapatkan hasil deteksi yang paling bagus. Oleh karena itu, jarak sangat mempengaruhi hasil pendektsian. Kami melakukan pengujian menggunakan kondisi air yang berbeda. Dan didapatkan hasil ketika kondisi air yang didapatkan baik maka hasil pembacaan akan lebih optimal. Sedangkan ketika kondisi air buruk maka akan terjadi pembacaan dan perhitungan objek benih ikan nila yang kurang akurat. Berikut ini adalah beberapa hasil dari pengujian secara *realtime* yang akan dipaparkan pada tabel 4.1.

Tabel 4.1. Hasil Pengujian Secara *Realtime*

AKTUAL

Benih Ikan Nila : 10

Kondisi Air : Jernih

**HASIL**

Jumlah Ikan: 10

Pintu Tertutup

LED Strip: Mati

Kekeruhan: 130 (Jernih)

NTU: 4.91

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 20

Kondisi Air : Jernih



Jumlah Ikan: 20

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 130 (Jernih)

NTU: 4.91

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 30

Kondisi Air : Jernih



Jumlah Ikan: 30

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 131 (Jernih)

NTU: 6.81

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 40

Kondisi Air : Jernih



Jumlah Ikan: 40

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 130 (Jernih)

NTU: 4.91

Toggle LED Strip

Screenshot GUI

Ambil Gambar

AKTUAL**HASIL**

Benih Ikan Nila : 50

Kondisi Air : Jernih



Jumlah Ikan: 50

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 130 (Jernih)

NTU: 4.91

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 10

Kondisi Air : Agak Keruh



Jumlah Ikan: 10

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 155 (Agak Keruh)

NTU: 52.57

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 20

Kondisi Air : Agak Keruh



Jumlah Ikan: 20

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 155 (Agak Keruh)

NTU: 52.57

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 30

Kondisi Air : Agak Keruh



Jumlah Ikan: 29

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 154 (Agak Keruh)

NTU: 50.67

Toggle LED Strip

Screenshot GUI

Ambil Gambar

AKTUAL

HASIL

Benih Ikan Nila : 40

Kondisi Air : Agak Keruh



Jumlah Ikan: 40

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 153 (Agak Keruh)

NTU: 48.76

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 50

Kondisi Air : Agak Keruh



Jumlah Ikan: 48

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 154 (Agak Keruh)

NTU: 50.67

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 10

Kondisi Air : Keruh



Jumlah Ikan: 9

Pintu Tertutup

LED Strip: Mati

Kekeruhan: 232 (Keruh)

NTU: 199.39

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 20

Kondisi Air : Keruh



Jumlah Ikan: 20

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 241 (Keruh)

NTU: 216.55

Toggle LED Strip

Screenshot GUI

Ambil Gambar

AKTUAL

HASIL

Benih Ikan Nila : 30

Kondisi Air : Keruh



Jumlah Ikan: 28

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 284 (Keruh)

NTU: 298.54

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 40

Kondisi Air : Keruh



Jumlah Ikan: 39

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 285 (Keruh)

NTU: 300.44

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Benih Ikan Nila : 50

Kondisi Air : Keruh



Jumlah Ikan: 46

Pintu Terbuka

LED Strip: Mati

Kekeruhan: 287 (Keruh)

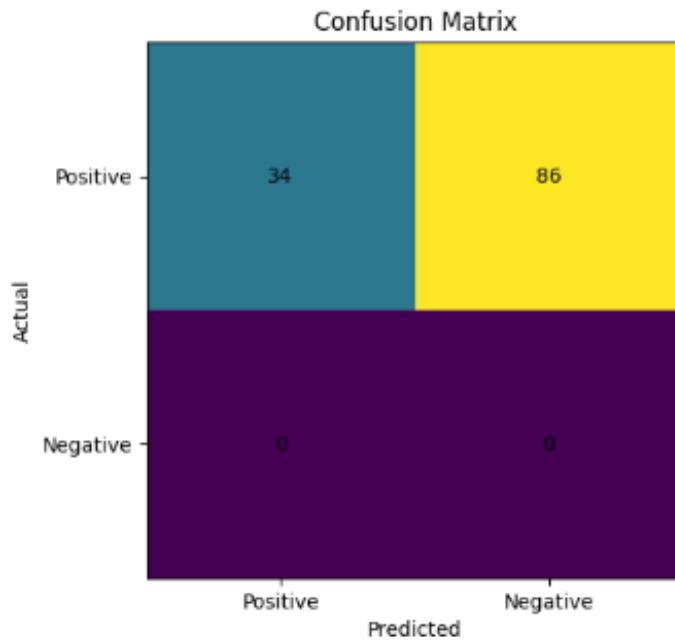
NTU: 304.26

Toggle LED Strip

Screenshot GUI

Ambil Gambar

Pada hasil pengujian diatas didapatkan hasil *output* pada layar monitor dimana objek yang terdeteksi yaitu benih ikan nila. Tulisan jumlah ikan pada kanan layar akan menampilkan output banyaknya benih ikan yang terdeteksi. Kemudian dibawahnya terdapat keterangan yaitu tingkat kekeruhan pada air yang dibagi menjadi 3 parameter yaitu jernih, agak keruh, dan keruh. Berdasarkan pada tabel pengujian diatas, maka akan dibuat *Confusion Matrix* hasil deteksi objek benih ikan nila seperti pada Gambar 4.25.



Gambar 4.25. *Confusion Matrix*

Pada gambar *Confusion Matrix* di atas menunjukkan bahwa sistem berhasil mendeteksi jumlah ikan dengan benar sebanyak 34 data (True Positive), namun masih terdapat 86 data (False Negative) di mana hasil deteksi webcam tidak sesuai dengan perhitungan manual. Tidak terdapat False Positive maupun True Negative, karena seluruh data pengujian berada pada kondisi positif (terdapat ikan). Hal ini menandakan bahwa sistem memiliki ketepatan deteksi yang baik saat hasilnya benar, tetapi masih perlu peningkatan agar mampu mendeteksi seluruh data secara konsisten. Maka akan dibuat *Confusion Matrix* hasil deteksi objek benih ikan nila seperti pada tabel 4.2.

Tabel 4.2. Hasil *Confusion Matrix*

<i>Confusion Matrix</i>		Aktual	
		Benih Ikan Nila	Background
Prediksi	Benih Ikan Nila	0.28	-
	Background	0.72	-
	Tidak Terdeteksi	2	-

Berdasarkan Tabel diatas dapat dicari nilai *Confusion Matrix*, yaitu nilai *True Positive*, *True Negative*, *False Positive*, *False Negative*, *Accuracy*, *Precision*, *Recall*, dan *F1 Score*. Berikut persamaan untuk mencari nilai-nilai tersebut.

Tabel 4.3. Nilai *Confusion Matrix*

Nama Ukuran	Simbol / Rumus	Keterangan
<i>True Positive (TP)</i>	Benih Ikan Nila → Benih Ikan Nila	Prediksi benar, sistem berhasil mengenali benih ikan nila sebagai benih ikan nila (34 data / 0,28)
<i>False Negative (FN)</i>	Benih Ikan Nila → <i>Background</i>	Prediksi salah, seharusnya benih ikan nila tetapi terdeteksi sebagai background (86 data / 0,72)
<i>True Negative (TN)</i>	<i>Background</i> → <i>Background</i>	<i>Background</i> dikenali dengan benar sebagai background (0 data)
<i>False Positive (FP)</i>	<i>Background</i> → Benih Ikan Nila	<i>Background</i> yang salah dikenali sebagai benih (0 data)
<i>Accuracy</i>	$TP + TN / (TP + TN + FP + FN)$	Akurasi total model dalam memprediksi objek (0,2833 / 28.33%)
<i>Precision</i>	$TP / (TP + FP)$	Ketepatan model saat memprediksi Benih Ikan Nila
<i>Recall / Sensitivity</i>	$TP / (TP + FN)$	Kemampuan model mendeteksi seluruh benih ikan nila (0,2833 / 28.33%)

<i>F1 Score</i>	$\frac{2 \times (Precision \times Recall)}{(Precision + Recall)}$	Rata-rata antara precision dan recall ($0.4416 / 44.16\%$)	harmonik
-----------------	---	--	----------

Berdasarkan tabel di atas, nilai True Positive (TP) sebesar 34 data menunjukkan bahwa sistem mampu mengenali benih ikan nila dengan benar sebagai benih ikan nila. Hal ini menandakan bahwa pada sebagian data, model memiliki kemampuan klasifikasi yang sesuai dengan kondisi aktual. Nilai False Negative (FN) sebesar 86 data menunjukkan bahwa masih terdapat kesalahan deteksi, di mana objek benih ikan nila seharusnya terdeteksi sebagai ikan, namun oleh sistem diklasifikasikan sebagai background. Kondisi ini mengindikasikan bahwa sistem masih belum optimal dalam mendekripsi seluruh objek benih ikan nila yang ada.

Sementara itu, nilai True Negative (TN) dan False Positive (FP) masing-masing bernilai 0. Hal ini disebabkan karena pada data pengujian tidak terdapat kondisi background murni, sehingga sistem tidak diuji dalam membedakan objek non-ikan secara langsung. Berdasarkan hasil perhitungan performa, diperoleh nilai accuracy sebesar 0,2833 atau 28,33%, yang menunjukkan bahwa tingkat ketepatan keseluruhan model dalam memprediksi objek masih tergolong rendah. Namun, nilai precision yang tinggi menunjukkan bahwa setiap prediksi benih ikan nila yang dihasilkan oleh sistem merupakan prediksi yang benar.

Nilai recall sebesar 0,2833 atau 28,33% mengindikasikan bahwa kemampuan sistem dalam mendekripsi seluruh benih ikan nila masih terbatas. Selanjutnya, nilai F1-score sebesar 0,4416 atau 44,16% menunjukkan keseimbangan antara precision dan recall, yang menggambarkan bahwa meskipun ketepatan prediksi tinggi, kemampuan model dalam menangkap seluruh objek masih perlu ditingkatkan.

BAB V

KESIMPULAN

3.1 Kesimpulan

Berdasarkan hasil pelaksanaan tugas akhir dan analisa dari proyek akhir yang berjudul “Implementasi Teknologi *Image Processing* untuk Alat Penghitung Benih Ikan Nila” ini, maka kesimpulan yang dapat diambil adalah sebagai berikut:

1. Perancangan alat penghitung benih ikan nila berhasil dilakukan dengan memanfaatkan teknologi *image processing* berbasis YOLOv5. Alat ini terdiri dari rangka konstruksi sederhana yang menopang kamera (*webcam*) serta wadah *box* sterofoam sebagai tempat benih ikan. Sistem dikembangkan menggunakan bahasa pemrograman Python dan dijalankan di perangkat laptop. Hasil perancangan menunjukkan bahwa alat mampu bekerja secara *real-time* dan dapat digunakan oleh pengguna secara praktis di lapangan.
2. Identifikasi dan deteksi benih ikan nila dalam gambar dilakukan melalui tahapan pengumpulan data citra, pelabelan objek, pelatihan model YOLOv5, dan proses deteksi. Model YOLOv5 terbukti mampu mendeteksi objek benih ikan dengan tingkat akurasi yang tinggi, meskipun objek bergerak dan memiliki ukuran kecil. Berdasarkan hasil pengujian, model mencapai nilai presisi sebesar 100%, *recall* 28,33%, akurasi 28,33%, dan *F1-score* 44,16%, yang menunjukkan bahwa sistem mampu mengidentifikasi objek benih ikan nila dengan sangat baik.
3. Untuk mengatasi kondisi benih ikan yang bergerak cepat, digunakan kamera dengan *frame rate* tinggi dan penempatan pada posisi yang strategis, serta kondisi air yang cukup agar objek tetap terlihat jelas dalam citra. Penggunaan algoritma YOLOv5 yang dirancang untuk deteksi objek *realtime* juga menjadi faktor penting dalam keberhasilan mendeteksi benih meskipun dalam kondisi bergerak. Selain itu, proses pra-pengolahan citra seperti penyesuaian

pencahayaan dan kontras turut membantu dalam meningkatkan visibilitas benih di depan kamera.

3.2 Saran

Dari hasil proyek akhir ini maka terdapat beberapa saran terkait dengan proyek ini, yaitu:

1. Penelitian ini masih berfokus pada satu jenis benih ikan yaitu benih ikan nila, sehingga diharapkan dapat dilakukan pengembangan kepada semua jenis benih ikan.
2. Penelitian ini masih belum bisa membedakan antara benih ikan nila dengan objek yang lain dikarenakan proses *labeling* yang hanya berfokus pada benih ikan nila saja sehingga penulis menyarankan untuk selanjutnya dapat dilakukan *labeling* terhadap objek lain, seperti kotoran ikan yang nantinya dapat terbaca oleh sistem.

DAFTAR PUSTAKA

- [1] Ardiyasa, I. W. (2015). Pemanfaatan Raspberry PI dan Webcam untuk Layanan Monitoring Ruangan Berbasis Web. Prosiding Konferensi Nasional Sistem dan Informatika (KNS&I).
- [2] Djunaedi, A., Pribadi, R., Hartati, R., Redjeki, S., Astuti, R. W., & Septiarani, B. (2016). Pertumbuhan ikan nila larasati (*Oreochromis niloticus*) di tambak dengan pemberian ransum pakan dan padat penebaran yang berbeda. Jurnal kelautan tropis, 19(2), 131-142.
- [3] Dendi, A. S., & Sunardi, S. (2021). ALAT PENGHITUNG BENIH IKAN LELE MENGGUNAKAN PENGOLAHAN CITRA (Doctoral dissertation, Politeknik Manufaktur Negeri Bangka Belitung).
- [4] Fang, W., Wang, L., & Ren, P. (2019). Tinier-YOLO: A real-time object detection method for constrained environments. Ieee Access, 8, 1935-1944.
- [5] Han, X., Chang, J., & Wang, K. (2021). Real-time object detection based on YOLO-v2 for tiny vehicle object. Procedia Computer Science, 183, 61-72.
- [6] Khumaidi, A., & Pradana, R. L. (2022). Identifikasi penyebab cacat pada hasil pengelasan dengan image processing menggunakan metode yolo. Jurnal Teknik Elektro dan Komputer TRIAC, 9(2), 107-112.
- [7] KHABIB, Y. N. (2019). RANCANG BANGUN IoT SMART FISH FARM DENGAN KENDALI RASPBERRY PI DAN WEBCAM (Doctoral dissertation, Universitas Muhammadiyah Ponorogo).
- [8] Nataliana, D., Syamsu, I., & Giantara, G. (2014). Sistem Monitoring Parkir Mobil menggunakan Sensor Infrared berbasis RASPBERRY PI. ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, 2(1), 68.
- [9] Purbowaskito, W., & Handoyo, R. (2017). Perancangan Alat Penghitung Benih Ikan Berbasis Sensor Optik. Jurnal Rekayasa Mesin, 8(3).
- [10] Rakhman, E., Candrasyah, F., & Sutera, F. D. (2014). Raspberry Pi-Mikrokontroler Mungil yang Serba Bisa. Yogyakarta: Penerbit Andi.
- [11] Samara, H., & Fardiyatulh, M. (2024). IMPLEMENTASI TEKNOLOGI IMAGE PROCESSING UNTUK IDENTIFIKASI

PENGUKURAN BOBOT IKAN HIAS DALAM PENENTUAN JUMLAH PAKAN (Doctoral dissertation, Politeknik Manufaktur Negeri Bangka Belitung).

- [12] Statistik, B. P. (2022). Statistik Produksi Perikanan Budidaya Indonesia 2020–2021. Jakarta: BPS.
- [13] PERIKANAN, S. P. K. (2018). Kementerian Kelautan dan Perikanan. Satu Data.[Internet].[diacu 2019 Desember 29]. Tersedia dari: <https://kkp.go.id/djp/ artikel/304-kkp-tegaskankinerja-neraca-perdagangan-ikan-kerapu-positif>.
- [14] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

LAMPIRAN



DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Fadillah Aslammy Pratama Sayti

Tempat & Tanggal Lahir : Sungailiat, 12 agustus 2004

Alamat : Jl.Pemuda Gg. Raya 1 No.28

Telp : -

Hp : +62 851 8075 7323

Email : punyafadilini@gmail.com

Jenis Kelamin : Laki laki

Agama : Islam



2. Riwayat Pendidikan

SD Negeri 10 Sungailiat 2010 – 2016

SMP Negeri 2 Sungailiat 2016 – 2019

SMA Negeri 1 Sungailiat 2019 – 2022

Politeknik Manufaktur Negeri Bangka Belitung 2022 – Sekarang

3. Pendidikan Non-Formal

-

Sungailiat, 08 Desember 2025

A handwritten signature in black ink, appearing to read "Fadillah Aslammy Pratama Sayti".

Fadillah Aslammy Pratama Sayti

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Muhammad Putra Deswalki
Tempat & Tanggal Lahir : Sungailiat, 9 Desember 2003
Alamat : Jl. Muhibin
Telp : -
Hp : +62 895 3404 09172
Email : kakang.deswal@gmail.com
Jenis Kelamin : Laki laki
Agama : Islam



2. Riwayat Pendidikan

SD Negeri 15 Sungailiat 2010 – 2016
SMP Negeri 2 Sungailiat 2016 – 2019
SMA Negeri 1 Sungailiat 2019 – 2022
Politeknik Manufaktur Negeri Bangka Belitung 2022 – Sekarang

3. Pendidikan Non-Formal

Sungailiat, 08 Desember 2025

Muhammad Putra Deswalki