

**IMPLEMENTASI TEKNOLOGI *IMAGE PROCESSING*
UNTUK IDENTIFIKASI PENGUKURAN BOBOT IKAN HIAS
DALAM PENENTUAN JUMLAH PAKAN**

PROYEK AKHIR

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



Disusun Oleh :

Harry Samara NIM : 1052014

Mauliana Fardiyatulh NIM : 1052017

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2024**

LEMBAR PENGESAHAN

IMPLEMENTASI TEKNOLOGI *IMAGE PROCESSING* UNTUK IDENTIFIKASI PENGUKURAN BOBOT IKAN HIAS DALAM PENENTUAN JUMLAH PAKAN

Oleh :

Harry Samara /1052014

Mauliana Fardiyatulh /1052017

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1 Pembimbing 2

 (Irwan, M.Sc., Ph.D)

 (Yuhni, M.T)

Penguji 1 Penguji 2

 (Aan Febriansyah, M.T)

 (Eko Sulistyoyo, M.T)

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa 1 : Harry Samara NIM : 1052014

Nama Mahasiswa 2 : Mauliana Fardiyatulh NIM : 1052017

Dengan Judul : Implementasi Teknologi *Image Processing* Untuk Identifikasi
Pengukuran Bobot Ikan Hias Dalam Penentuan Jumlah Pakan

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, 19 Januari 2024

Nama Mahasiswa

1. Harry Samara
2. Mauliana Fardiyatulh

Tanda Tangan

.....
.....

ABSTRAK

Ikan Mas Koki (Carassius Auratus) merupakan ikan hias yang banyak digemari oleh penggemar. Dalam pemeliharaan Ikan Mas Koki, pemberian pakan memegang peranan penting dalam mendukung pertumbuhan pada ikan sehingga pakan yang diberikan harus sesuai dengan ukuran, jumlah, dan kandungan nutrisi yang dibutuhkan oleh ikan tersebut. Salah satu cara untuk menentukan jumlah pakan ikan yang tepat adalah dengan melakukan pengukuran bobot dari ikan tersebut. Namun, pengukuran secara konvensional ketidakakuratan dapat terjadi karena beberapa faktor, seperti tingkat keahlian pengamat dan ikan yang cenderung bergerak. Tujuan dari penelitian ini adalah untuk membuat sistem yang dapat menentukan pemberian jumlah pakan pada Ikan Mas Koki sesuai dengan bobot dari ikan tersebut. Pada penelitian ini, Penulis memanfaatkan teknologi image processing menggunakan teknik segmentasi citra untuk melakukan pengukuran bobot ikan hias dalam penentuan jumlah pakan yang tepat. Berdasarkan hasil pengujian yang ditampilkan dengan serial monitor, sistem ini dapat melakukan pengukuran bobot dengan memasukkan Ikan Mas Koki secara bersamaan pada akuarium menghasilkan akurasi untuk panjang sebesar 93,8%, lebar 89%, dan bobot 93,78% dengan akurasi penentuan jumlah pakan sebesar 90,3% dan selisih pakan antara perhitungan secara manual dan sistem sebesar 1,3 gram. Kemudian, pengukuran bobot ikan/ekor menghasilkan nilai akurasi panjang 96,77%, lebar 80,05%, dan bobot 90,79% serta penentuan jumlah pakan sebesar 93,31% dengan selisih pakan manual dan sistem sebesar 0,88 gram. Sedangkan hasil pengujian yang ditampilkan menggunakan Bot Telegram, untuk tingkat akurasi bobot sebesar 93,12% dan pakan sebesar 93,73%.

Kata kunci: *Ikan Mas Koki, Image Processing, Segmentasi Citra*

ABSTRACT

The Goldfish (Carassius Auratus) is a popular ornamental fish among enthusiasts. In the care of Goldfish, proper feeding plays a crucial role in supporting their growth, requiring the right amount, size, and nutritional content of the provided feed. One method to determine the appropriate feeding amount is by measuring the weight of the fish. However, conventional measurement methods can be prone to inaccuracies due to factors such as the observer's skill level and the fish's tendency to move. The objective of this research is to develop a system capable of determining the optimal feeding amount for Goldfish based on their weight. In this study, the author utilizes image processing technology employing image segmentation techniques to measure the weight of ornamental fish for accurate determination of the feeding quantity. According to the test results displayed on the serial monitor, the system successfully measures the length with an accuracy of 93,8%, width 89%, and weight 93,78% when introducing Goldfish simultaneously into the aquarium. The accuracy in determining the feeding amount reaches 90,3%, and the feed difference between manual calculation and the system is 1.3 grams. Furthermore, individual fish weight measurement yields length accuracy of 96,77%, width 80,05%, and weight 90,79%, along with a feeding amount determination accuracy of 93,31%. The feed difference between manual and system calculation is only 0.88 grams. The test results using the Telegram Bot display a weight accuracy of 93,12% and a feeding accuracy of 93,73%.

Key words: *Goldfish, Image Processing, Image Segmentation*

KATA PENGANTAR

Puji dan syukur Penulis panjatkan atas kehadiran Allah SWT atas karunia dan limpahan rahmat-Nya, sehingga Penulis dapat menyelesaikan laporan proyek akhir ini yang berjudul **“Implementasi Teknologi *Image Processing* untuk Identifikasi Pengukuran Bobot Ikan Hias dalam Penentuan Jumlah Pakan”**. Shalawat serta salam selalu tersampaikan kepada Nabi Muhammad SAW yang telah membawa umat manusia ke dunia yang damai, terang dan penuh dengan ilmu pengetahuan. Tujuan Penulis membuat laporan proyek akhir ini adalah untuk memenuhi salah satu syarat kelulusan pendidikan Sarjana Terapan di Politeknik Manufaktur Negeri Bangka Belitung. Dalam laporan proyek akhir, penulis membahas tentang penelitian yang Penulis laksanakan selama proyek akhir berlangsung. Dengan adanya sistem ini diharapkan pemberian pakan pada ikan hias disesuaikan berdasarkan bobot dari ikan tersebut untuk mencegah keterlambatan dalam pertumbuhan.

Pada kesempatan ini, ucapan terima kasih juga disampaikan kepada pihak yang telah banyak membantu serta memberikan motivasi, saran, dan kritik yang tentunya sangat diharapkan dalam menyelesaikan proyek akhir ini. Berikut ini adalah pihak-pihak yang ikut membantu baik secara langsung maupun tidak langsung, diantaranya:

1. Orang tua dan keluarga Penulis yang telah banyak memberikan sumbangsih moril dalam bentuk doa dan dukungan kepada Penulis.
2. Bapak Irwan, M.Sc., Ph.D. selaku Wakil Direktur 1 dan dosen pembimbing 1 proyek akhir Penulis di Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Yudhi, M.T. selaku dosen pembimbing 2 proyek akhir penulis di Politeknik Manufaktur Negeri Bangka Belitung.
4. Bapak I Made Andik Setiawan, M.Eng., Ph.D. selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
5. Bapak Indra Dwisaputra, M.T. selaku Kepala Prodi D-IV Teknik Elektronika Politeknik Manufaktur Negeri Bangka Belitung.

6. Seluruh dosen dan PLP yang telah banyak membantu Penulis dalam menyelesaikan proyek akhir ini.
7. Teman-teman seperjuangan penulis, yaitu Dhea Vharisha, Lulu Mutialisa, Diah Ambarwati, dan Zulaika yang telah banyak membantu dan memberikan semangat sehingga Penulis dapat menyelesaikan laporan proyek akhir ini.
8. Seluruh pihak-pihak bersangkutan yang tidak dapat Penulis sebutkan satu per satu.
9. Terakhir, ucapan terima kasih Penulis berikan kepada diri sendiri yang telah berjuang hingga akhir untuk menyelesaikan proyek akhir ini dengan hasil yang maksimal walaupun selama proses penyelesaian banyak disertai dengan tangisan dan keringat yang bercucuran.

Penulis sangat berharap laporan akhir ini memberikan manfaat dalam rangka menambah wawasan dan ilmu pengetahuan yang terdapat dalam proyek akhir ini. Penulis menyadari bahwa dalam penulisan laporan akhir ini masih terdapat banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu, Penulis mengharapkan kritik dan saran yang membangun demi perbaikan laporan Penulis di masa depan yang akan datang. Mudah-mudahan laporan proyek akhir ini dapat dipahami dan bermanfaat untuk semua orang khususnya bagi pembaca laporan akhir ini.

Sungailiat, 19 Januari 2024

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	Error! Bookmark not defined.
PERNYATAAN BUKAN PLAGIAT	ii
ABSTRAK	iii
<i>ABSTRACT</i>	iv
KATA PENGANTAR.....	v
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Proyek Akhir.....	3
1.4 Batasan Masalah	3
BAB II DASAR TEORI.....	4
2.1 Ikan Mas Koki.....	4
2.2 Pengukuran Ikan Berdasarkan Citra	5
2.3 Perhitungan Bobot Berdasarkan Panjang dan Lebar Ikan.....	7
2.4 Pemberian Jumlah Pakan Ikan	8
2.5 <i>Image Processing</i>	8
2.5.1 Segmentasi Citra.....	8
2.6 <i>Visual Studio Code</i>	9
2.7 <i>Python</i>	10
2.7.1 <i>OpenCV</i>	10
2.8 <i>Telegram</i>	10
BAB III METODE PELAKSANAAN.....	12
3.1 Tahap Pelaksanaan	12
3.2 Studi Literatur	13

3.3 Pengembangan Sistem Pengukuran Bobot Ikan Hias	13
3.3.1 Pengumpulan Data Penelitian dan Desain Alat	13
3.3.2 Melakukan Kalibrasi pada <i>Webcam</i>	15
3.3.3 Pembuatan Program Pengaturan <i>Webcam</i>	15
3.3.4 Pembuatan Program Utama Sistem	16
3.4 Penentuan Jumlah Pakan Ikan Hias	17
3.5 Pembuatan <i>Bot Telegram</i>	18
3.6 Pengujian dan Evaluasi	18
3.7 Pembuatan Laporan Proyek Akhir dan Publikasi Ilmiah.....	19
BAB IV PEMBAHASAN	20
4.1 Deskripsi Alat	20
4.2 Pengembangan Sistem Pengukuran Bobot Ikan Hias	21
4.2.1 Pengumpulan Data Objek	21
4.2.2 Proses Kalibrasi <i>Webcam</i>	23
4.2.3 Proses Pengaturan <i>Webcam</i>	24
4.2.4 Proses Pengolahan Citra	29
4.2.5 Pengujian Pengukuran Bobot Ikan Hias	31
4.3 Pengujian Penentuan Jumlah Pakan.....	35
4.4 Pengujian Dengan Menggunakan <i>Bot Telegram</i>	39
4.4.1 Langkah-Langkah Menjalankan <i>Bot Telegram</i>	39
BAB V KESIMPULAN DAN SARAN	44
5.1 Kesimpulan	44
5.2 Saran	45
DAFTAR PUSTAKA.....	46
DAFTAR LAMPIRAN	49
Lampiran 1: Daftar Riwayat Hidup Penulis.....	49
Lampiran 2: <i>Source Code</i> Program Utama Ikan Mas Koki	51

DAFTAR TABEL

Tabel 3.1 Ukuran Pemberian Pakan Ikan	17
Tabel 4.1 Hasil Pengukuran Ikan Hias Secara Konvensional	21
Tabel 4.2 Pengujian 1 Secara Bersamaan Dengan <i>Serial Monitor</i>	32
Tabel 4.3 Pengujian 2 Ikan Hias/ Ekor Dengan <i>Serial Monitor</i>	34
Tabel 4.4 Pengujian 1 Jumlah Pakan Ikan Dengan <i>Serial Monitor</i>	36
Tabel 4.5 Pengujian 2 Jumlah Pakan Ikan Dengan <i>Serial Monitor</i>	37
Tabel 4.6 Hasil Pengujian Menggunakan <i>Bot Telegram</i>	42



DAFTAR GAMBAR

Gambar 2.1 Tampilan <i>Visual Studio Code</i>	9
Gambar 3.1 Diagram Alir Pelaksanaan	12
Gambar 3.2 Desain Alat Sistem Pengukuran	14
Gambar 3.3 Ukuran Akuarium	14
Gambar 3.4 Posisi Yang Diharapkan Ketika Pengujian	14
Gambar 3.5 Kalibrasi <i>Webcam</i>	15
Gambar 3.6 <i>Flowchart</i> Sistem Kerja Alat	16
Gambar 4.1 Rancangan Sistem.....	21
Gambar 4.2 Pengukuran Konvensional	22
Gambar 4.3 <i>Output</i> Kalibrasi <i>Webcam</i>	23
Gambar 4.4 <i>Grayscale</i>	29
Gambar 4.5 <i>Gaussian Blur</i>	30
Gambar 4.6 <i>Threshold</i>	30
Gambar 4.7 Hasil Pengujian Secara Sistem Pada <i>Serial Monitor</i>	31
Gambar 4.8 Hasil Pengujian Secara Sistem Pada <i>Bot Telegram</i>	31
Gambar 4.9 Aplikasi <i>Telegram</i> Pada <i>SmartPhone</i>	39
Gambar 4.10 Tampilan Menu <i>Telegram</i>	40
Gambar 4.11 Menu <i>Chat Bot</i>	40
Gambar 4.12 <i>Start</i> Pada <i>Bot</i>	41
Gambar 4.13 <i>Stop</i> Pada <i>Bot</i>	41

DAFTAR LAMPIRAN

Lampiran 1: Daftar Riwayat Hidup Penulis	49
Lampiran 2: <i>Source Code</i> Program Utama Ikan Mas Koki.....	51



BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam pemeliharaan Ikan Mas Koki (*Carrasius Auratus*) terdapat beberapa kendala yang dapat terjadi, salah satunya adalah laju pertumbuhan ikan yang cenderung lambat. Upaya yang dapat dilakukan untuk mengatasi hal tersebut dengan melakukan pemberian pakan yang tepat sesuai dengan ukuran, jumlah dan kandungan nutrisi yang dibutuhkan oleh ikan tersebut. Kelebihan dalam pemberian pakan dari kebutuhan Ikan Mas Koki dapat menghambat pertumbuhan mereka. Pemberian pakan yang optimal pada ikan akan menghasilkan pertumbuhan yang baik. Namun, jika pakan yang diberikan secara berlebihan dapat mengakibatkan kerja enzim pada pencernaan ikan akan terganggu. Pertumbuhan ikan akan terganggu jika protein pada ikan terlalu berlebihan dikarenakan protein yang seharusnya terserap dalam tubuh banyak digunakan sebagai sumber energi dalam proses metabolisme, dan sebagian besar berubah menjadi ammonia yang dibuang oleh ikan [1].

Salah satu cara untuk menentukan jumlah pakan yang tepat adalah dengan melakukan pengukuran bobot dari ikan hias. Data ukuran ikan berperan penting dalam menentukan umur, parameter pertumbuhan, dan tingkat kematian populasi ikan. Dengan melakukan pengukuran ikan, tahap perkembangan dari ikan akan mudah dipantau dan dapat menentukan jumlah pakan yang sesuai. Dalam metode konvensional, ketidakakuratan pengukuran ikan memang dapat terjadi akibat beberapa faktor, seperti tingkat keahlian, penglihatan, dan arah pengamat [2].

Dalam penelitian R. W. Tri Hartono dkk, yang berjudul “Implementasi Algoritma *Canny Edge Detection* untuk Identifikasi Panjang dan Berat Ikan Koi Saat Bergerak” menjelaskan bahwa akurasi pengukuran menggunakan cara elektronik yang memanfaatkan teknologi *image processing* lebih baik dibandingkan cara konvensional. Hal ini karena dengan cara elektronik dapat dilakukan walaupun ikan sedang bergerak, sedangkan dengan cara konvensional, ikan harus dalam posisi

diam. Dari hasil pengukuran terhadap 10 Ikan Koi yang diukur masing-masing 5 kali dengan cara konvensional, terdapat selisih data yang cukup signifikan kesalahannya dibandingkan diukur secara elektronik sebanyak 867 kali. Hasil dari pengukuran konvensional panjang dan konversi berat ikan memiliki *margin error* 4,77% dan 6,3%, sedangkan elektronik 0,18% dan 1,36%. Pengukuran secara elektronik juga dari segi waktu rata-rata 107,23 lebih cepat dari konvensional [3]. Penelitian lainnya oleh Akhmad Qashlim dkk, melakukan pengukuran bobot ikan bandeng dengan menggunakan teknologi *image processing* dengan sistem *computer vision*. Penelitian ini dilakukan dengan menentukan pola pengukuran untuk menemukan parameter (HSVHS) dan menggunakan deteksi *contour* untuk mendapatkan tepi objek sebagai kotak pembatas. Hasil penelitian menunjukkan akurasi pengukuran bobot menggunakan 20 sampel ikan mencapai 86,66% dan akurasi pendeteksian objek sebesar 95% dengan tingkat kesalahan pengukuran sebesar 1 cm dari panjang seharusnya [4].

Berdasarkan keterangan di atas, perlu adanya sistem yang dapat membantu menentukan pemberian pakan yang sesuai berdasarkan bobot ikan hias. Oleh karena itu, proyek akhir kami yang berjudul **“Implementasi Teknologi *Image Processing* untuk Identifikasi Pengukuran Bobot Ikan Hias Dalam Penentuan Jumlah Pakan”** ini bertujuan untuk membantu dalam mengidentifikasi ikan dengan beragam ukuran dan berat menggunakan teknik segmentasi citra dengan *output* berupa panjang, lebar, bobot, dan jumlah pakan Ikan Mas Koki yang ditampilkan pada *serial monitor* dan *Telegram*. Dengan demikian, sistem ini dapat memberikan pakan kepada ikan hias dengan takaran yang tepat.

1.2 Rumusan Masalah

Rumusan masalah yang terdapat pada penelitian ini, yaitu:

1. Bagaimana merancang sistem yang dapat mengukur bobot ikan hias secara akurat?
2. Bagaimana menentukan pemberian pakan yang efisien dan sesuai dengan kebutuhan ikan hias?

1.3 Tujuan Proyek Akhir

Adapun tujuan dari penelitian ini, yaitu:

1. Mengimplementasikan teknologi *image processing* untuk identifikasi pengukuran bobot ikan hias.
2. Mengefisienkan pemberian pakan sesuai dengan bobot ikan hias.

1.4 Batasan Masalah

Agar masalah dalam penelitian ini lebih terarah dan sesuai dengan apa yang sudah direncanakan dan mempermudah mendapatkan data serta informasi yang diperlukan, maka ditetapkan beberapa batasan masalah, yaitu:

1. Lingkup penelitian terbatas hanya dalam akuarium dengan jumlah objek penentuan bobot minimal lima ekor.
2. Pengukuran ikan hias dilakukan secara sistem (menggunakan *image processing*) dan manual (menggunakan penggaris dan alat timbang) dengan toleransi pengukuran kurang lebih 5%.
3. Sistem kerja hanya dilakukan kepada objek ikan yang berada di akuarium tanpa ada objek selain ikan karena sistem bersifat *universal* (dapat mendeteksi objek selain ikan).
4. *Output* yang ditampilkan pada *Telegram* merupakan hasil *capture* citra secara *realtime* oleh *webcam*.

BAB II

DASAR TEORI

2.1 Ikan Mas Koki

Ikan Mas Koki (*Carassius auratus*) merupakan ikan hias yang cukup populer dan banyak diminati penggemar. Yang menjadi daya tarik dari ikan ini adalah kualitas warna yang dimilikinya. Ikan Mas Koki dengan jenis oranda memiliki warna yang beragam, mulai dari kuning pudar hingga merah. Dalam pemeliharaan Ikan Mas Koki, pemberian pakan memegang peranan penting dalam mendukung pertumbuhan, mempertahankan kecerahan warna dan pembentukan sirip pada ikan menjadi lebih sempurna [5].

Dalam penelitian Risdawati dan Irawati Mei Widiastuti menjelaskan bahwa pemberian pakan yang optimal akan menghasilkan pertumbuhan yang baik. Pemberian pakan yang berlebihan dapat mengakibatkan terganggunya kerja enzim pada pencernaan ikan. Pertumbuhan ikan akan terganggu jika protein pada ikan terlalu berlebihan dikarenakan protein yang seharusnya terserap dalam tubuh banyak digunakan sebagai sumber energi dalam proses metabolisme dan juga akan terbuang menjadi ammonia [1].

Untuk mengoptimalkan pemberian pakan pada ikan dapat dilakukan dengan cara melakukan pengukuran panjang dan lebar ikan untuk mendapatkan bobot dari ikan tersebut. Pada penelitian yang dilakukan oleh Raihan Islamadina dkk, menjelaskan bahwa pengukuran panjang dan lebar ikan secara manual kurang efektif dan membutuhkan waktu yang lama, terutama jika ikan dalam jumlah yang banyak. Di penelitian ini dengan menggunakan teknik pengolahan citra dapat melakukan pengukuran panjang dan lebar ikan secara otomatis. Hasil yang didapatkan dari menggunakan sistem menghasilkan nilai yang hampir mendekati nilai ukur manual. Tingkat akurasi dari sistem pengolahan citra mampu menghasilkan nilai untuk panjang dan lebar sebesar 80% hingga 90% [6].

Pada penelitian lainnya oleh R. W. Tri Hartono dkk, yang mengidentifikasi panjang dan berat Ikan Koi menggunakan metode *canny edge detection*

mendapatkan hasil simpulan bahwa tingkat akurasi pengukuran menggunakan cara elektronik lebih baik dibandingkan cara konvensional. Pengukuran secara elektronik dapat dilakukan walaupun ikan dalam kondisi bergerak, sedangkan dengan konvensional, ikan harus dalam posisi diam. Hasil pengukuran panjang dan berat ikan secara konvensional masing-masing memiliki *margin error* 4,77% dan 6,3%, sementara dengan cara elektronik 0,18% dan 1,36% [3].

2.2 Pengukuran Ikan Berdasarkan Citra

Data ukuran ikan berperan penting dalam memantau dan menentukan umur, parameter pertumbuhan, dan tingkat kematian populasi ikan. Dengan melakukan pengukuran ikan berguna untuk mengawasi perkembangan ikan dan menentukan jumlah pakan yang diperlukan. Dengan metode konvensional, ketidakakuratan dalam pengukuran ikan memang dapat terjadi akibat beberapa faktor, seperti keahlian, ketajaman mata dan arah pandangan pengamat [2].

Berdasarkan penelitian Chi-Hsuan Tseng dkk, yang melakukan pengukuran panjang tubuh ikan menggunakan metode *Convolutional Neural Network* (CNN), terdapat beberapa langkah-langkah yang dilakukan untuk mengetahui panjang tubuh ikan, yaitu (1) pengumpulan gambar dan *patch* pelatihan, terdapat 9.000 data gambar yang akan digunakan. Lima ribu gambar digunakan untuk mengembangkan pengklasifikasi CNN dan empat ribu gambar sisanya untuk menilai kinerja pengklasifikasi CNN yang dikembangkan. Dari gambar pelatihan untuk mengembangkan pengklasifikasi CNN dipotong menjadi beberapa *patch*. Terdapat 30.000 *patch* dibuat dengan 4.000 kepala ikan, 4.000 ekor, 4.000 badan ikan, 4.000 pelat warna, dan 14.000 *background*. *Patch* diubah ukurannya menjadi 36 – 36 piksel, (2) arsitektur pengklasifikasi CNN, pengklasifikasi CNN dikembangkan untuk membedakan *patch* dari lima kelas tersebut. Pengklasifikasi CNN dilatih menggunakan *backpropagation* dengan 100 *epoch*. Pada setiap *epoch*, sampel pelatihan disusun menjadi 195 *batch* dengan ukuran *batch* 128. Filter, *saliency maps*, dan Grad-CAM dari pengklasifikasi CNN yang dikembangkan divisualisasikan untuk mendemonstrasikan fitur yang dipelajari pengklasifikasi CNN dari gambar pelatihan. Hasil yang didapatkan dari

pengklasifikasi CNN dengan akurasi keseluruhan sebesar 98,78%. Dalam memperkirakan panjang ikan dengan menggunakan metode CNN, *Mean Absolute Error* (MAE) dan *Mean Absolute Relative Error* (MARE) yang terjadi sebesar 5,36 cm dan 4,46% [7].

Pada penelitian lainnya oleh Raihan Islamadina dkk, yang berjudul “Estimasi Panjang dan Lebar Ikan Berdasarkan *Visual Capture*”. Tahapan pada penelitian ini dimulai dengan melakukan pengukuran panjang dan lebar secara manual pada lima ekor ikan yang dijadikan sampel. Selanjutnya pada sistem, melakukan proses pengambilan gambar dari objek ikan menggunakan kamera digital. Gambar tersebut akan masuk ke tahap *pre-processing* untuk mengubah citra berwarna menjadi citra *grayscale*. Citra *grayscale* akan di segmentasikan objeknya untuk memisahkan objek-objek yang tidak dibutuhkan dalam suatu citra ikan dan menghilangkan bagian yang tidak penting dari objek tersebut. Kemudian, melakukan proses *calculation* berupa nilai panjang dan lebar dari sampel citra ikan yang diukur secara manual. Selanjutnya, nilai yang didapatkan dari pengukuran manual dihitung nilai kalibrasi dari masing-masing panjang dan lebar sampel citra ikan. Nilai kalibrasi dari sampel citra ikan akan dihitung nilai kalibrasi rerataan sebagai acuan untuk mengukur citra uji. Hasil keakuratan dengan pengukuran sistem untuk panjang dan lebar sebesar 80% hingga 90% [6].

Nor Salwa Damahuri dkk, dalam penelitiannya yang melakukan pengukuran panjang otomatis Ikan Nila berdasarkan teknik pengolahan citra dan di kombinasikan dengan *Graphical User Interface* (GUI). Tahapannya dimulai dengan pengambilan gambar sebanyak 25 gambar dari lima sampel ikan yang berbeda dengan 2 jarak yang berbeda, yaitu 20 cm dan 25 cm. Dimensi gambar digital adalah 4032 x 3024 piksel dan panjang focus 4 mm dengan format citra digital adalah format file JPEG. Kemudian, pada tahap *pre-processing* digunakan untuk membuka kotak dialog yang berisi daftar file di folder. Spesifikasi file JPEG digunakan untuk mendapatkan citra data input ke sistem. Di proses selanjutnya, segmentasi citra untuk memisahkan citra berwarna ke *grayscale* dengan menggunakan *rgb2gray* pada algoritma pengolahan citra. Citra *grayscale* akan di konversi menjadi citra biner dengan teknik *thresholding*. Citra biner akan

di ekstraksi fitur untuk mendeteksi bentuk ikan. *Graphical User Interface* (GUI) digunakan untuk mengunggah gambar Ikan Nila dan menghitung panjang Ikan Nila secara otomatis. Hasil pengukuran Ikan Nila dengan jarak 20 cm diperoleh *Mean Absolute Error* (MAE) 1,03 cm dan *Mean Relative Error* (MRE) 8,50%. Akurasi yang di dapatkan sebesar 91,50%. Pada pengukuran panjang jarak 25 cm, analisis kinerja *Mean Absolute Error* (MAE) dan *Mean Relative Error* (MRE) masing-masing 1,36 cm dan 10,40% dengan tingkat akurasi sebesar 89,49% [2].

2.3 Perhitungan Bobot Berdasarkan Panjang dan Lebar Ikan

Dalam penelitian yang dilakukan oleh Massimo Lorenzoni dkk, yang melakukan analisis terhadap ciri biologis Ikan Mas Koki (*Carassius Auratus*). Penelitian ini dilakukan dengan menganalisis struktur populasi, kemudian setiap kelompok diidentifikasi berdasarkan tahun perkawinan masing-masing specimen, bukan berdasarkan kelas umur. Hasil dari penelitian ini dengan menggunakan 3.111 specimen Ikan Mas Koki sebagai sampel mendapatkan panjang tubuh (TL), berat (W), dan umur specimen yang dianalisis bervariasi antara 4,30 dan 40,60 cm, 1 dan 1,137 g, serta 0,2 dan 7,9 tahun. Ikan Mas betina terbukti lebih besar dan lebih tua dengan rata-rata (\pm SE) panjang tubuh (TL) = $23,24 \pm 0,18$ cm, berat (W) = $354,03 \pm 6,69$ g, umur = $3,08 \pm 0,03$ tahun, dibandingkan dengan jantan (TL = $15,69 \pm 0,63$ cm, W = $96,73 \pm 14,15$ g, umur = $2,27 \pm 0,14$ tahun). Adapun rumus yang digunakan untuk menghitung hubungan antara panjang tubuh (TL) dan berat (W) dalam penelitian ini adalah [8] :

$$W = a.T.L^b$$

Keterangan :

W = Berat Ikan Mas Koki dalam gram

a = 0,015

T = Lebar Ikan Mas Koki

L = Panjang Ikan Mas Koki

b = 3,062

Rumus diatas digunakan oleh Penulis dalam menghitung bobot Ikan Mas Koki dalam penelitian ini dikarenakan objek yang digunakan dalam penelitian sama.

2.4 Pemberian Jumlah Pakan Ikan

Menurut Erlangga dkk, pakan adalah makanan atau asupan yang dicerna oleh hewan ternak atau hewan peliharaan yang berguna sebagai sumber energi dan materi untuk pertumbuhan dan kehidupan organisme. Penyesuaian konsumsi pakan ikan terdiri dari pengaturan energi yang diberikan sehingga jumlah pakan yang dikonsumsi disesuaikan dengan laju metabolismenya. Adapun akibat jika pemberian pakan berlebih (*overfeeding*) yaitu berpengaruh pada peningkatannya kadar ammonia dan nitrat, pH air pada akuarium menjadi rendah, serta gangguan pencernaan pada ikan itu sendiri yang dapat mengganggu proses metabolismenya [9]. Pada penelitian yang dilakukan oleh Muhammad Thoifur Ibnu Fajar, menjelaskan bahwa jumlah pakan yang tepat antara 3% - 5% dari pakan pellet formulasi HiProVite (-2), tipe apung dapat memberikan pertumbuhan dan tingkat kelangsungan hidup ikan secara maksimal. Penelitian tersebut bertujuan untuk mengidentifikasi jumlah pakan yang tepat dengan konsentrasi pakan 3%, 4%, dan 5% dalam pertumbuhan panjang dan bobot ikan mas [10].

2.5 Image Processing

Image Processing merupakan sebuah pemrosesan citra dengan menggunakan komputer untuk membuat kualitas citra menjadi lebih baik yang sesuai dengan keinginan pengguna. *Image processing* bertujuan untuk meningkatkan kualitas citra agar lebih mudah dipahami baik oleh manusia maupun komputer, dimana proses dilakukan dengan masukan berupa citra dan keluaran berupa citra juga [11]. Citra tersebut dapat berupa gambar diam (foto) ataupun video yang didapat dari *webcam*. Secara matematis, citra adalah representasi kontinu dari intensitas cahaya dalam bidang dua dimensi. Citra harus direpresentasikan secara numerik dengan nilai dikrit agar dapat diproses oleh komputer digital [12].

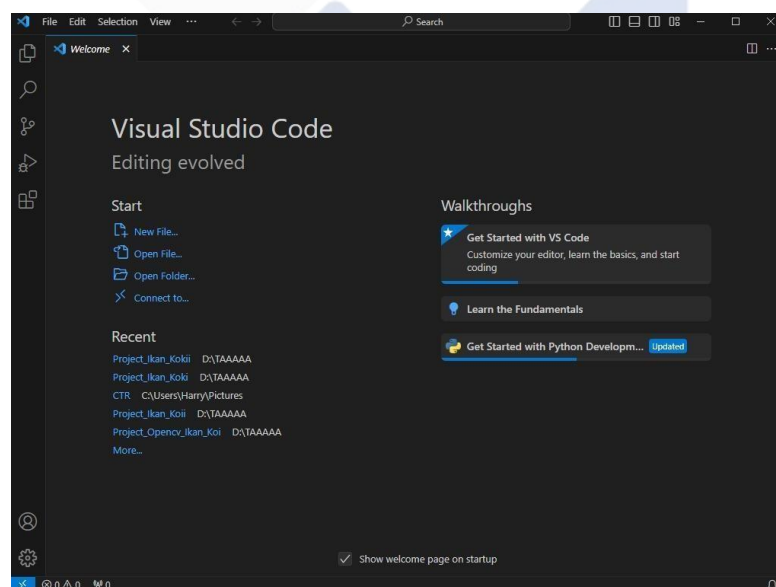
2.5.1 Segmentasi Citra

Segmentasi citra adalah salah satu tahapan dalam *image processing*. Tahap ini bertujuan untuk membagi citra menjadi beberapa bagian utama yang berisi informasi penting. Misalnya, memisahkan objek dengan latar belakang [13].

Segmentasi bukanlah proses tunggal yang dilakukan dalam *image processing*. Pada proses segmentasi, objek yang menjadi target akan diambil untuk proses selanjutnya. Pada teknik segmentasi citra terdapat dua karakteristik nilai derajat kecerahan citra, yaitu *discontinuity* dan *similarity*. Pada *discontinuity*, citra dibagi berdasarkan perubahan yang signifikan dari derajat kecerahannya, *discontinuity* ini biasanya diterapkan pada garis, deteksi tepi, area, dan sisi citra. Sedangkan *similarity*, citra dibagi berdasarkan *thresholding*, *region growing* dan *region spiltting* dan *merging*, yang diterapkan untuk citra statis dan dinamis [14].

2.6 Visual Studio Code

Visual studio merupakan salah satu aplikasi yang dibuat oleh perusahaan Microsoft untuk menulis kode program yang bersifat *code open source*. Aplikasi *visual studio code* dapat beroperasi pada sistem operasi *Windows*, *Mac OS* dan sistem operasi lainnya. Aplikasi ini sangat memudahkan pengguna untuk menulis kode program dan dapat digunakan untuk bahasa pemrograman seperti bahasa pemrograman *C#*, *C++*, *PHP*, *Phython*, *Java*, dan *GO* [15].



Gambar 2.1 Tampilan *Visual Studio Code*

Dalam penelitian ini, *visual studio code* digunakan untuk memudahkan dalam penulisan *code* dengan bahasa pemrograman *python*.

2.7 Python

Python adalah bahasa pemrograman yang dibuat oleh seorang *programmer* Belanda bernama Guido Van Rossum. Bahasa pemrograman *python* sangat interpretatif multiguna yang berfokus pada tingkat keterbacaan kode dengan filosofi perancangan. *Python* ditentukan sebagai bahasa yang menggabungkan kemampuan dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif [16]. Kelebihan dari *python* ialah mudah dipelajari, mudah untuk menjalankan program dengan fungsi yang kompleks, kode yang digunakan lebih sedikit, dan program yang sangat kompleks mudah dimodifikasi. Untuk kekurangannya sendiri, *python* berjalan sangat lambat, tidak adanya dukungan untuk *Android* dan *iOS*, akses *database* yang dimiliki terbatas, dan tidak cocok untuk bekerja dengan banyak inti/multiprosesor. *Python* banyak diterapkan di berbagai sistem operasi seperti *Microsoft Windows*, *Linux*, *Mac OS*, *Symbian OS*, *Android*, dan *Amiga Palm* [12].

2.7.1 OpenCV

OpenCV merupakan *library* yang difungsikan untuk pemrograman yang ditujukan ke *computer vision*. Terdapat lebih dari 2500 algoritma optimasi yang dapat digunakan untuk memproses dan mengedit *image* menggunakan *OpenCV*. *OpenCV* mendukung berbagai bahasa pemrograman seperti C, C++, *Python*, *Java*, dan PHP. Ada beberapa aplikasi *OpenCV* yang bisa diterapkan, yaitu interaksi manusia komputer (HCI), Identifikasi dan Pengenalan Objek, Deteksi dan Pengenalan Wajah, Pengenalan Gestur, Proses pada *Image*, *Tracking* Gerakan, dan *Mobile Robotics* [17].

2.8 Telegram

Telegram Messenger merupakan aplikasi layanan pesan seperti *Whatsapp*, *Line*, dan lainnya. Salah satu keunggulan *Telegram* dibandingkan aplikasi layanan pesan lainnya ialah *Telegram Bot*. *Telegram* menyediakan fitur yang dapat membuat *Bot* pada aplikasinya. Fitur *Bot* tersebut dapat terintegrasi langsung dengan berbagai layanan melalui *internet* [18]. Pada *Bot*, *user* dapat mengirimkan

pesan, perintah, dan *inline request*. *Bot* dapat dikontrol dengan menggunakan HTTPS ke API *Telegram*. Banyak kegunaan dari *Bot Telegram* antara lain [19]:

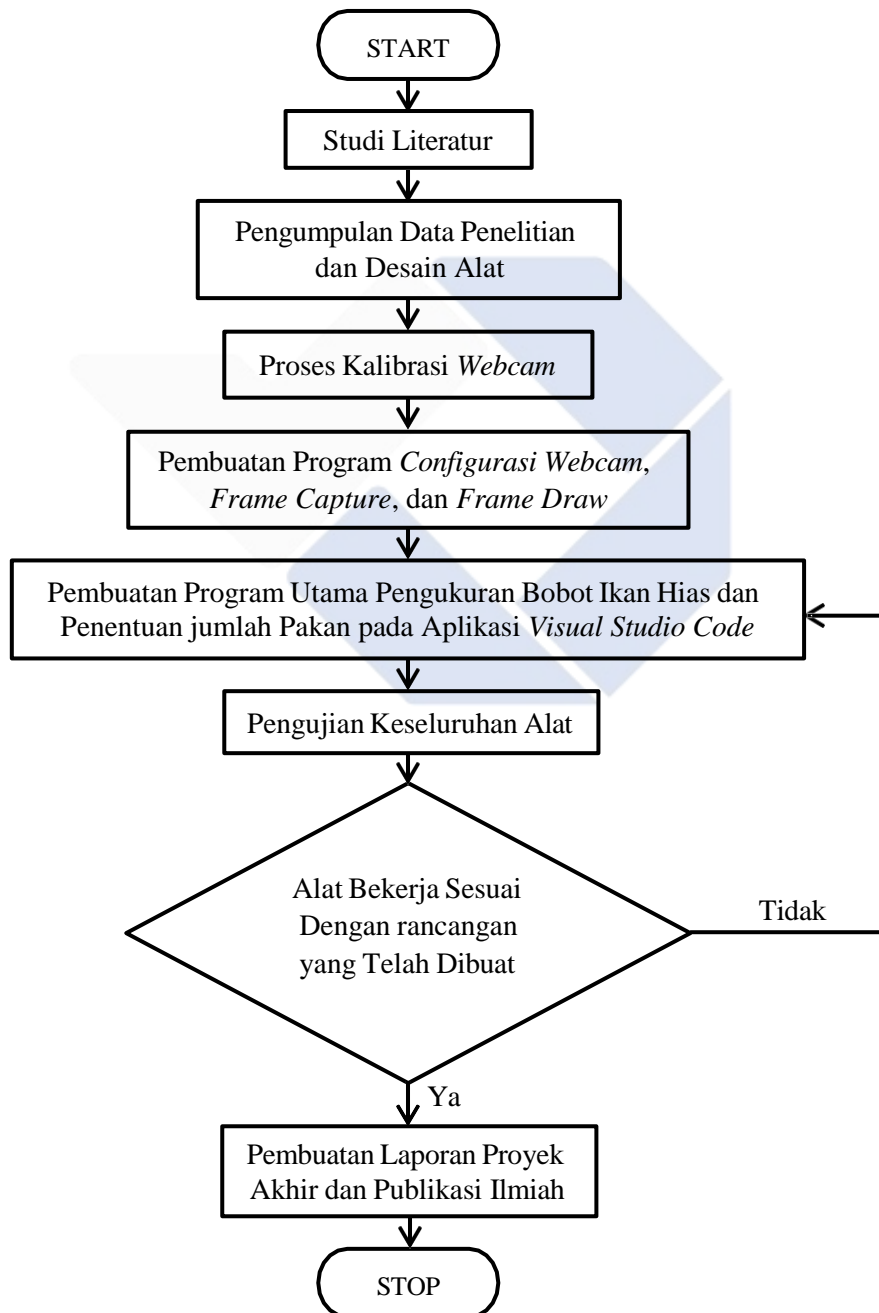
1. Dapat digunakan sebagai Koran pintar (*smart newspaper*) yang dapat memberikan berita kepada pelanggan *Bot* tersebut.
2. Dapat digunakan untuk mengakses layanan lain seperti *Gmail*, Gambar, GIF, IMDB, *Wiki*, Musik, *Youtube*, dan *GitHub*.
3. Dapat digunakan untuk menerima pembayaran layanan berbayar.
4. Dapat digunakan sebagai alat khusus seperti memberikan peringatan, ramalan cuaca, terjemahan, pemformatan, atau layanan lainnya.
5. Dapat digunakan sebagai *game* baik *single-player* ataupun *multi-player*.
6. Dapat juga digunakan sebagai layanan sosial untuk menghubungkan orang yang mencari mitra percakapan berdasarkan minat atau kedekatan yang sama.

Pada penelitian ini, *Bot* digunakan oleh Penulis sebagai alat yang menampilkan hasil *output* yang diterima dari *webcam* sekaligus dapat digunakan untuk *monitoring* objek dalam penelitian.

BAB III METODE PELAKSANAAN

3.1 Tahap Pelaksanaan

Metode dan langkah yang digunakan dalam penyelesaian proyek akhir ini ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram Alir Pelaksanaan

3.2 Studi Literatur

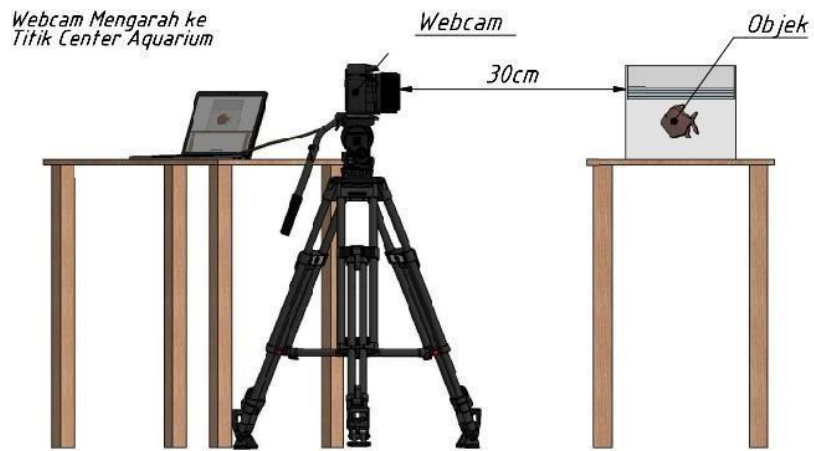
Pada tahapan ini telah dilakukan studi terhadap pustaka-pustaka yang relevan dengan proyek akhir. Beberapa literatur terkait dengan pengukuran panjang tubuh ikan seperti pada jurnal Chi-Hsuan Tseng dkk, tentang pengukuran panjang tubuh ikan secara otomatis menggunakan metode *Convolutional Neural Network* (CNN) [7]. Judul “Estimasi Panjang dan Lebar Ikan Berdasarkan *Visual Capture*” pada jurnal Raihan Islamadina dkk [6]. Penelitian oleh Nor Salwa Damahuri dkk, tentang sistem pengukuran panjang ikan nila secara otomatis berdasarkan teknik pengolahan citra [2]. Dalam penelitian R. W. Tri Hartono dkk, yang berjudul “Implementasi Algoritma *Canny Edge Detection* untuk Identifikasi Panjang dan Berat Ikan Koi Saat Bergerak”[3]. Penelitian Akhmad Qashlim dkk, tentang estimasi bobot ikan bandeng berdasarkan teknologi *image processing* [4]. Selain dari jurnal-jurnal terkait, penulis mengumpulkan referensi-referensi dari laporan akhir tingkat atas sebelumnya dan dari internet.

3.3 Pengembangan Sistem Pengukuran Bobot Ikan Hias

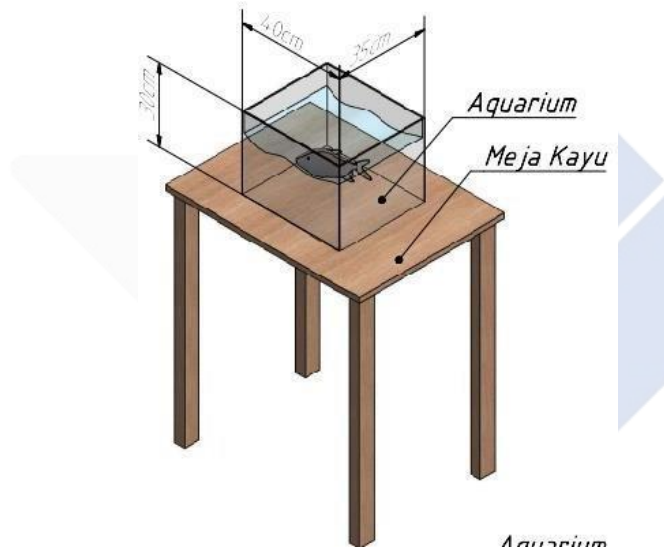
Dalam tahap ini dilakukan beberapa proses dalam pengembangan sistem pada proyek akhir ini, antara lain:

3.3.1 Pengumpulan Data Penelitian dan Desain Alat

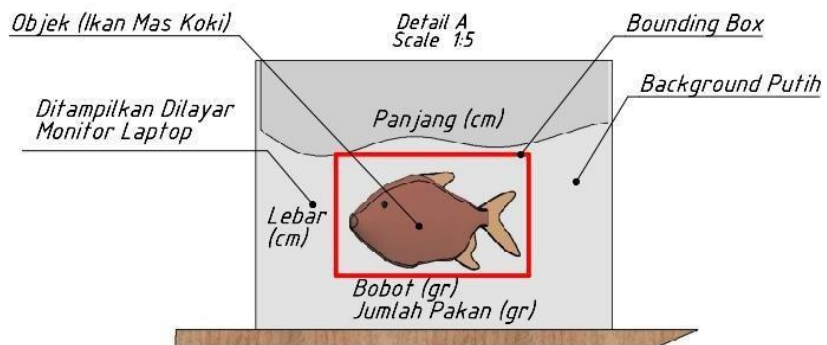
Pada penelitian ini, Penulis menggunakan 5 hingga 8 sampel Ikan Mas Koki yang memiliki ukuran dan bobot yang beragam. Sampel Ikan Mas Koki ini akan diletakkan di akuarium yang berukuran 40 cm x 25 cm x 25 cm dengan *background* putih. *Webcam* diletakkan pada *tripod* dengan jarak antara *webcam* dengan akuarium sejauh 30 cm dengan *webcam* yang mengarah ke titik tengah pada akuarium. Dalam penelitian ini, objek yang berada di akuarium hanya sampel Ikan Mas Koki tanpa ada objek lain dikarenakan sistem ini bersifat *universal* (dapat mendeteksi objek selain Ikan Mas Koki) tanpa ada *dataset* dari sampel tersebut.



Gambar 3.2 Desain Alat Sistem Pengukuran



Gambar 3.3 Ukuran Aquarium

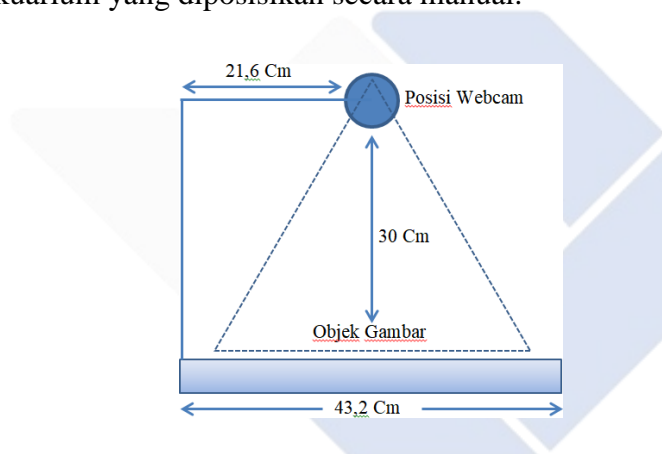


Gambar 3.4 Posisi Yang Diharapkan Ketika Pengujian

Pada Gambar 3.4 menunjukkan bagaimana hasil dari pengujian sistem. Pada penelitian ini, *background* putih pada akuarium berfungsi untuk mengurangi penyerapan cahaya yang masuk ke akuarium dan juga agar *webcam* dapat fokus dalam mendeteksi objek pada akuarium. Setelah melewati proses pada sistem, hasil yang didapatkan akan ditampilkan pada layar monitor laptop.

3.3.2 Melakukan Kalibrasi pada Webcam

Pada proses kalibrasi ini terdapat 3 parameter yang digunakan, yaitu parameter id, parameter *pixel*, dan parameter nilai skala. Nilai skala ini merupakan skala kalibrasi yang akan digunakan untuk di *converting* cm. Nilai *pixel* yang digunakan 1460 *pixel* yang didapatkan dari ukuran tinggi *webcam*. *Webcam* berjarak sekitar 30 cm dari akuarium yang diposisikan secara manual.



Gambar 3.5 Kalibrasi Webcam

3.3.3 Pembuatan Program Pengaturan Webcam

Dalam tahap ini, terdapat 3 proses yang dilakukan, antara lain:

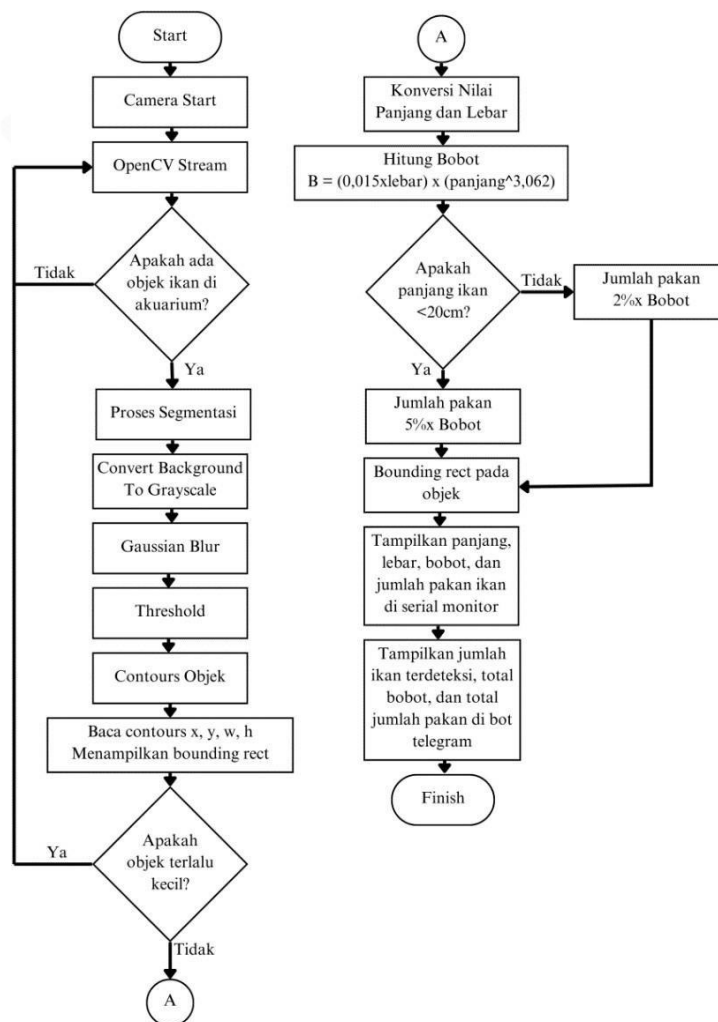
1. Konfigurasi webcam, pada proses dilakukan pengaturan pada *webcam* mulai dari mengatur id *webcam*, menentukan resolusi atau jumlah *pixel* yang akan digunakan oleh *webcam* dimana dalam hal ini, resolusi *webcam* diatur menjadi 2560 x 1440 *pixel*. Selain itu, *frame rate* pada *webcam* diatur menjadi 60 *frame/detik* yang berfungsi untuk menentukan seberapa sering *webcam* merekam citra/detik. Proses ini juga menentukan presentase dari gambar yang akan diambil secara otomatis, mengatur nilai *threshold*, dan menentukan

tingkat blur pada citra. Program konfigurasi ini mengatur parameter-parameter *webcam* serta nilai-nilai untuk proses pengukuran otomatis dan normalisasi citra.

2. Kemudian, pada *frame capture* berfungsi untuk mengambil objek/citra yang ada pada *webcam*.
3. Selanjutnya membuat program *frame draw*. Program ini berfungsi untuk untuk menampilkan teks dan *bounding rect* pada objek.

3.3.4 Pembuatan Program Utama Sistem

Berikut dibawah ini *flowchart* sistem kerja pengukuran bobot ikan hias dan penentuan jumlah pakan yang telah dirancang.



Gambar 3.6 *Flowchart* Sistem Kerja Alat

Tahap ini merupakan proses terakhir pada sistem dimana proses ini dimulai dengan mengambil citra menggunakan *webcam*. Citra tersebut akan masuk ke proses segmentasi dimana pada proses ini *OpenCV* melakukan beberapa proses, yaitu meng-*convert background* citra dari RGB menjadi *grayscale*, kemudian mengaburkan citra dengan *Gaussian blur*, dan mengaplikasikan *threshold* untuk mendeteksi objek dalam citra.

Selanjutnya, *OpenCV* akan digunakan untuk menemukan *contours* objek dalam citra yang telah di *threshold*. *Contour* yang ditemukan berupa x, y, w (*width*), h (*high*) dibaca menggunakan *bounding rect*. Hasil yang didapatkan dari menghitung panjang dan lebar dari *bounding rect* akan dihitung sebagai hasil pendeteksian panjang dan lebar ikan. Panjang dan lebar tersebut dikonversi menjadi *centimeter*. Nilai yang telah dikonversi akan digunakan untuk menghitung bobot Ikan Mas Koki menggunakan rumus dari jurnal terkait [8]:

$$W = a.T.L^b$$

Keterangan :

W = Berat

a = 0,015

T = Lebar

L = Panjang

b = 3,062

3.4 Penentuan Jumlah Pakan Ikan Hias

Setelah mengetahui bobot dari Ikan Mas Koki, tahap selanjutnya, yaitu menentukan jumlah pakan. Dalam penelitian ini, untuk jumlah pakan yang diberikan dibagi kedalam 2 kategori, dimana jika panjang ikan yang terdeteksi kurang dari 20 cm maka jumlah pakan yang diberikan sebesar 5% x bobot ikan tersebut, jika lebih dari 20 cm maka jumlah pakan sebesar 2% x bobot ikan. Jumlah pakan dari keseluruhan ikan ditotal untuk mengetahui banyaknya pakan dalam 1 kali pemberian pakan dikarenakan pada penelitian ini bobot masing-masing ikan berbeda sehingga jumlah pakan dari tiap ikan otomatis berbeda.

Berikut adalah perhitungan pemberian pakan ikan yang menjadi acuan dalam penelitian ini:

Tabel 3.1 Ukuran Pemberian Pakan Ikan [20]

Perbandingan	Ikan Dewasa	Ikan Kecil
Pemberian Pakan	2% dari bobot ikan	5% dari bobot ikan
Ukuran	20 cm – 50 cm	10 cm – 20 cm
Jumlah Pakan Ikan/ekor	2% x bobot ikan	5% x bobot ikan
Pemberian Pakan Ikan 1 Hari	3 kali sehari	3 kali sehari
Jumlah Pakan Ikan 1 Kali Makan	Jumlah pakan ikan/ekor x banyak ikan di akuarium	
Jumlah Pakan Ikan/Hari	Pakan ikan 1 kali makan x 3 kali sehari	

3.5 Pembuatan *Bot Telegram*

Setelah hasil pengukuran bobot ikan dan penentuan jumlah pakan telah ditampilkan pada *serial monitor* laptop, hasil tersebut akan diteruskan ke *Bot Telegram*. Namun poin yang ditampilkan pada *Bot* hanya jumlah ikan yang terdeteksi, total bobot, dan total jumlah pakan dari Ikan Mas Koki. Adapun langkah-langkah untuk membuat *Bot Telegram* sebagai berikut:

1. Ketik *botfather* pada kotak pencarian *Telegram*, kemudian klik *BotFather*.
2. Klik *start* untuk mulai berinteraksi dengan *BotFather*.
3. Klik *icon command Bot “/”*, kemudian pilih */newbot*.
4. Tuliskan judul *Bot* yang akan dibuat.
5. Setelah itu tulis *username* yang diakhiri dengan “bot”. Penulis menggunakan *username* PA Ikan Koki_bot.
6. Setelah *Bot* berhasil dibuat akan didapatkan *API token* yang digunakan untuk mengakses *API Bot* yang disediakan oleh *Telegram*.

3.6 Pengujian dan Evaluasi

Tahap pengujian ini dilakukan untuk mengetahui keberhasilan sistem yang dirancang. Dalam tahap ini, *webcam* akan diarahkan ke akuarium yang berisi ikan hias sebagai objek utama tanpa ada objek lain. Pada pengujian nanti yang menjadi

bahan penilaiannya, yaitu kesesuaian antara ukuran objek yang diukur secara manual dan sistem, selisih berat objek sebenarnya dengan berat objek dari sistem. Apabila terjadi *error* saat pengujian, maka proses selanjutnya ialah melakukan evaluasi terhadap perbaikan pada sistem. Adapun tahapan pengujian sistem yang akan dilakukan, yaitu sebagai berikut.

1. Pengujian dilakukan dengan mengecek apakah program dapat melakukan *capture* citra secara *realtime* pada objek yang akan di uji.
2. Pengujian dilakukan dengan 3 tahap yang berbeda, pengujian pertama objek dimasukkan secara bersamaan di akuarium. Kedua, masing-masing objek dilakukan pengujian sebanyak 5 kali. Pada pengujian tahap 1 dan tahap 2 hasil pengujian akan ditampilkan pada *serial monitor*. Sedangkan pada tahap 3, hasil pengujian akan ditampilkan pada *Bot Telegram*.
3. Pengambilan data dilakukan dengan meng-*capture* layar monitor pada laptop dan juga menampilkan pada *Bot telegram*, kemudian melakukan pengecekan data pada sistem dan *Bot* dengan membandingkan data yang didapatkan secara manual.

3.7 Pembuatan Laporan Proyek Akhir dan Publikasi Ilmiah

Setelah pengujian dan evaluasi sistem telah selesai dilakukan, maka selanjutnya akan dilakukan analisa terhadap kinerja alat yang telah dibuat. Analisa pada penelitian ini mencakup beberapa poin, antara lain selisih pengukuran bobot sebenarnya dengan pengukuran secara sistem, perhitungan tingkat akurasi dari pengukuran secara sistem, dan perhitungan jumlah pemberian pakan Ikan Mas Koki. Setelah analisa didapatkan, selanjutnya melakukan pembuatan laporan proyek akhir dan publikasi ilmiah dalam bentuk jurnal.

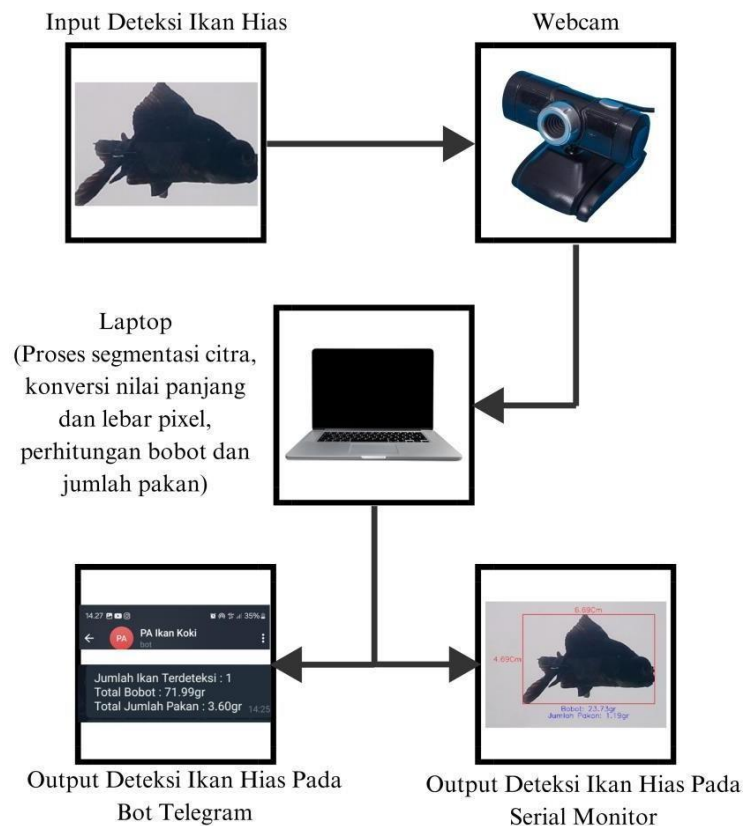
BAB IV PEMBAHASAN

Pada bab ini akan membahas mengenai proses pembuatan dari Proyek Akhir dengan judul **“Implementasi Teknologi *Image Processing* untuk Identifikasi Pengukuran Bobot Ikan Hias Dalam Penentuan Jumlah Pakan”**.

4.1 Deskripsi Alat

Sistem pengukuran bobot ikan hias ini merupakan sistem yang memanfaatkan teknologi *image processing* dengan bahasa pemrograman *Python* dan menggunakan aplikasi *Visual Studio Code* dan *Library OpenCV* untuk menentukan jumlah pakan dari ikan. *Library OpenCV* ini digunakan untuk mengolah citra yang masuk dari *webcam* yang selanjutnya akan mengimplementasikan proses pengukuran bobot ikan hias dan penentuan jumlah pakan. Sistem ini telah terintegrasi dengan *webcam* tipe NYK Nemesis A96 2K 30 fps yang digunakan untuk mendeteksi objek ikan pada akuarium yang berukuran 40 cm x 25 cm x 25 cm dengan *background* putih. Dalam penelitian ini, objek yang berada di akuarium hanya objek berupa Ikan Mas Koki tanpa ada objek yang lain dikarenakan sistem ini bersifat *universal* (dapat mendeteksi objek selain Ikan Mas Koki) tanpa ada *dataset* dari Ikan Mas Koki tersebut.

Ketika *webcam* telah mendeteksi adanya objek ikan, Citra yang ditangkap akan masuk ke proses *image processing* dengan memanfaatkan teknik segmentasi citra. Dalam proses inilah citra tersebut diolah untuk menemukan panjang dan lebar objek. Nilai yang telah didapatkan akan dikonversi dari *pixel* menjadi cm. Dari hasil konversi tersebut akan dilakukan perhitungan bobot untuk menentukan jumlah pakan ikan untuk 1 kali pemberian pakan/ikan. Selanjutnya, gambar yang telah diproses akan menghasilkan *output* pada *bounding rect* berupa panjang dan lebar ikan, bobot ikan, dan jumlah pakan ikan yang ditampilkan dilayar monitor laptop. Ilustrasi ditunjukkan pada Gambar 4.1.



Gambar 4.1 Rancangan Sistem

4.2 Pengembangan Sistem Pengukuran Bobot Ikan Hias

Pada tahapan ini, terdapat beberapa proses yang dilakukan untuk mendapatkan hasil pengujian dari bobot ikan hias.

4.2.1 Pengumpulan Data Objek







Pada penelitian ini, Penulis menggunakan 5 hingga 8 sampel Ikan Mas Koki yang memiliki ukuran dan bobot yang beragam. Sebelum melakukan pengukuran menggunakan sistem, sampel terlebih dahulu diukur secara konvensional untuk mengetahui nilai selisih/ tingkat akurasi dari pengukuran menggunakan sistem. Pengukuran konvensional dilakukan menggunakan alat ukur penggaris untuk menentukan panjang dan lebar ikan hias, kemudian untuk bobot menggunakan alat timbang.





Gambar 4.2 Pengukuran Konvensional

Berikut dibawah ini pada Tabel 4.1 Pengukuran Ikan Hias Secara Konvensional.

Tabel 4.1 Hasil Pengukuran Ikan Hias Secara Konvensional

Kode	Objek Ikan Hias	Panjang (cm)	Lebar (cm)	Bobot (gram)
1		7 cm	3 cm	15 gram
2		7 cm	4 cm	15 gram
3		7,5 cm	3 cm	20 gram
4		8 cm	3 cm	20 gram
5		8,5 cm	4 cm	45 gram
6		8,5 cm	5 cm	58 gram

Kode	Objek Ikan Hias	Panjang (cm)	Lebar (cm)	Bobot (cm)
7		9 cm	5 cm	64 gram
8		9,5 cm	5 cm	74 gram

Pada Tabel 4.1 diatas, maksud dari poin pertama (kode) menjelaskan bahwa dari setiap nomor pada kolom kode mewakili kolom objek Ikan Hias.

4.2.2 Proses Kalibrasi Webcam

Pada proses kalibrasi ini terdapat 3 parameter yang digunakan, yaitu parameter id, parameter *pixel*, dan parameter nilai skala. Parameter id ini digunakan untuk mengubah koordinat objek yang dideteksi menjadi koordinat sistem pada citra yang dihasilkan oleh *webcam*. Dalam parameter ini akan dilakukan pengaturan *focal length*, *principal point*, dan koefisien distorsi lensa. Kemudian, pada parameter *pixel* digunakan untuk mengatur ukuran dan jarak antara setiap titik di sensor *webcam* yang menangkap citra ikan. Pada parameter nilai skala ini merupakan skala kalibrasi yang akan digunakan untuk di *converting* cm. Nilai *pixel* yang digunakan 1460 *pixel* yang didapatkan dari ukuran tinggi *webcam*. Webcam berjarak sekitar 30 cm dari akuarium yang diposisikan secara manual. Berikut dibawah ini *output* dari proses kalibrasi *webcam*.

```

calibra3ov
1 d,0,0,150821044201152742
2 d,10,0,150821044201152742
3 d,20,0,150821044201152742
4 d,30,0,150821044201152742
5 d,40,0,155041418236518058
6 d,50,0,155041418236518058
7 d,60,0,155041418236518058
8 d,70,0,1508104202728509918
9 d,80,0,1508104202728509918
10 d,90,0,1508104202728509918
11 d,100,0,155030488772240006
12 d,110,0,155030488772240006
13 d,120,0,155030488772240006
14 d,130,0,155030488772240006
15 d,140,0,15666712054940126
16 d,150,0,15666712054940126
17 d,160,0,15666712054940126
18 d,170,0,15623091000542115
19 d,180,0,15623091000542115
20 d,190,0,15623091000542115
21 d,200,0,157976921579809096
22 d,210,0,157976921579809096
23 d,220,0,157976921579809096
24 d,230,0,157976921579809096
25 d,240,0,1503968141456676
26 d,250,0,1503968141456676
27 d,260,0,1503968141456676
28 d,270,0,1505144017652503
29 d,280,0,1505144017652503
30 d,290,0,1505144017652503
31 d,300,0,15023091000542115
32 d,310,0,15023091000542115
33 d,320,0,15623091000542115
34 d,330,0,15666712054940126

```

Gambar 4.3 Output Kalibrasi Webcam

4.2.3 Proses Pengaturan Webcam

Pada tahapan ini terdapat 3 proses yang akan dilakukan, antara lain:

1. Proses konfigurasi Webcam

Untuk mengoptimalkan penggunaan webcam, ada proses yang perlu dilakukan mulai dari mengatur id webcam, menentukan resolusi atau jumlah *pixel* yang akan digunakan oleh webcam agar menghasilkan citra dengan kejelasan yang tinggi, dalam hal ini, resolusi webcam diatur menjadi 2560 x 1440 *pixel*. Selain itu, *frame rate* pada webcam diatur menjadi 60 *frame/detik* yang berfungsi untuk menentukan seberapa sering webcam merekam citra/detik. Proses ini juga menentukan presentase gambar yang akan diambil secara otomatis, mengatur nilai *threshold* untuk menentukan batas atau nilai minimum dalam suatu kondisi tertentu, dan menentukan tingkat blur pada citra untuk mengurangi *noise* yang tidak diinginkan pada gambar. Program konfigurasi ini mengatur parameter-parameter webcam serta nilai-nilai untuk proses pengukuran otomatis dan normalisasi citra. Berikut dibawah ini program konfigurasi webcam.

```
# default settings

# camera values
camera_id = 1
camera_width = 2560
camera_height = 1440
camera_frame_rate = 60
#camera_fourcc = cv2.VideoWriter_fourcc(*"YUYV")
camera_fourcc = cv2.VideoWriter_fourcc(*"MJPG")

# auto measure mouse events
auto_percent = 0.2
auto_threshold = 100
auto_blur = 5

# normalization mouse events
norm_alpha = 0
norm_beta = 254
```

2. Program *Frame Capture*

Pada tahapan ini, *frame capture* berfungsi untuk mengambil objek/citra yang ada pada *webcam*.

```
def loop(self):

    # load start frame
    frame = self.black_frame
    if not self.buffer.full():
        self.buffer.put(frame,False)

    # status
    self.frame_grab_on = True
    self.loop_start_time = time.time()

    # frame rate
    fc = 0
    t1 = time.time()

    # loop
    while 1:

        # external shut down
        if not self.frame_grab_run:
            break

        # true buffered mode (for files, no loss)
        if self.buffer_all:

            # buffer is full, pause and loop
            if self.buffer.full():
                time.sleep(1/self.camera_frame_rate)

            # or load buffer with next frame
            else:

                grabbed,frame = self.camera.read()

                if not grabbed:
                    break

                self.buffer.put(frame,False)
                self.frame_count += 1
                fc += 1

        # false buffered mode (for camera, loss allowed)
        else:

            grabbed,frame = self.camera.read()
            if not grabbed:
                break

            # open a spot in the buffer
```

```

        if self.buffer.full():
            self.buffer.get()

        self.buffer.put(frame,False)
        self.frame_count += 1
        fc += 1

    # update frame read rate
    if fc >= 10:
        self.current_frame_rate = round(fc/(time.time()-t1),2)
        fc = 0
        t1 = time.time()

```

Keterangan :

Program diatas merupakan bagian inti dari kelas „**camera_Thread**“ pada program *frame capture* secara keseluruhan yang bertanggung jawab untuk mengambil dan memproses *frame* dari *webcam*. Ini adalah metode „**loop()**“ yang dieksekusi dalam sebuah thread untuk menangani pembacaan *frame* secara terus-menerus.

3. Program *Frame Draw*

Program ini berfungsi untuk untuk menampilkan teks dan *bounding rect* pada objek.

```

import cv2

class DRAW:

    # frame
    width = 640
    height = 480

    # colors B-G-R
    colors = {'red'   :( 0, 0,255),
              'green' :( 0,255, 0),
              'blue'  :(255, 0, 0),
              'yellow' :( 0,255,255),
              'gray'  :(200,200,200),
              }

    # add text block
    def add_text_top_left(self,frame,text):

        if type(text) not in (list,tuple):
            text = text.split('\n')
            text = [line.rstrip() for line in text]

```

```

color = self.colors.get('blue', (0,255,255))

lineloc = 10
lineheight = 30

for line in text:
    lineloc += lineheight
    cv2.putText(frame,
                line,
                (10,lineloc), # location
                #cv2.FONT_HERSHEY_PLAIN, # font
                cv2.FONT_HERSHEY_SIMPLEX, # font
                0.8, # size
                color, # color
                1, # line width
                cv2.LINE_AA, #
                False) #

# add text anywhere
def
add_text(self, frame, text, x, y, size=0.8, color='yellow', center=False, mid
dle=False, top=False, right=False):

    color = self.colors.get(color, (0,255,255))
    #font = cv2.FONT_HERSHEY_PLAIN
    font = cv2.FONT_HERSHEY_SIMPLEX

    textsize = cv2.getTextSize(text, font, size, 1)[0]

    if center:
        x -= textsize[0]/2
    elif right:
        x -= textsize[0]
    if top:
        y += textsize[1]
    elif middle:
        y += textsize[1]/2

    cv2.putText(frame,
                text,
                (int(x),int(y)),
                font,
                size,
                color,
                1, # line width
                cv2.LINE_AA,
                False)

# line
def line(self, frame, x1, y1, x2, y2, weight=1, color='green'):

cv2.line(frame, (int(x1),int(y1)), (int(x2),int(y2)), self.colors.get(co
lor, (0,255,0)), weight)

```

```

# vertical line
def vline(self, frame, x=0, weight=1, color='green'):
    if x <= 0:
        x = self.width/2
    x = int(x)

cv2.line(frame, (x,0), (x, self.height), self.colors.get(color, (0,255,0)),
,weight)

# horizontal line
def hline(self, frame, y=0, weight=1, color='green'):
    if y <= 0:
        y = self.height/2
    y = int(y)

cv2.line(frame, (0,y), (self.width,y), self.colors.get(color, (0,255,0)),
weight)

# rectangle
def
rect(self, frame, x1,y1,x2,y2, weight=1, color='green', filled=False):
    if filled:
        weight = -1

cv2.rectangle(frame, (int(x1),int(y1)), (int(x2),int(y2)), self.colors.g
et(color, (0,255,0)), weight)

# circle
def
circle(self, frame, x1,y1,x2,y2, r, weight=1, color='green', filled=False):
    if filled:
        weight = -1

cv2.circle(frame, (int(x1),int(y1)), (int(x2),int(y2)), int(r), self.colo
rs.get(color, (0,255,0)), weight)

# centered crosshairs full frame
def crosshairs_full(self, frame, weight=1, color='green'):
    self.vline(frame,0,weight,color)
    self.hline(frame,0,weight,color)

# centered crosshairs
def
crosshairs(self, frame, offset=10, weight=1, color='green', invert=False):
    offset = self.width*offset/200
    xcenter = self.width/2
    ycenter = self.height/2
    if invert:
        self.line(frame,0,ycenter,xcenter-
offset,ycenter,weight,color)

self.line(frame,xcenter+offset,ycenter,self.width,ycenter,weight,colo
r)

```



```

        self.line(frame,xcenter,0,xcenter,ycenter-
offset,weight,color)

self.line(frame,xcenter,ycenter+offset,xcenter,self.height,weight,col
or)
    else:
        self.line(frame,xcenter-
offset,ycenter,xcenter+offset,ycenter,weight,color)
        self.line(frame,xcenter,ycenter-
offset,xcenter,ycenter+offset,weight,color)

```

Keterangan :

Program diatas merupakan kelas „DRAW“ yang memiliki fungsi untuk menggambar pada *frame* citra menggunakan *OpenCV*.

4.2.4 Proses Pengolahan Citra

Pada tahapan ini, proses dimulai dengan mengambil citra menggunakan *webcam*. Citra tersebut akan masuk ke proses segmentasi dimana pada proses ini *OpenCV* melakukan beberapa proses, yaitu:

1. *Convert Background RGB ke Grayscale*

Dalam proses ini, citra yang ditangkap oleh *webcam* akan diubah terlebih dahulu ke *grayscale* agar lebih mudah untuk mengenali objek.



(a)

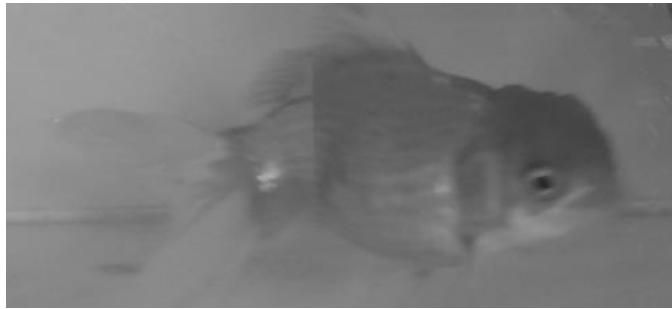
(b)

Gambar 4.4 *Grayscale*

Pada gambar diatas, terdapat dua gambar dimana (a) merupakan citra dari ikan yang memiliki warna cerah, yaitu warna *orange* dan (b) citra dengan warna gelap/hitam pekat.

2. *Gaussian Blur*

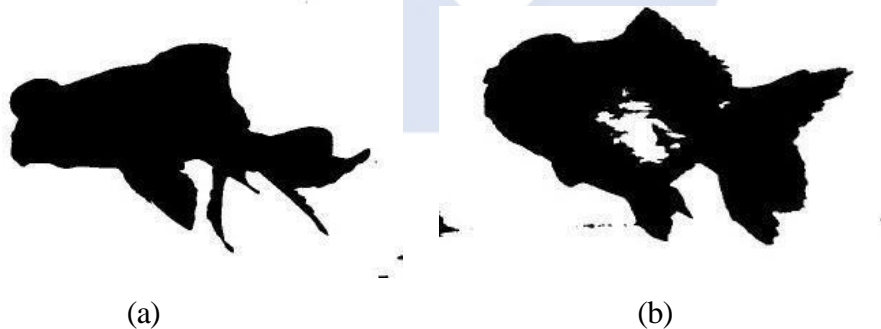
Setelah gambar diubah ke *grayscale*, citra akan masuk ke proses *gaussian blur* menggunakan filter *gaussian* yang bertujuan untuk mengurangi detail dan gangguan dalam citra.



Gambar 4.5 *Gaussian Blur*

3. *Threshold*

Dalam proses ini, citra akan diubah menjadi citra biner yang setiap *pixel*-nya menjadi nilai hitam atau putih, tergantung dari nilai *pixel* tersebut melebihi atau kurang dari nilai ambang batasnya. Dalam program ini, nilai ambang batas diatur sebesar 100.



Gambar 4.6 *Threshold*

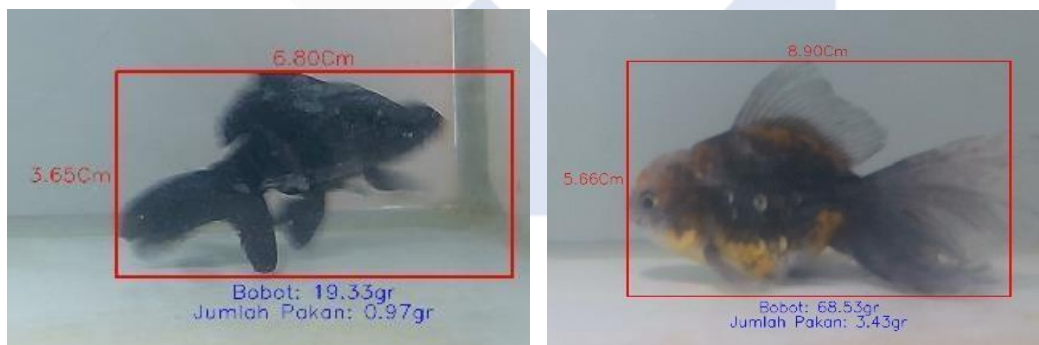
Seperti pada *grayscale*, pada keterangan (a) citra memiliki warna gelap/hitam pekat sehingga dalam proses ini hasil yang didapatkan tidak terdapat *noise* pada citra, sedangkan pada (b) dengan warna citra yang cerah (*orange*), terdapat *noise* karena warna tersebut mudah mengikuti warna dari *background*.

4. *Contour* Objek

Proses terakhir pada tahapan ini, yaitu *contour* objek dimana latar belakang pada objek akan dipisahkan dengan garis atau tepi. *Contour* dibaca menggunakan *bounding rect* untuk menemukan nilai x , y , w (*width*), dan h (*high*). Hasil dari panjang dan lebar pada *bounding rect* akan dihitung sebagai hasil panjang dan lebar dari ikan. Panjang dan lebar tersebut dikonversi menjadi cm.

4.2.5 Pengujian Pengukuran Bobot Ikan Hias

Setelah mendapatkan nilai panjang dan lebar dari ikan dalam bentuk cm, dengan adanya *library Math* maka nilai tersebut dapat dihitung untuk mendapatkan bobot ikan dengan rumus yang telah dimasukkan pada program utama. Untuk mendapatkan hasil yang dengan nilai akurasi yang tinggi, pengujian untuk pengukuran bobot ikan hias dilakukan dalam beberapa kali pengujian. Hasil yang didapatkan dalam perhitungan akan ditampilkan pada *serial monitor* laptop secara *realtime* seperti pada Gambar 4.7 dibawah ini.



Gambar 4.7 Hasil Pengujian Secara Sistem Pada *Serial Monitor*








Gambar 4.8 Hasil Pengujian Secara Sistem Pada *Bot Telegram*

Berbeda dengan hasil pengujian pada Gambar 4.7, untuk hasil pengujian dari Gambar 4.8 hanya menampilkan 3 poin, yaitu jumlah ikan terdeteksi, total bobot, dan total jumlah pakan dari ikan.

Berikut dibawah ini Tabel 4.2 yang melakukan pengujian dengan cara memasukkan objek Ikan Mas Koki secara bersamaan pada akuarium. Pada pengujian ini, objek yang diuji berjumlah 7 ekor.

Tabel 4.2 Pengujian 1 Secara Bersamaan Dengan *Serial Monitor*

Gambar Objek	Panjang (cm)		Lebar (cm)		Bobot (gram)	
	Manual	Sistem	Manual	Sistem	Manual	Sistem
	7	6,39	3	4,38	15	12
	7	5,62	4	3,84	15	12
	7,5	6,62	3	3,97	20	26,09
	8	8,23	3	3,53	20	15
	8,5	8,90	5	5,31	58	62,71
	9	8,63	5	5,50	64	65,31
	9,5	8,62	5	4,57	74	45,08
Total	8,07	7,57	4	4,44	36,28	34,02

Untuk menguji tingkat keakuratan sistem pada Tabel 4.2 diatas, dalam penelitian ini dibutuhkan galat agar dapat mengetahui tingkat akurasi kebenaran dan kesalahan dari sistem dengan galat yang minim. Nilai galat yang ditentukan adalah 50% untuk tingkat akurasi kebenaran dan 50% tingkat kesalahan. Perhitungan galat dilakukan dengan PM (Panjang Manual untuk nilai eksak), PH (Panjang Hasil untuk nilai uji), LM (Lebar Manual untuk nilai eksak), LH (Lebar Hasil untuk nilai uji), BM (Bobot Manual untuk nilai eksak), dan BH (Bobot Hasil untuk nilai uji) sebagai berikut.

$$\begin{aligned} \text{Galat } \xi \text{ Panjang} &= |\text{Total PM} - \text{Total PH}| \\ &= |8,07 - 7,57| \\ &= 0,5 \end{aligned}$$

$$\begin{aligned} \text{Kesalahan Relatif } \xi_4 &= |0,5/8,07| * 100\% \\ &= 6,2\% \end{aligned}$$

Jadi, keakuratan sistem adalah $100\% - 6,2\% = 93,8\%$

$$\begin{aligned} \text{Galat } \xi \text{ Lebar} &= |\text{Total LM} - \text{Total LH}| \\ &= |4 - 4,44| \\ &= 0,44 \end{aligned}$$

$$\begin{aligned} \text{Kesalahan Relatif } \xi_4 &= |0,44/4| * 100\% \\ &= 11\% \end{aligned}$$

Jadi, keakuratan sistem adalah $100\% - 11\% = 89\%$



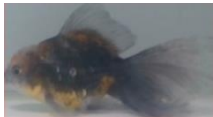

$$\begin{aligned} \text{Galat } \xi \text{ Bobot} &= |\text{Total BM} - \text{Total BH}| \\ &= |36,28 - 34,02| \\ &= 2,26 \end{aligned}$$


$$\begin{aligned} \text{Kesalahan Relatif } \xi_4 &= |2,26/36,28| * 100\% \\ &= 6,22\% \end{aligned}$$

Jadi, keakuratan sistem adalah $100\% - 6,22\% = 93,78\%$

Berikut dibawah ini Tabel 4.3 hasil pengujian yang melakukan pengukuran Ikan Mas Koki/ekor, dimana tiap ekor ikan dilakukan pengujian sebanyak 5 kali. Dalam pengujian ini hanya menggunakan 5 sampel Ikan Mas Koki.

Tabel 4.3 Pengujian 2 Ikan Hias/ Ekor Dengan Serial Monitor

Gambar Objek	Manual			Sistem		
	Panjang (cm)	Lebar (cm)	Bobot (gr)	Panjang (cm)	Lebar (cm)	Bobot (gr)
	7,5	3	20	6,80	3,65	19,33
				6,52	4,74	22,07
				7,09	3,81	22,99
				6,69	4,69	23,73
				7,16	4,81	29,95
Total	7,5	3	20	6,85	4,34	23,62
Tingkat Akurasi (%)				91,33%	55%	81,9%
	8,5	4	45	7,99	5,07	44,10
				8,63	5,27	46,54
				8,43	4,72	48,38
				8,55	4,96	52,99
				8,55	5,16	55,19
Total	8,5	4	45	8,43	5,03	49,44
Tingkat Akurasi (%)				99,17%	74,25%	90,14%
	8,5	5	58	8,68	5,40	60,49
				8,49	5,79	60,66
				8,65	5,53	61,33
				8,19	6,53	61,36
				8,25	6,41	61,59
Total	8,5	5	58	8,45	5,93	61,08
Tingkat Akurasi (%)				99,41%	81,4%	94,69%
	9	5	64	9,35	4,22	59,53
				9,57	4,50	68,17
				8,84	6,03	71,51
				9,78	4,52	72,99

Gambar Objek	Manual			Sistem		
	Panjang (cm)	Lebar (cm)	Bobot (gr)	Panjang (cm)	Lebar (cm)	Bobot (gr)
				9,78	4,71	76,09
Total	9	5	64	9,46	4,8	69,66
Tingkat Akurasi (%)				94,9%	96%	91,15%
				8,97	5,59	69,36
	9,5	5	74	9,21	5,49	73,70
				8,93	6,56	80,1
				10,6	4,57	80,61
				10,22	4,37	80,83
				Total	9,5	5
Tingkat Akurasi (%)				99,06%	93,6%	96,05%
Total Tingkat Akurasi (%)				96,77%	80,05%	90,79%

Pada Tabel 4.3 diatas merupakan hasil pengujian yang melakukan pengukuran Ikan Mas Koki/ekor, dimana tiap ekor ikan dilakukan pengujian sebanyak 5 kali. Setelah dilakukan perhitungan galat untuk masing-masing ikan dapat dilihat bahwa sistem mampu menghasilkan tingkat akurasi melebihi 50%. Dengan melakukan penjumlahan dari tingkat akurasi setiap ikan, dapat diketahui bahwa nilai akurasi dari pengukuran secara sistem sebesar 96,77% (panjang), 80,05% (lebar), dan 90,79% (bobot).






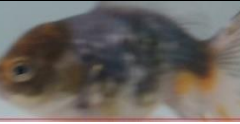
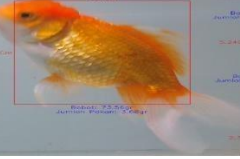
Dari hasil perhitungan galat pada Tabel 4.2 dan Tabel 4.3 dapat dilihat bahwa sistem mampu menghasilkan tingkat akurasi kebenaran melebihi 50%, sehingga semakin kecil nilai kesalahan relatif yang didapatkan, maka nilai perkiraan yang diperoleh akan semakin baik. Sebaliknya, semakin besar nilai kesalahan relatif, maka nilai perkiraan akan semakin buruk.

4.3 Pengujian Penentuan Jumlah Pakan

Setelah hasil bobot didapatkan, maka langkah terakhir dalam penelitian ini ialah menentukan jumlah pakan pada ikan hias. Untuk jumlah pakan sendiri ditentukan

dari Tabel 3.1 pada Bab 3. Berikut dibawah ini Tabel 4.4 jumlah pakan ikan/ekor dari pengukuran bobot Tabel 4.2 yang diperkirakan berdasarkan manual dan sistem.

Tabel 4.4 Pengujian 1 Jumlah Pakan Ikan Dengan *Serial Monitor*

Gambar Objek	Bobot (gram)		Pakan (gram)		Selisih Pakan (gram) M-S
	Manual	Sistem	Manual	Sistem	
	15	12	0,75	0,67	0,08
	15	12	0,75	0,64	0,11
	20	26,09	1,1	1,30	0,2
	20	15	1	0,83	0,17
	58	62,71	2,9	3,14	0,24
	64	65,31	3,2	3,27	0,07
	74	45,08	3,7	2,25	1,45
Total	38,28	34,02	13,4	12,1	1,3



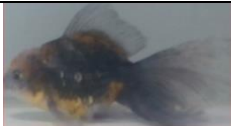
$$\begin{aligned}
 \text{Galat } \xi_4 \text{ Jumlah Pakan} &= |\text{Total Pakan Manual} - \text{Total Pakan Sistem}| \\
 &= |13,4 - 12,1| \\
 &= 1,3
 \end{aligned}$$


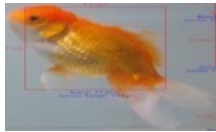
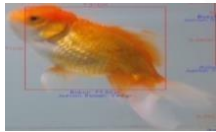
$$\begin{aligned} \text{Kesalahan Relatif } \xi_4 &= |1,3/13,4| * 100\% \\ &= 9,7\% \end{aligned}$$

Jadi, keakuratan sistem adalah $100\% - 9,7\% = 90,3\%$

Pada Tabel 4.4 diatas, merupakan hasil yang dihitung secara manual dan diperkirakan dengan sistem. Untuk total pakan ikan secara sistem tidak ada dikarenakan pada penelitian ini, penentuan jumlah pakan hanya terbatas pada jumlah pakan/ekor sehingga untuk total pakan secara sistem harus dihitung secara manual dengan menjumlahkan nilai pakan dari setiap ekor ikan. Dari hasil pengujian yang telah dilakukan dengan memasukkan ikan secara bersamaan ke akuarium dapat dilihat bahwa perhitungan jumlah pakan secara sistem mencapai tingkat akurasi 90,3% dengan selisih 1,3 gram.

Tabel 4.5 Pengujian 2 Jumlah Pakan Ikan Dengan *Serial Monitor*

Gambar Objek	Bobot (gram)		Pakan (gram)		Selisih Pakan (gram) M-S
	Manual	Sistem	Manual	Sistem	
	20	19,33	1,1	0,97	0,13
		22,07		1,10	0
		22,99		1,15	0,05
		23,73		1,19	0,09
		29,95		1,50	0,4
		Mean			1,18
	45	44,10	2,25	2,21	0,14
		46,54		2,33	0,02
		48,38		2,42	0,07
		52,99		2,65	0,3
		55,19		2,76	0,41
		Mean			2,47
	58	60,49	2,9	3,02	0,02
		60,66		3,03	0,03

Gambar Objek	Bobot (gram)		Pakan (gram)		Selisih Pakan (gram) M-S
	Manual	Sistem	Manual	Sistem	
		61,33		3,07	0,07
		61,36		3,07	0,07
		61,59		3,08	0,08
		Mean		3,05	0,15
		59,53		2,98	0,52
		68,17		3,41	0,09
	64	71,51	3,2	3,58	0,08
		72,99		3,65	0,15
		76,09		3,80	0,3
	Mean		3,48	0,28	
		69,36		3,47	0,53
		73,70		3,69	0,31
	74	80,1	3,7	4,01	0,01
		80,61		4,03	0,03
		80,83		4,04	0,04
	Mean		3,85	0,15	
	Total		13,15	14,03	0,88

$$\begin{aligned} \text{Galat } \xi_4 \text{ Jumlah Pakan} &= |\text{Total Pakan Manual} - \text{Total Pakan Sistem}| \\ &= |13,15 - 14,03| \\ &= 0,88 \end{aligned}$$

$$\begin{aligned} \text{Kesalahan Relatif } \xi_4 &= |0,68/13,15| * 100\% \\ &= 6,69\% \end{aligned}$$

Jadi, keakuratan sistem adalah $100\% - 6,69\% = 93,31\%$

Pada Tabel 4.5 diatas, merupakan hasil pengujian dengan melakukan pengukuran ikan/ekor pada akuarium yang dihitung secara manual dan diperkirakan dengan sistem. Untuk total pakan ikan secara sistem dilakukan secara manual dengan

menjumlahkan nilai *mean* pakan secara sistem dari tiap ekor ikan dikarenakan pada penelitian ini hanya berfokus untuk penentuan jumlah pakan ikan/ekor. Dari hasil pengujian, dapat dilihat bahwa tingkat akurasi perhitungan jumlah pakan secara sistem mencapai 93,31% dengan selisih 0,88 gram.

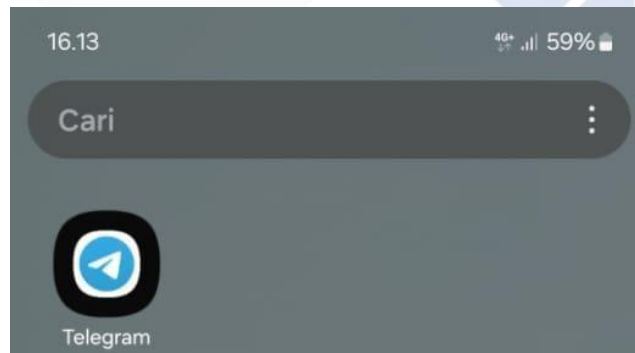
4.4 Pengujian Dengan Menggunakan *Bot Telegram*

Dalam pengujian ini dilakukan pengukuran Ikan Mas Koki/ekor, dengan setiap ikan diukur sebanyak 2 kali. Untuk pengujiannya menggunakan 5 sampel Ikan Mas Koki.

4.4.1 Langkah-Langkah Menjalankan *Bot Telegram*

Dalam proses ini, setelah sistem dijalankan dan *webcam* telah mendeteksi objek dalam akuarium, adapun langkah-langkah yang harus dilakukan agar hasil tersebut ditampilkan pada *Bot Telegram* sebagai berikut:

1. Buka aplikasi *Telegram* yang telah di-*install* pada *smartphone*.



Gambar 4.9 Aplikasi *Telegram* Pada *SmartPhone*

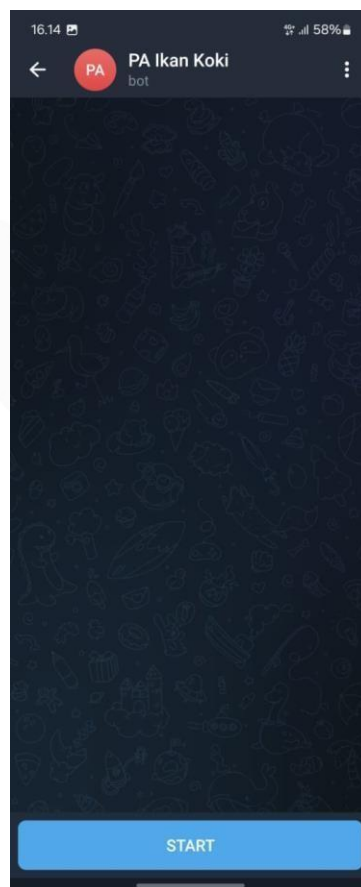
Dalam penelitian ini, *Telegram* di-*install* di *smartphone* dikarenakan lebih memudahkan pengguna menjalankan sistem, salah satunya dapat digunakan untuk *me-monitoring* dalam jarak jauh.

2. Kemudian akan masuk ke tampilan menu *Telegram* yang memuat *Bot Telegram* yang telah dibuat. Disini judul yang digunakan untuk *Bot Telegram* ialah “PA Ikan Koki”.



Gambar 4.10 Tampilan Menu *Telegram*

3. Klik *Bot Telegram* tersebut dan akan masuk ke menu *chat Bot* seperti gambar dibawah ini.



Gambar 4.11 Menu *Chat Bot*

4. Tekan **START** untuk memulai interaksi dengan *Bot* sesuai dengan gambar dibawah ini.



Gambar 4.12 Start Pada Bot

Hasil yang didapatkan dari *webcam* setelah objek terdeteksi akan masuk ke *Bot Telegram* dengan tiga poin, yaitu jumlah ikan yang terdeteksi, total bobot, dan total jumlah pakan.

5. Hasil akan terus ditampilkan pada *Bot Telegram* selama sistem berjalan dan *webcam* masih mendeteksi objek. Ketik `/stop` untuk menghentikan data masuk ke *Bot Telegram*.



Gambar 4.13 Stop Pada Bot





Hasil akan berhenti ditampilkan pada *Bot Telegram* walaupun sistem masih berjalan.

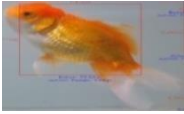
6. Untuk melihat kembali hasil yang didapatkan *webcam*, tekan **START**.

Setelah hasil yang didapatkan sesuai dengan yang ingin dicapai, akan dilakukan perhitungan untuk mengetahui tingkat akurasi dari hasil pengukuran

menggunakan *Bot Telegram* dan membandingkan dengan hasil pengukuran konvensional. Berikut dibawah ini Tabel 4.6 hasil pengujian yang ditampilkan pada *Bot Telegram*.

Tabel 4.6 Hasil Pengujian Dengan *Bot Telegram*

Gambar Objek	Manual		<i>Bot Telegram</i>		Gambar Hasil
	Bobot (gr)	Pakan (gr)	Bobot (gr)	Pakan (gr)	
	26	1,3	29,13	1,44	Jumlah Ikan Terdeteksi : 1 Total Bobot : 29.13gr Total Jumlah Pakan : 1.44gr 15:11
			30,19	1,50	Jumlah Ikan Terdeteksi : 1 Total Bobot : 30.19 Total Jumlah Pakan : 1.50 08:42
Mean			29,66	1,47	
Tingkat Akurasi (%)			85,93%	86,93%	
	45	2,25	44,89	2,24	Jumlah Ikan Terdeteksi : 1 Total Bobot : 44.89 Total Jumlah Pakan : 2.24 14:51
			52,07	2,60	Jumlah Ikan Terdeteksi : 1 Total Bobot : 52.07 Total Jumlah Pakan : 2.60 14:51
Mean			48,48	2,42	
Tingkat Akurasi (%)			92,27%	94,25%	
	58	2,9	54,25	2,71	Jumlah Ikan Terdeteksi : 1 Total Bobot : 54.25 Total Jumlah Pakan : 2.71 15:00
			67,34	3,37	Jumlah Ikan Terdeteksi : 1 Total Bobot : 67.34 Total Jumlah Pakan : 3.37 14:59
Mean			60,79	3,04	
Tingkat Akurasi (%)			95,19%	95,18%	
	64	3,2	66,07	3,31	Jumlah Ikan Terdeteksi : 1 Total Bobot : 66.07 Total Jumlah Pakan : 3.31 08:30
			68,21	3,42	Jumlah Ikan Terdeteksi : 1 Total Bobot : 68.21 Total Jumlah Pakan : 3.42 08:32

Gambar Objek	Manual		Bot Telegram		Gambar Hasil
	Bobot (gr)	Pakan (gr)	Bobot (gr)	Pakan (gr)	
Mean			67,14	3,36	
Tingkat Akurasi (%)			95,1%	95%	
	74	3,7	73,72	3,78	Jumlah Ikan Terdeteksi : 1 Total Bobot : 78.55 Total Jumlah Pakan : 3.89 08:39
			78,55	3,89	Jumlah Ikan Terdeteksi : 1 Total Bobot : 78.55 Total Jumlah Pakan : 3.89 08:39
Mean			76,13	3,8	
Tingkat Akurasi (%)			97,13%	97,3%	
Total Tingkat Akurasi (%)			93,12%	93,73%	

Berbeda dengan Tabel 4.2 hingga Tabel 4.4 yang hasil pengujiannya ditampilkan pada *serial monitor* dan untuk mendapatkan data dengan meng-*capture* layar *monitor*. Pada Tabel 4.6, hasil pengujian akan ditampilkan pada *Bot Telegram*. Untuk tingkat akurasi sendiri didapatkan dengan perhitungan galat, yaitu:

$$\text{Galat } \xi_4 = |\text{Total Manual} - \text{Total Bot Telegram}|$$

$$\text{Kesalahan Relatif } \xi_4 = |\text{Galat } \xi_4 / \text{Total Manual}| * 100\%$$

$$\text{Tingkat Akurasi (\%)} = 100\% - \text{Kesalahan Relatif } \xi_4$$

Sedangkan,

$$\text{Total Tingkat Akurasi (\%)} = \frac{\text{Jumlah Tingkat Akurasi (\%)}}{\text{Jumlah Objek Ikan Hias}}$$

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan dari tujuan proyek akhir ini maka Penulis dapat mengambil kesimpulan sebagai berikut:

1. Sistem pengukuran bobot Ikan Mas Koki untuk penentuan jumlah pakan ini bersifat *universal* karena tidak menggunakan *dataset* sehingga objek yang berada pada akuarium hanya Ikan Mas koki tanpa ada objek lain dengan *background* putih pada akuarium agar *webcam* dapat fokus mendeteksi Ikan Mas Koki.
2. Sistem ini dapat mengenali Ikan Mas Koki yang menjadi objek dan *bounding rect* dapat mengikuti arah gerak dari objek tersebut.
3. Sistem pengukuran bobot Ikan Mas Koki untuk penentuan jumlah pakan ini dapat berjalan secara *realtime* mengikuti arah gerak objek namun belum bekerja secara optimal.
4. Banyak faktor yang mempengaruhi sistem bekerja kurang optimal, seperti tidak adanya dataset pada sistem menyebabkan sistem mudah mendeteksi objek selain Ikan Mas koki, faktor jarak yang terlalu dekat ataupun jauh, pencahayaan yang terlalu minim maupun terlalu terang, dan kualitas perangkat yang digunakan.
5. Hasil tingkat akurasi yang diperoleh dari pengukuran bobot secara sistem yang ditampilkan dengan *serial monitor* untuk pengujian dengan memasukkan ikan secara bersamaan sebesar 93,78% dengan tingkat akurasi jumlah pakan 90,3%. Sedangkan pengukuran bobot dengan pengujian ikan/ekor sebesar 90,79% dengan tingkat akurasi jumlah pakan 93,31%. Untuk pengujian ikan/ekor yang ditampilkan *Bot Telegram* mendapat hasil akurasi bobot sebesar 93,12% dan pakan 93,73%.

6. Semakin kecil nilai kesalahan relatif, maka nilai perkiraan yang diperoleh akan semakin baik. Sebaliknya, semakin besar nilai kesalahan relatif, maka nilai perkiraan akan semakin buruk.
7. Pengukuran yang akurat pada sistem ialah ketika kondisi ikan menghadap samping dengan posisi sempurna.

5.2 Saran

Pada penelitian proyek akhir ini masih terdapat beberapa kekurangan dalam penelitiannya. Maka dari itu, Penulis mempunyai beberapa saran untuk perkembangan penelitian kedepannya sebagai berikut:

1. Sistem dibuat dengan menggunakan *dataset* sehingga *webcam* tidak mendeteksi objek lain selain objek penelitian.
2. Menambahkan sistem *monitoring* yang lebih baik sehingga memudahkan para pengguna untuk mengetahui perkembangan dari objek penelitian.
3. Menggunakan perangkat dengan spesifikasi dan kualitas yang tinggi.

DAFTAR PUSTAKA

- [1] I. M. Widiastuti, "Pertumbuhan dan sintasan Ikan Mas Koki (*Carassius auratus* L.) PADA BERBAGAI DOSIS PAKAN ALAMI *Tubifex* sp .," vol. 22, no. 1, pp. 32–40, 2021, [Online]. Available: <https://ejurnal.fapetkan.untad.ac.id/%0Aindex.php/agrisains>
- [2] N. S. Damanhuri *et al.*, "An automated length measurement system for tilapia fish based on image processing technique," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1088, no. 1, p. 012049, 2021, doi: 10.1088/1757-899x/1088/1/012049.
- [3] R. W. T. Hartono, G. A. Sasono, and ..., "Implementasi Algoritma Canny Edge Detection untuk Identifikasi Panjang dan Berat Ikan Koi Saat Bergerak," *Sent. 2018 Semin. Nas. Tek. Elektro 2018*, pp. 35–44, 2018, [Online]. Available: <http://senter.ee.uinsgd.ac.id/repositori/index.php/prosiding/article/view/senter2018p5>
- [4] A. Qashlim, Basri, Haeruddin, I. Nurtanio, A. Ahmad Ilham, and A. Ilham, "Estimation of milkfish physical weighting as fishery industry support system using image processing technology," *J. Phys. Conf. Ser.*, vol. 1175, no. 1, 2019, doi: 10.1088/1742-6596/1175/1/012029.
- [5] M. Uly, Pinandoyo, and S. Hastuti, "Pengaruh karotenoid dari tepung alga *Haematococcus pluvialis* dan marigold berbasis isokarotenoid pada pakan buatan terhadap kecerahan warna oranye, efisiensi pemanfaatan pakan dan pertumbuhan ikan mas koki (*Carassius auratus*)," *J. Aquac. Manag. Technol.*, vol. 6, no. 3, pp. 169–178, 2017, [Online]. Available: <http://ejournal-s1.undip.ac.id/index.php/jamt%0APENGARUH>
- [6] R. Islamadina, N. Pramita, F. Arnia, and K. Munadi, "Estimasi Panjang dan Lebar Ikan Berdasarkan Visual Capture," *Semin. Nas. dan Expo Tek. Elektro*, pp. 97–101, 2017, [Online]. Available: <http://snete.unsyiah.ac.id/2017/wp-content/uploads/2018/10/Naskah-17->

- [7] C. H. Tseng, C. L. Hsieh, and Y. F. Kuo, "Automatic measurement of the body length of harvested fish using convolutional neural networks," *Biosyst. Eng.*, vol. 189, no. 1, pp. 36–47, 2020, doi: 10.1016/j.biosystemseng.2019.11.002.
- [8] M. Lorenzoni, L. Ghetti, G. Pedicillo, and A. Carosi, "Analysis of the biological features of the goldfish *carassius auratus auratus* in Lake Trasimeno (Umbria, Italy) with a view to drawing up plans for population control," *Folia Zool.*, vol. 59, no. 2, pp. 142–156, 2010, doi: 10.25225/fozo.v59.i2.a9.2010.
- [9] Erlangga, E. Riri, and Mawardi, "PERUBAHAN RESPON PAKAN PADA IKAN MAS KOKI (*Carasias auratus*) DENGAN RANSANGAN WARNA LAMPU," *Berk. Perikan. Terubuk*, vol. 45, no. 2, pp. 12–18, 2017.
- [10] M. T. Fajar, "Pengaruh Perbedaan Konsentrasi Pakan Pelet Terhadap Bobot Dan Panjang Ikan Mas (*Cyprinus Carpio*)," *INSOLOGI J. Sains dan Teknol.*, vol. 1, no. 5, pp. 498–504, 2022, doi: 10.55123/insologi.v1i5.801.
- [11] A. Sulistyowati, Y. S. Hariyanti, and A. Novianti, "Perancangan aplikasi pembaca warna dan bentuk berbasis pengolahan citra untuk daftar katalog perpustakaan," vol. 4, no. 3, pp. 2554–2566, 2018.
- [12] M. F. H. Dermawan, D. Witasryah, and H. Fakhruroja, "Penerapan Image Processing untuk Mengetahui Tingkat Kematangan Kopi Menggunakan Algoritma K-Nearest Neighbor (KNN) pada Perkebunan Kopi Malabar Bandung," vol. 10, no. 3, pp. 3246–3252, 2023.
- [13] R. Indriati Purba, "ANALISIS DAN IMPLEMENTASI ALGORITMA GAUSSIAN DAN HIGH PASS FILTER UNTUK MENGHASILKAN CITRA HIGH DYNAMIC RANGE," *J. Pembang. Wil. Kota*, vol. 1, no. 3, pp. 82–91, 2014.
- [14] M. Orisa and T. Hidayat, "Analisis Teknik Segmentasi Pada Pengolahan Citra," *J. Mnemon.*, vol. 2, no. 2, pp. 9–13, 2019, doi: 10.36040/mnemonic.v2i2.84.
- [15] J. Ulfah and N. Nurdin, "Implementasi Metode Deteksi Tepi Canny Untuk

- Menghitung Jumlah Uang Koin Dalam Gambar Menggunakan Opencv,” *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3, pp. 420–426, 2023, doi: 10.23960/jitet.v11i3.3147.
- [16] C. A. Lorentius, R. Adipranata, and A. Tjondrowiguno, “Pengenalan Aksara Jawa dengan Menggunakan Metode Convolutional Neural Network,” *e-Proceeding Eng.*, vol. 7, no. 1, pp. 2558–2567, 2020.
- [17] B. Santoso and R. P. Kristianto, “Implementasi Penggunaan Opencv Pada Face Recognition Untuk Sistem Presensi Perkuliahan Mahasiswa,” *Sistemasi*, vol. 9, no. 2, p. 352, 2020, doi: 10.32520/stmsi.v9i2.822.
- [18] E. Fadly, S. Adi Wibowo, and A. Panji Sasmito, “Sistem Keamanan Pintu Kamar Kos Menggunakan Face Recognition Dengan Telegram Sebagai Media Monitoring Dan Controlling,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 5, no. 2, pp. 435–442, 2021, doi: 10.36040/jati.v5i2.3796.
- [19] A. D. Mulyanto, “Pemanfaatan Bot Telegram Untuk Media Informasi Penelitian,” *Matics*, vol. 12, no. 1, p. 49, 2020, doi: 10.18860/mat.v12i1.8847.
- [20] Y. Marliza and R. Aisuwarya, “Rancang Bangun Sistem Pemberian Pakan Ikan Secara Otomatis Berdasarkan Perilaku Ikan Menggunakan Kamera Berbasis Mini PC,” *ISSN*, vol. 2, no. 01, pp. 11–19, 2021, doi: 10.25077/chipset.2.01.11-19.2021.

DAFTAR LAMPIRAN

Lampiran 1: Daftar Riwayat Hidup Penulis

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Harry Samara
Tempat, Tanggal Lahir : Sungailiat, 16 April 2002
Alamat Rumah : Jl. Batin Tikal GG. Batam Kec.
Pemali
Hp : 085768458324
Email : harrysamara123@gmail.com
Jenis Kelamin : Laki-Laki
Agama : Islam



2. Riwayat Pendidikan

SD Negeri 13 Kelapa Kamplit
SMP Negeri 1 Kelapa Kamplit
MAN Insan Cendekia Bangka Tengah

Sungailiat, 19 Januari 2024

Harry Samara

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Mauliana Fardiyatulh
Tempat, Tanggal Lahir : Belinyu, 9 Juni 2002
Alamat Rumah : Padang Labu, Kel. Riding Panjang,
Kec. Belinyu
Hp : 083891964316
Email : maulianario2410@gmail.com
Jenis Kelamin : Perempuan
Agama : Islam



2. Riwayat Pendidikan

SD Negeri 25 Belinyu
SMP Negeri 2 Belinyu
SMK YPN Belinyu

Sungailiat, 19 Januari 2024

Mauliana Fardiyatulh

Lampiran 2: *Source Code* Program Utama Ikan Mas Koki

```
# Library -----
import os,sys,time,traceback
from math import hypot # library math digunakan untuk rumus
matematika

import numpy as np # library numpy digunakan untuk
perhitungan array pada olah citra
import cv2 # library opencv digunakan untuk olah citra

import frame_capture # digunakan untuk init camera, stream
opencv dan frame
import frame_draw # digunakan untuk menampilkan text dan
line

import asyncio
from telegram import Bot

# init camera
camera_id = 0 # ID Camera
camera_width = 2560 # Resolusi panjang pixel camera
camera_height = 1440 # Resolusi tinggi pixel camera
camera_frame_rate = 60 # Resolusi FPS camera
camera_fourcc = cv2.VideoWriter_fourcc(*"MJPG") # Format
Video Camera

# filter object
auto_percent = 0.2 # Seting Filter untuk object apabila
semakin kecil nilainya maka object" kecil akan terdeteksi
dan apabila nilai semakin besar maka hanya object" besar
yang akan terdeteksi sehingga object kecil akan terfilter
auto_threshold = 100 # seting threshold
auto_blur = 5 # seting blur

# clearing view object
norm_alpha = 0 # setting nilai alpha pada normalize
norm_beta = 255 # setinggan nilai beta pada normalize
```

```

total_jumlah_pakan = 0
total_bobot = 0
TOKEN_BOT = '6839833210:AAHR7Sz46Zh1BezC8XspuCed5D4uW7hHwQ0'

# program untuk membaca file config
configfile = 'config.csv'
if os.path.isfile(configfile):
    with open(configfile) as f:
        for line in f:
            line = line.strip()
            if line and line[0] != '#' and (',' in line or
            '=' in line):
                if ',' in line:
                    item,value = [x.strip() for x in
line.split(',',1)]
                elif '=' in line:
                    item,value = [x.strip() for x in
line.split('=',1)]
                else:
                    continue
                if item in 'camera_id camera_width
camera_height camera_frame_rate camera_fourcc auto_percent
auto_threshold auto_blur norm_alpha norm_beta'.split():
                    try:
                        exec(f'{item}={value}')
                        print('CONFIG:',(item,value))
                    except:
                        print('CONFIG ERROR:',(item,value))

if len(sys.argv) > 1:
    camera_id = sys.argv[1]
    if camera_id.isdigit():
        camera_id = int(camera_id)

camera = frame_capture.Camera_Thread()
camera.camera_source = camera_id # SET THE CORRECT CAMERA
NUMBER
camera.camera_width = camera_width

```



```

camera.camera_height = camera_height
camera.camera_frame_rate = camera_frame_rate
camera.camera_fourcc = camera_fourcc

camera.start()

width = camera.camera_width
height = camera.camera_height
area = width*height
cx = int(width/2)
cy = int(height/2)
dm = hypot(cx,cy) # max pixel distance
frate = camera.camera_frame_rate
print('CAMERA:',[camera.camera_source,width,height,area,frate])

draw = frame_draw.DRAW()
draw.width = width
draw.height = height

pixel_base = 10 # nilai perhitungan per 10 pixel

cal_range = 216 # nilai kalibrasi untuk panjang view
setengah kamera yaitu 216 mm

cal = dict([(x,cal_range/dm) for x in
range(0,int(dm)+1,pixel_base)]) # variable array untuk
pembacaan file kalibrasi

# Program membaca file kalibrasi
calfile = 'kalibrasi.csv'
if os.path.isfile(calfile):
    with open(calfile) as f:
        for line in f:
            line = line.strip()
            if line and line[0] in ('d',):
                axis,pixels,scale = [_.strip() for _ in
line.split(',',2)]
                if axis == 'd':

```

```

        print(f'LOAD: {pixels} {scale}')
        cal[int(pixels)] = float(scale)

async def send_message(chat_id, text):
    bot = Bot(TOKEN_BOT)
    await bot.sendMessage(chat_id, text)

async def send_telegram_message(chat_id, total_bobot,
total_jumlah_pakan):
    message_text = (
        f"Jumlah Ikan Terdeteksi : 1\n"
        f"Total Bobot : {total_bobot:.2f}\n"
        f"Total Jumlah Pakan : {total_jumlah_pakan:.2f}\n"
    )
    await send_message(chat_id, message_text)

def conv(x,y):
    try:
        d = distance(0,0,x,y)
        scale = cal[baseround(d,pixel_base)]
        return x*scale,y*scale
    except Exception as error:
        return 1,1

def baseround(x,base=1):
    return int(base * round(float(x)/base))

def distance(x1,y1,x2,y2):
    return hypot(x1-x2,y1-y2)

framenam = " Project Ikan Koki "
cv2.namedWindow(framenam,flags=cv2.WINDOW_NORMAL|cv2.WINDOW
_GUI_NORMAL)

async def main():
    global total_jumlah_pakan, total_bobot

    while 1:

```

```

# Fungsi Get Frame
frame0 = camera.next(wait=1)
if frame0 is None:
    time.sleep(0.1)
    continue

cv2.normalize(frame0,frame0,norm_alpha,norm_beta,cv2.NORM_MINMAX) # Fungsi Untuk Clearing view object agar hasil video stream bisa clear atau cerah

    text = []

    mouse_mark = None

    frame1 = cv2.cvtColor(frame0,cv2.COLOR_BGR2GRAY) #
Opendv set object Convert ke gray

    frame1
cv2.GaussianBlur(frame1,(auto_blur,auto_blur),0) # Opendv
Set blur object

    frame1
cv2.threshold(frame1,auto_threshold,255,cv2.THRESH_BINARY)[1
] # Opendv Set Threshold dengan binary atau hitam putih

    frame1 = ~frame1

    contours,nada
cv2.findContours(frame1,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE) # Opendv deteksi countour (apabila ada object maka akan ada countour)

    cnt = 0

    objek_terdeteksi = False
    objek_ada = False

```

```

# Program pendeteksian object countours yang
terdeteksi
for c in contours:

    x1,y1,w,h = cv2.boundingRect(c) # Fungsi
mendeteksi sisi tepi object dalam bentuk sudah kotak
    x2,y2 = x1+w,y1+h
    x3,y3 = x1+(w/2),y1+(h/2)

    percent = 100*w*h/area

    if percent < auto_percent: # Apabila percent
object dibawah 0.2 percent maka object tidak di proses
(Object Terlalu Kecil)
        continue

    elif percent > 60: # Apabila percent object
diatas 60 percent maka object tidak di proses (Object
Terlalu Besar)
        continue

    x1c,y1c = conv(x1-(cx),y1-(cy)) # Convert Pixel
to mm
    x2c,y2c = conv(x2-(cx),y2-(cy)) # Convert Pixel
to mm
    xlen = abs(x1c-x2c) # Hitung panjang X dalam
satuan mm
    ylen = abs(y1c-y2c) # Hitung panjang Y dalam
satuan mm

    panjang = xlen/10 # Panjang = x/10 ( karena
satuan cm, apabila satuan mm maka panjang = x )
    lebar = ylen/10 # Lebar = y/10 ( karena satuan
cm, apabila satuan mm maka Lebar = y )
    bobot = (0.015 * lebar) * (panjang ** 3.062) #
Rumus Bobot
    jumlah_pakan = 0
    if panjang <= 20: # Jika Panjang ikan dibawah
20cm

```

```

        jumlah_pakan = 0.05 * bobot # Rumus: 5% x
berat ikan
    else: # Jika Panjang ikan diatas 20cm
        jumlah_pakan = 0.02 * bobot # Rumus: 2% x
berat ikan

    total_bobot = 0
    if objek_ada == False:
        total_bobot += bobot
        objek_ada = True
        await send_telegram_message('5306168160',
total_bobot, total_jumlah_pakan)
    else:
        total_bobot = 0

    total_jumlah_pakan = 0
    if objek_terdeteksi == False:
        total_jumlah_pakan += jumlah_pakan
        objek_terdeteksi = True
        await send_telegram_message('5306168160',
total_bobot, total_jumlah_pakan)
    else:
        total_jumlah_pakan = 0

draw.rect(frame0,x1,y1,x2,y2,weight=2,color='red') # Program
untuk draw kotak pada object dengan warna merah
    draw.add_text(frame0,f'{panjang:.2f}Cm',x1-((x1-
x2)/2),min(y1,y2)-8,center=True,color='red') # Program
Menampilkan panjang dengan warna merah
    draw.add_text(frame0,f'{lebar:.2f}Cm',x1-
4,(y1+y2)/2,middle=True,right=True,color='red') # Program
Menampilkan Lebar dengan warna merah
    draw.add_text(frame0,f'Bobot:
{bobot:.2f}gr',x3,y2+8,center=True,top=True,color='blue') #
Program Menampilkan bobot dengan warna biru
    draw.add_text(frame0,f'Jumlah          Pakan:
{jumlah_pakan:.2f}gr',x3,y2+32,center=True,top=True,color='b
lue') # Program Menampilkan jumlah pakan dengan warna biru

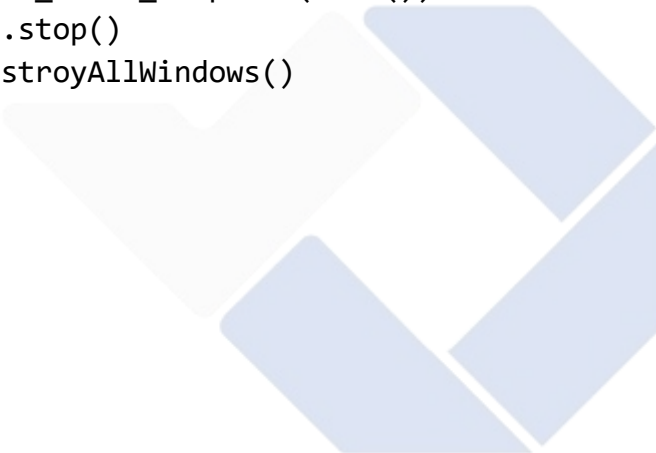
```

```
        draw.add_text(frame0, f'Total Jumlah Pakan:
{total_jumlah_pakan:.2f}gr', 20, 40, color='purple')
        draw.add_text(frame0, f'Total Bobot:
{total_bobot:.2f}gr', 20, 60, color='purple')

cv2.imshow(framename,frame0)# tampilkan Frame Camera
Stream

key = cv2.waitKey(1) & 0xFF

if key in (27,113):
    break
if __name__ == "__main__":
    loop = asyncio.get_event_loop()
    loop.run_until_complete(main())
    camera.stop()
    cv2.destroyAllWindows()
    exit()
```



Lampiran Nomor : 020/PROYEKAKHIR/DIV/2024

SURAT PERNYATAAN

Kami yang bertanda tangan dibawah ini telah menyelesaikan Proyek Akhir yang berjudul:

Implementasi Teknologi *Image Processing* Untuk Identifikasi Pengukuran Bobot Ikan Hias Dalam Penentuan Jumlah Pakan

Oleh:

1. Harry Samara /1052014
2. Mauliana Fardiyatulh /1052017


Dengan ini menyatakan bahwa isi laporan akhir proyek akhir sama dengan *hardcopy*.

Demikian surat pernyataan ini dibuat dengan sebenar-benarnya.

Sungailiat, 24 Januari 2024

1. Harry Samara ()
2. Mauliana Fardiyatulh ()

Mengetahui,
Pembimbing 1,


(Irwan, M.Sc., Ph.D)

Pembimbing 2,


(Yudy, M.T)