

**PENERAPAN *IMAGE PROCESSING* UNTUK MENDETEKSI  
PERGERAKAN IKAN KOKI DENGAN METODE *CASCADE*  
*CLASSIFIER***

**SKRIPSI**

untuk memenuhi salah satu persyaratan  
mencapai derajat Sarjana S1



**Disusun oleh:**

**Muhammad Mahbub Al Choedori**

**14524116**

**Jurusan Teknik Elektro  
Fakultas Teknologi Industri  
Universitas Islam Indonesia  
Yogyakarta**

**2021**

# LEMBAR PENGESAHAN

PENERAPAN *IMAGE PROCESSING* UNTUK MENDETEKSI PERGERAKAN IKAN

KOKI DENGAN METODE *CASCADE CLASSIFIER*

**TUGAS AKHIR**

**Diajukan sebagai Salah Satu Syarat untuk Memperoleh  
Gelar Sarjana Teknik  
pada Program Studi Teknik Elektro  
Fakultas Teknologi Industri  
Universitas Islam Indonesia**

**Disusun oleh:**

**Muhammad Mahbub Al Choedori  
144524116**

الجامعة الإسلامية  
الاستاذ الدكتور

**Yogyakarta, 10-November-2021**

**Menyetujui,**

**Pembimbing 1**

22  


**Almira Budiyanto, S.Si., M.Eng.  
155240103**

**Pembimbing 2**

3180y!  


**Elvira Sukma Wahyuni S.Pd., M.Eng.  
155231301**

# LEMBAR PENGESAHAN

## SKRIPSI

**PENERAPAN *IMAGE PROCESSING* UNTUK MENDETEKSI PERGERAKAN IKAN**

**KOKI DENGAN METODE *CASCADE CLASSIFIER***

Dipersiapkan dan disusun oleh:

**Muhammad Mahbub Al Choedori**

**14524116**

Telah dipertahankan di depan dewan penguji

Pada tanggal: **14 Desember 2021**

Susunan dewan penguji

Ketua Penguji : **Elvira Sukma Wahyuni S.Pd., M.Eng.,** 

Anggota Penguji 1: **Firdaus S. T., M.T. Ph.D.,** 

Anggota Penguji 2: **Dwi Ana Ratna Wati S.T., M.Eng.,** 

Skripsi ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Sarjana

Tanggal: **14 Januari 2022**

**Ketua Program Studi Teknik Elektro**



**Yusuf Aziz Amrullah, S.T., M.Sc., Ph.D.**

## PERNYATAAN

Dengan ini Saya menyatakan bahwa:

1. Skripsi ini tidak mengandung karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan Saya juga tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.
2. Informasi dan materi Skripsi yang terkait hak milik, hak intelektual, dan paten merupakan milik bersama antara tiga pihak yaitu penulis, dosen pembimbing, dan Universitas Islam Indonesia. Dalam hal penggunaan informasi dan materi Skripsi terkait paten maka akan diskusikan lebih lanjut untuk mendapatkan persetujuan dari ketiga pihak tersebut diatas.

Yogyakarta, 20 Januari 2022



Muhammad Mahbub Al Choedori

## KATA PENGANTAR

*Alhamdulillahirobil'alamin*, puji syukur kehadirat Allah SWT yang telah melimpahkan karunia dan rahmatnya-Nya kepada kita semua, sehingga penulis diberikan kelancaran dan kemudahan dalam menyelesaikan skripsi dengan judul “Penerapan *Image Processing* untuk mendeteksi pergerakan ikan koki dengan metode *Cascade Classifier*”. Sholawat dan salam juga tercurahkan kepada Nabi Muhammad SAW beserta keluarga, sahabat, dan para pengikutnya sampai akhir zaman nanti.

Skripsi dibuat sebagai syarat guna mendapatkan gelar Sarjana S1 pada Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Indonesia. Yangmana dibutuhkan perjuangan dan kerja keras dalam menyelesaikan skripsi ini, hal tersebut tak lepas dari bimbingan, motivasi dan semangat dari para dosen, sahabat, dan orang tua penulis.

Penulis mengungkapkan banyak terimakasih kepada:

1. Bapak Yusuf Aziz Amrullah, S.T., M.Sc., Ph.D selaku ketua jurusan Teknik Elektro, Universitas Islam Indonesia.
2. Ibu Almira Budiyanto, S.Si., M.Eng selaku pembimbing pertama, dan Ibu Elvira Sukma Wahyuni S.Pd., M.Eng selaku pembimbing kedua penyusunan tugas akhir, yang selalu memberian masukan, pengarahan, dan motivasi mengenai berbagai hal sampai terselesaikannya tugas akhir ini.
3. Kedua orang tua penulis Bapak Mochammad Rifqi Nasir dan Ibu Zahirotus Saniyah yang selalu memberikan doa dan motivasi sampai tugas akhir ini dapat terselesaikan.
4. Teman-teman Teknik Elektro angkatan 2014 Universitas Islam Indonesia yang tidak dapat penulis sebutkan namanya satu persatu.
5. Teman-teman KKN Universitas Islam Indoneisa Unit 212 Desa Semawung, Kabupaten Purworejo angkatan 55 yaitu Rizky, Ilham, Adnan, Andin, Beby, Laili, dan Kiky yang selalu memotivasi dalam penyelesaian tugas akhir.
6. Dan juga berbagi pihak yang telah membantu penulis dalam memfasilitasi dan membantu dalam penyusunan tugas akhir ini.

## ABSTRAK

Ikan hias merupakan jenis yang banyak dicari sebagian orang karena keindahannya, namun kebanyakan pengusaha ikan hias belum mampu melakukan budidaya karena terkendala modal, pengetahuan dan kurang teknologi dalam penanganannya. Perburuan ikan hias baik air tawar dan laut sering mengabaikan kelestarian terumbu karang, karena masih ada sebagian pemburu menggunakan peledak sebagai alat yang mereka gunakan untuk mencari ikan hias atau menggunakan alat penyengat listrik yang akan mengurangi dari kualitas daya hidup ikan tersebut. Melihat masalah mengenai perburuan atau pencarian ikan hias baik tawar maupun laut, skripsi ini mencoba memberi solusi mengenai masalah tersebut, dengan menerapkan teknologi *Image Processing*. Ada beberapa penelitian terdahulu mengenai aplikasi *image processing* sebagai pendeteksi objek, yang umumnya diterapkan pada deteksi wajah, pengenalan huruf angka dan deteksi warna. Ada penelitian serupa akan tetapi lebih membahas pada objek yang diam seperti foto ikan atau citra ikan serta mendeteksi jenis ikan berdasarkan mata ikan untuk dikenali. Skripsi ini mencoba menerapkan *image processing* dalam pengenalan dan deteksi pergerakan ikan menggunakan *software* MATLAB dalam sebuah video, dengan bantuan *software* MATLAB sebagai pemrograman *Image Processing* dan training *Cascade Classifier*, dilaksanakan deteksi ikan koki yang berada pada sebuah file video untuk dikenali dan dideteksi keberadaanya. Hasil pengujian yang dilakukan sistem dapat mendeteksi keberadaan ikan koki. Dengan kesimpulan tingkat keberhasilan sistem dalam mendeteksi ikan koki adalah 85% didapat dengan jumlah stage 50 dan citra negatif 200. Penelitian ini bertujuan memberikan solusi dan aplikasi mengenai deteksi pergerakan ikan dan pengenalan jenis ikan dengan menggunakan *software* MATLAB dan *Image Processing*, serta memberi gambaran bagi pengusaha ikan hias mengenai pencarian dan deteksi ikan menggunakan *image processing*. Metodologi dalam skripsi ini meliputi pengumpulan data, *preprocessing*, pengusulan model, pengujian model, dan analisis hasil percobaan.

**Kata kunci:** MATLAB, *Image Processing*, *Cascade Classifier*, deteksi ikan koki

# DAFTAR ISI

LEMBAR PENGESAHAN.....	i
PERNYATAAN.....	iii
KATA PENGANTAR.....	iv
ABSTRAK .....	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR .....	viii
DAFTAR TABEL.....	ix
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	3
BAB 2 TINJAUAN PUSTAKA .....	4
2.1 Studi Literatur .....	4
2.2 Tinjauan Teori.....	5
2.2.1 MATLAB.....	5
2.2.2 Citra atau Gambar.....	5
2.2.3 <i>Image Processing</i> .....	6
2.2.4 <i>Cascade Classifier</i> .....	6
BAB 3 METODOLOGI.....	8
3.1 Pengumpulan Data .....	8
3.2 <i>Preprocessing</i> .....	9
3.3 Pengusulan Model.....	11
3.4 Pengusulan Model.....	13

3.5 Analisis Hasil Percobaan.....	16
BAB 4 HASIL DAN PEMBAHASAN.....	18
4.1 Pengujian dengan 10 gambar positif ikan koki .....	18
4.2 Pengujian dengan 30 gambar positif ikan koki .....	21
BAB 5 KESIMPULAN DAN SARAN.....	23
5.1 Kesimpulan.....	23
5.2 Saran.....	23
DAFTAR PUSTAKA .....	24
LAMPIRAN.....	26





## DAFTAR GAMBAR

Gambar 2.1 Sistem kerja training <i>Cascade Classifier</i> [8].....	7
Gambar 3.1 Folder penyimpanan citra positif, negatif, dan video.....	9
Gambar 3.2 Isi folder positif_10 dan positif_30 .....	9
Gambar 3.3 ROI pada citra positif_10 .....	10
Gambar 3.4 ROI pada citra positif_30 .....	10
Gambar 3.5 Eksport ROI pada Workspace .....	10
Gambar 3.6 <i>Flow chart</i> atau skema model yang diusulkan .....	12
Gambar 3.7 Model <i>Haar Feature</i> yang dipakai.....	13
Gambar 3.8 program <i>training_process.m</i> .....	14
Gambar 3.9 Tahap <i>training stage</i> .....	15
Gambar 3.10 Program <i>main_prog.m</i> .....	15
Gambar 3.11 Positif Benar .....	16
Gambar 3.12 Positif Palsu.....	16
Gambar 3.13 Negatif Palsu .....	17
Gambar 4.1 pengujian Positif Benar (PB).....	19
Gambar 4.2 pengujian Positif Palsu (PP).....	20
Gambar 4.3 pengujian Negatif Palsu (NP).....	21

## DAFTAR TABEL

Tabel 3.1 Tabel variabel <i>roi_10.m</i> .....	11
Tabel 4.1 Pengujian dengan 10 gambar positif ikan koki .....	18
Tabel 4.2 Pengujian dengan 30 gambar positif ikan koki .....	21



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Ikan hias merupakan jenis yang banyak dicari sebagian orang khususnya penghobi karena keindahannya, namun kebanyakan pengusaha ikan hias belum mampu melakukan budidaya karena terkendala modal, pengetahuan dan kurang teknologi dalam penanganannya. Perburuan ikan hias baik tawar dan laut sering mengabaikan lingkungan hidup ikan, karena masih ada sebagian pemburu ikan menggunakan peledak sebagai alat yang digunakan untuk mencari ikan hias tanpa mengenali jenis ikan apa yang nantinya mereka dapatkan [1], serta menggunakan alat pernyengat listrik yang akan mengurangi kualitas daya hidup ikan. Melihat masalah mengenai perburuan atau pencarian ikan hias, skripsi ini mencoba memberi solusi mengenai masalah tersebut, yaitu dengan menerapkan teknologi *image processing*.

*Image Processing* atau pengolahan citra merupakan sebuah proses pengolahan *pixel-pixel* dalam gambar atau citra digital yang digunakan untuk tujuan tertentu, seperti memperbaiki kualitas gambar [2]. Ada beberapa penelitian terdahulu mengenai aplikasi *image processing* sebagai deteksi objek, salah satu contohnya adalah penelitian yang telah dilakukan oleh M.M. Saberioon mengenai pelacakan ikan dengan menggunakan sensor cahaya berupa kamera kinect, akan tetapi penelitian yang dilakukan hanya melacak pergerakan berdasarkan perpindahan ikan dalam akuarium [3].

Dengan mempertimbangkan mahalnya sensor yang digunakan serta masalah yang dihadapi pengusaha ikan hias, penulis memutuskan membuat sebuah sistem yang dapat mengenali dan mendeteksi pergerakan ikan hias. Dengan mendeteksi pergerakan ikan, diharapkan akan mengetahui dimana ikan yang sedang dicari, kemudian untuk skripsi ini objek yang diamati adalah ikan koki. Memusatkan penelitian menggunakan *image processing* karena objek yang akan diteliti berupa citra ikan yang dapat diolah gambar atau citranya. Untuk pemrogramannya menggunakan *software* MATLAB dan metode yang dipakai ialah training *Cascade Classifier*, metode ini menerapkan sistem training berupa cita positif dan negatif, untuk menentukan objek yang dipilih yaitu berdasarkan input pada sistem, dan output berupa pengambilan keputusan jika objek yang dipilih sesuai input pada *frame*.

Sistem ini bekerja dengan menggunakan file video sebagai frame lingkungan objek ikan yang akan dideteksi keberadaannya. Sistem akan melakukan training dengan membaca input *database* citra positif dan citra negatif yang diproses dalam *software* MATLAB, input citra positif

berisi beberapa citra ikan yang ingin dideteksi yang diambil citranya dari berbagai sisi, hasil training akan memberikan keputusan berupa ada atau tidaknya objek yang sama dengan input pada frame video, frame ini berupa video lingkungan habitat tempat tinggal ikan di akuarium.

Skripsi ini merupakan tahap awal karena belum diterapkan langsung secara *realtime* dengan *hardware* seperti kamera yang dapat bekerja merekam di kedalaman air. Serta memberi solusi mengenai deteksi ikan hias dengan fokus pada *software*, memberi kemudahan dalam penelitian dari segi biaya yang digunakan, tanpa harus menggunakan perangkat keras seperti kamera kinect dalam penelitian sebelumnya. Memberi solusi dan gambaran bagaimana cara mengenali dan mendeteksi jenis ikan hias dengan menggunakan data berupa citra ikan yang ingin dicari. Skripsi ini diharapkan dapat dikembangkan oleh peneliti atau pelajar yang sedang menekuni bidang *image processing*, khususnya pendeteksian objek.

## **1.2 Rumusan Masalah**

1. Bagaimana merancang sistem agar dapat mendeteksi keberadaan ikan koki dengan *Image Processing* ?
2. Bagaimana performa metode training *Cascade Classifier* dengan bantuan *software* MATLAB dalam mengenali ikan koki ?

## **1.3 Batasan Masalah**

Adapun batasan-batasan masalah pada skripsi ini ialah sebagai berikut:

1. Hanya mengenali ikan berdasarkan data citra ikan dan mendeteksi keberadaan ikan.
2. Objek yang dideteksi berupa ikan hias jenis koki yang berada pada sebuah video.
3. Sistem masih bersifat simulasi, serta gambaran sebelum pengaplikasian pada *hardware*.
4. Hanya ada satu ikan koki yang akan dideteksi.
5. Hasil yang ditampilkan hanya berupa tampilan video dengan tanda deteksi ikan koki.

## **1.4 Tujuan Penelitian**

Menciptakan sebuah sistem yang dapat mengenali dan mendeteksi keberadaan ikan koki dengan menggunakan bantuan *software* MATLAB dan *Image Processing* dengan metode training *Cascade Classifier* dalam sebuah file video.

## 1.5 Manfaat Penelitian

1. Sebagai pre-studi mengenai bagaimana mendeteksi dan mengenali suatu objek dengan *Image Processing*, khususnya ikan atau benda bergerak dalam sebuah file video.
2. Memberi gambaran bagi pengusaha dan pemburu ikan dalam mencari ikan hias, tanpa menggunakan bahan peledak yang dapat merusak lingkungan hidup ikan.
3. Memberi edukasi bagi pelajar dan masyarakat dalam mengenal jenis ikan, khususnya ikan hias dengan sebuah sistem yang dapat mengenali dan mendeteksi.
4. Memudahkan pencarian jenis ikan yang nelayan atau pemburu ikan hias inginkan, karena deteksi fokus pada satu jenis ikan yang dicari.



## BAB 2

### TINJAUAN PUSTAKA

#### 2.1 Studi Literatur

Ada beberapa jurnal dan paper yang digunakan sebagai referensi dalam membuat skripsi ini, yang berkaitan dengan deteksi objek menggunakan program matlab. Seperti melacak objek bergerak dengan urutan *frame* video, dengan mengimplemantasikan aliran optik (*Horn-Schunck*) pada simulink matlab, yang biasanya dapat digunakan dalam keamanan, komunikasi, dan kontrol lalu lintas [4]. Ada juga metode dengan mengambil foto yang dijadikan citra uji untuk dideteksi beberapa titik pada citra tersebut kemudian dibandingkan dengan citra referensi, dengan menggunakan algoritma *Speeded up Robust Features (SURF)*, yang dapat mendeteksi posisi objek yang menarik pada citra asli dengan menggunakan transformasi geometrik [5].

Ada juga penelitian yang memaparkan tentang segmentasi citra digital ikan menggunakan metode *thresholding* oleh Max R. Kumaesh yang fokus pada objek mata ikan untuk dikenali. Prosesnya ialah menginput citra ikan yang ingin diproses kemudian dikonversi ke citra *grayscale*, selanjutnya menentukan nilai *thresholding* dengan tipe local dan global, dilanjutkan dengan proses *segmentasi* (pemisahan objek dengan *background*) terhadap citra *grayscale*, kemudian memilih hasil *segmentasi* yang ditandai dengan proses deteksi tepi dengan operator *Canny* yang ditajamkan citranya dengan proses dilasi, terakhir pembuatan *plot contour*. Hasilnya ialah proses *segmentasi* berhasil memisahkan objek mata ikan dengan menggunakan metode *thresholding local* [6].

Selanjutnya oleh Devit Bee membangun sistem untuk mendeteksi tingkat kesegaran ikan selar. Prosesnya ialah melakukan pemotongan atau *cropping* pada bagian citra mata ikan, kemudian disimpan dengan format *\*png*. Setelah semua citra tersimpan kemudian dilakukan perhitungan nilai rata-rata *grayscale* dan diperoleh persamaan yang dijadikan sebagai data training. Data training selanjutnya diinput ke sistem untuk dikonversi kedalam bentuk *grayscale* dan ditempatkan pada bagian tertentu bersamaan dengan histogram *rgb* karena menggunakan GUI, kemudian dilakukan perhitungan dengan menggunakan metode kuadrat terkecil. Dan yang terakhir ialah pencocokan antara citra uji dan citra hasil training, yang menghasilkan 125 citra ikan sesuai dan 25 tidak sesuai, jadi dari 150 citra ikan yang diambil maka 125 citra ikan dinyatakan segar dan 25 citra ikan tidak segar, dan presentasi dari keakuratan sistem ialah 83,333 % [7].

Penelitian yang lainnya ialah dari Indrabayu dengan menggunakan pengoahan citra untuk memeriksa kesegaran ikan bandeng, menggunakan metode mencari selisih nilai R, G, B dari citra latih (referensi) dengan citra uji (masukan) yang hasil dari selisih nilai RGB tadi dihitung

dengan menggunakan persamaan *Euclidian* dan dibandingkan dengan nilai *Threshold*. Datanya berupa ikan bandeng segar yang berjumlah 5 ekor. Citra referensi dihasilkan  $R = 160$ ,  $G = 35$ , dan  $B = 55$ . Dan dari hasil pengujian ini dibandingkan dengan pengujian dari nelayan yang diperoleh hasil sama, dan berarti tingkat keakuratan pengujian sistem dengan pengujian dari nelayan 100% sama. Dari hasil penelitian ini disimpulkan bahwa menggunakan bagian tubuh selain mata membuat penilaian semakin akurat [8].

Penelitian yang terakhir dan juga menjadi latar belakang tugas akhir ialah M. M. Saberioon. Pelacakan yang dilakukan yaitu melacak pergerakan ikan pada ruang 3D atau akuarium, dan tujuan dari penelitian ini adalah mengembangkan sistem pelacakan ikan pada ruang 3D dengan menggunakan sensor cahaya berupa kamera kinect. Hal yang dilakukan pertama kali ialah merekam ikan yang berenang bebas pada ruang 3D selama 10 menit, kemudian hasil rekaman diolah untuk mengidentifikasi posisi masing-masing ikan dalam ruang 3D ( $x$ ,  $y$ , dan  $z$ ) dalam frame, sehingga tercipta jalur perpindahan ikan. Hasil dari sistem yang telah dibuat ialah 98 % sistem akurat dalam melacak pergerakan ikan dalam ruang 3D [3].

Dari beberapa penelitian yang sudah dipaparkan, penulis membuat sebuah sistem untuk melengkapi dan menambahkan dari penelitian yang sudah ada yaitu sebuah sistem yang dapat mendeteksi dan mengenali objek berupa ikan koi pada sebuah video.

## **2.2 Tinjauan Teori**

### **2.2.1 MATLAB**

MATLAB merupakan sebuah bahasa tingkat tinggi yang digunakan untuk teknik komputasi dan analisis numerik. Dalam program MATLAB terdapat *library* dan toolbox *Image Processing* yang menyediakan seperangkat referensi algoritma dan aplikasi guna pemrosesan gambar, visualisasi, analisis, serta pengembangan algoritma hal ini membuat penulis memilih bahasa matlab untuk membangun sistem [9].

### **2.2.2 Citra atau Gambar**

Citra adalah sebuah gambaran dari sebuah objek atau secara matematis didefinisikan sebagai fungsi dua dimensi  $f(x, y)$ , dimana  $x$  dan  $y$  merupakan spasial (*plane*) dan  $f$  adalah nilai intensitas warna pada koordinat  $x$  dan  $y$ . Nilai  $f$ ,  $x$ , dan  $y$  merupakan nilai berhingga, sehingga bila nilainya bersifat kontinu maka citranya disebut citra analog, dan bila nilainya bersifat diskret maka disebut citra digital.

Citra digital umumnya berupa dua dimensi (2D) yang dinyatakan dalam bentuk matriks dengan jumlah elemen berhingga. Setiap elemen memiliki koordinat  $x$  dan  $y$  yang memiliki nilai, yang secara umum citra digital representasi dari pixel-pixel dalam ruang dua dimensi yang dinyatakan dalam matriks  $N$  baris dan  $M$  kolom. Seperti pada persamaan 2.1 berikut:

$$F = [f(x, y)] = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,M) \\ f(1,0) & f(1,1) & \cdots & f(1,M) \\ \vdots & \vdots & \cdots & \vdots \\ f(N,0) & f(N,1) & \cdots & f(N,M) \end{bmatrix} \quad (2.1)$$

Setiap elemen matriks citra disebut *pixel*, nilai setiap piksel  $f$  pada koordiant  $x$  dan  $y$  mempresentasikan warna dan dapat dikodekan dalam 24 bit untuk citra yang memiliki warna (dengan komponen warna RGB: R = *red*, G = *green* dan B = *blue*), 8 bit untuk citra gray-level atau 1 bit untuk citra biner [10].

### 2.2.3 Image Processing

Pengolahan citra atau *Image Processing* merupakan proses dari perbaikan sebuah citra atau gambar. Komponen dalam pengolahan citra berupa sensor akusisi citra dan digitalisasi citra, komputer, memori penyimpanan, perangkat visualisasi dan perangkat pencetakan citra, perangkat keras pengolahan citra untuk aplikasi khusus dan perangkat lunak pengolah citra.

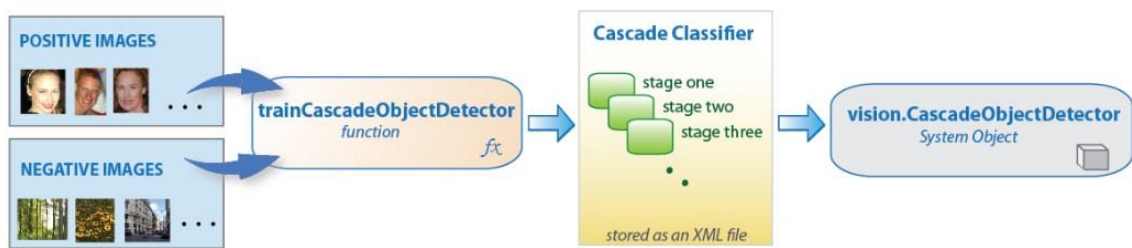
Akan tetapi dalam skripsi ini tidak semua komponen dipakai, misalnya perangkat keras pengolah citra, karena penulis ingin membangun sistem berupa simulasi deteksi objek tanpa bantuan perangkat keras aplikasi pengolah citra. Mengingat objek yang akan dikaji berupa video yang merupakan bagian dari citra, maka pengolahan citra digital juga merupakan pengolahan video digital [10].

### 2.2.4 Cascade Classifier

*Cascade Classifier* merupakan *cascade* objek yang fungsinya berada pada *Computer Vision Toolbox* MATLAB, mendeteksi kategori objek dengan rasio aspek yang tidak terlalu signifikan, contohnya ialah wajah, markah jalan, dan mobil yang dilihat dari satu sisi. Cara kerja dari *cascade classifier* adalah dengan melakukan beberapa tahapan, setiap tahapan dilatih dengan teknik *boosting*, teknik ini memberikan kemampuan untuk melatih pengklasifikasi menjadi akurat dengan mengambil nilai rata-rata. Tahap klasifikasi ditentukan saat jendela yang berganti baik dari positif maupun negatif, positif menunjukan bahwa objek yang dicari ada atau ditemukan dan negatif menunjukan bahwa objek yang dicari tidak ada. Tahapan dirancang untuk menolak sampel negatif



dengan cepat, dengan asumsi bahwa sebagian besar jendela tidak mengandung objek yang menarik, untuk menggambarkan klasifikasi dan tahapan *Cascade Classifier* dapat dilihat pada Gambar 2.1 berikut:



Gambar 2.1 Sistem kerja training *Cascade Classifier* [11]

Dalam *cascade classifier* diperlukan dua sampel positif *image* dan negatif *image* untuk ditraining dengan fungsi *trainCascadeObjectDetector*, dalam training ini kedua sampel akan dilatih menjadi beberapa tahapan atau stage sesuai input yang dimasukkan, semakin banyak tahapan maka semakin banyak dan lama pula training yang dilakukan, akan tetapi hasil yang didapat akan semakin baik. Sistem training ini berjalan dengan pengandaian sebagai berikut:

- Sebuah *Positif Benar* terjadi ketika sampel positif diklasifikasi atau dikelompokkan dengan benar.
- *Positif Palsu* terjadi ketika sampel negatif salah diklasifikasi atau dikelompokkan sebagai positif.
- *Negatif Palsu* terjadi karena kekeliruan pengelompokan antara sampel positif dan negatif.

Agar sistem berjalan baik maka *cascade* harus memiliki tingkat negatif palsu yang rendah, keseluruhan positif palsu dari *cascade classifier* adalah  $f^s$  dimana  $f$  adalah tingkat *false positif* per tahap dalam rentang (0 1), dan  $s$  adalah jumlah tahap. Demikian pula *true false* keseluruhan adalah  $t^s$ , dimana  $t$  adalah tingkat positif sejati per tahapan dalam rentang (0 1). Dengan demikian penambahan tahapan mengurangi tingkat positif palsu, tetapi juga mengurangi tingkat positif keseluruhan [12].

## BAB 3

### METODOLOGI

Metodologi yang dipakai dalam penelitian ini atau tugas akhir ini adalah sebagai berikut:

1. Pengumpulan Data

Tahap ini merupakan langkah awal dari penelitian, data pada penelitian ini berupa gambar positif dan gambar negatif, gambar positif memuat bagian ikan sebagai objek yang ingin dideteksi, dan gambar negatif yang tidak memuat bagian ikan.

2. *Preprocessing*

Tahap untuk mempersiapkan data yang diperoleh dari tahap pengumpulan data sebelum dilanjutkan ke tahap selanjutnya.

3. Pengusulan Model

Tahap ini membahas model yang akan digunakan dalam penelitian.

4. Pengujian Model

Dalam tahap ini menjelaskan teknik pengujian yang akan digunakan.

5. Analisis Hasil Percobaan

Tahap ini membahas mengenai hasil percobaan dan analisis dari penelitian yang telah dilakukan.

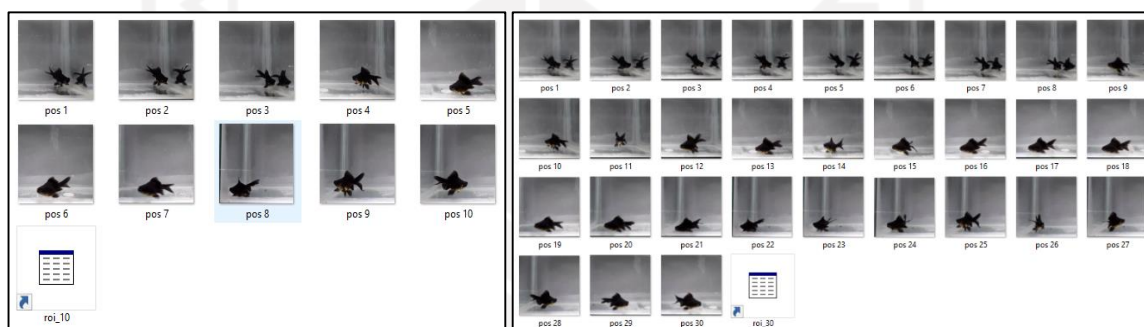
#### 3.1 Pengumpulan Data

*Datasheet* atau input dari penelitian ini meliputi *data training*, isi dari *datasheet* berupa kumpulan citra positif dan citra negatif. Citra positif adalah citra atau gambar yang berisi objek yang akan dikenali, dalam penelitian ini adalah gambar ikan koki, sedangkan citra negatif adalah citra atau gambar yang tidak memuat objek dari citra positif, atau gambar selain ikan koki. Pada penelitian ini jumlah citra positif akan dibagi menjadi dua bagian, yang bertujuan melihat tingkat keakuratan sistem jika dimasukan citra positif yang berbeda jumlahnya. Digunakan 10 gambar ikan koki dan 30 gambar ikan koki, 10 gambar ikan koi dimasukan pada folder yang diberi nama positif\_10 dan 30 gambar ikan koki sebagai pembandingan pada folder positif\_30, sedangkan citra negatifnya juga dibagi menjadi lima yaitu 10, 50, 100, 150, dan 200 gambar. Citra negatif dibagi menjadi lima bagian dengan tujuan menguji sistem apakah dengan lima sampel negatif yang berbeda jumlahnya akan mempengaruhi keakuratan sistem dalam bekerja. Dalam tahap ini juga dilakukan perekaman video sebagai *frame* atau latar yang akan dideteksi keberadaan ikan koki dalam akuarium. Semua *datasheet* yang telah didapatkan dikumpulkan dalam folder yang diberi nama berdasarkan fungsinya masing-masing. Seperti ditunjukkan pada Gambar 3.1 berikut:

file video	15/08/2020 02.38	File folder
negatif_10	15/08/2020 02.38	File folder
negatif_50	15/08/2020 02.38	File folder
negatif_100	15/08/2020 02.38	File folder
negatif_150	15/08/2020 02.38	File folder
negatif_200	15/08/2020 02.38	File folder
positif_10	15/08/2020 02.38	File folder
positif_30	15/08/2020 02.38	File folder
main_prog	14/08/2020 23.18	M File
training	15/08/2020 02.26	XML Document
training_process	15/08/2020 02.23	M File
VIDEO	21/08/2018 02.29	MP4 File

Gambar 3.1 Folder penyimpanan citra positif, negatif, dan video

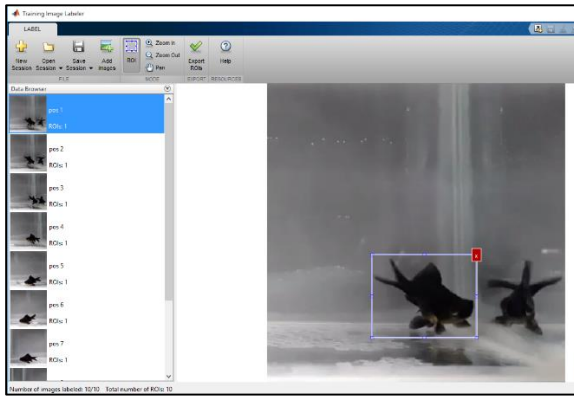
Citra positif berisi citra ikan koki yang diambil dari berbagai sisi dan dimasukkan kedalam folder positif\_10 dan positif\_30 terlihat pada Gambar 3.2 berikut:



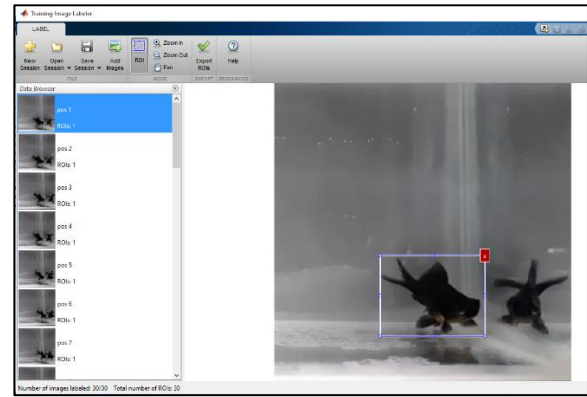
Gambar 3.2 Isi folder positif\_10 dan positif\_30

### 3.2 Preprocessing

Dalam tahap ini dipersiapkan semua tools yang diperlukan untuk melakukan pemisahan. Pemisahan atau *cropping* pada ikan dilakukan dengan membuat tanda ROI (*Region of Interest*) pada ikan tersebut. Untuk menandai area ikan digunakan tools *Training Image Labeler*, masukan citra atau gambar positif ikan dengan mengklik ikon *Add Images* setelah itu operator menandai ikan dengan cara menahan mouse (*drag mouse*) kemudian membuat persegi yang meliputi area ikan, hal ini berlaku untuk semua gambar ikan yang akan dijadikan citra positif, hasil yang diperoleh dari tahap ini adalah koordinat x, y yang mana merupakan tinggi dan lebar area ikan. Untuk gambaran proses penandaan ROI pada citra positif\_10 Gambar 3.3 dan positif\_30 Gambar 3.4 adalah seperti berikut:

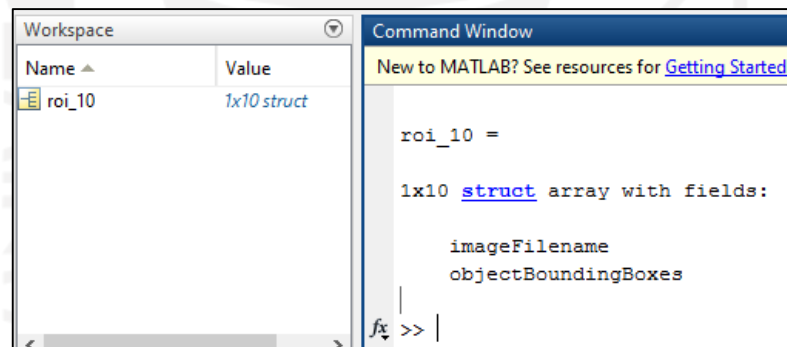


Gambar 3.3 ROI pada citra positif\_10



Gambar 3.4 ROI pada citra positif\_30

Setelah semua citra ikan di *cropping*, kemudian data ROI tadi dapat disimpan atau langsung di *ekspor*, dalam skripsi ini menyimpan data ROI citra koi dengan nama *roi\_10.m* dan *roi\_30* ke folder *positif\_10* dan *positif\_30* yang dapat dilihat pada Gambar 3.2, hal ini dilakukan jika ingin merubah ROI tidak perlu melakukannya dari awal cukup membuka citra yang sudah tersimpan. Setelah disimpan kemudian klik ikon *Eksport ROIs* maka file *roi\_10.m* akan masuk pada Workspace MATLAB seperti Gambar 3.5 berikut:



Gambar 3.5 Eksport ROI pada Workspace

Terlihat pada Command Window bahwa *roi\_koi.m* berisi  $1 \times 30$  struct array yang memiliki 2 *fields* yaitu *imageFilename* dan *objectBoundingBoxes*, jika kita klik *roi\_10* pada Workspace maka akan muncul tabel yang menunjukkan semua citra positif yang telah diberi label ROI, dan tabelnya seperti berikut:

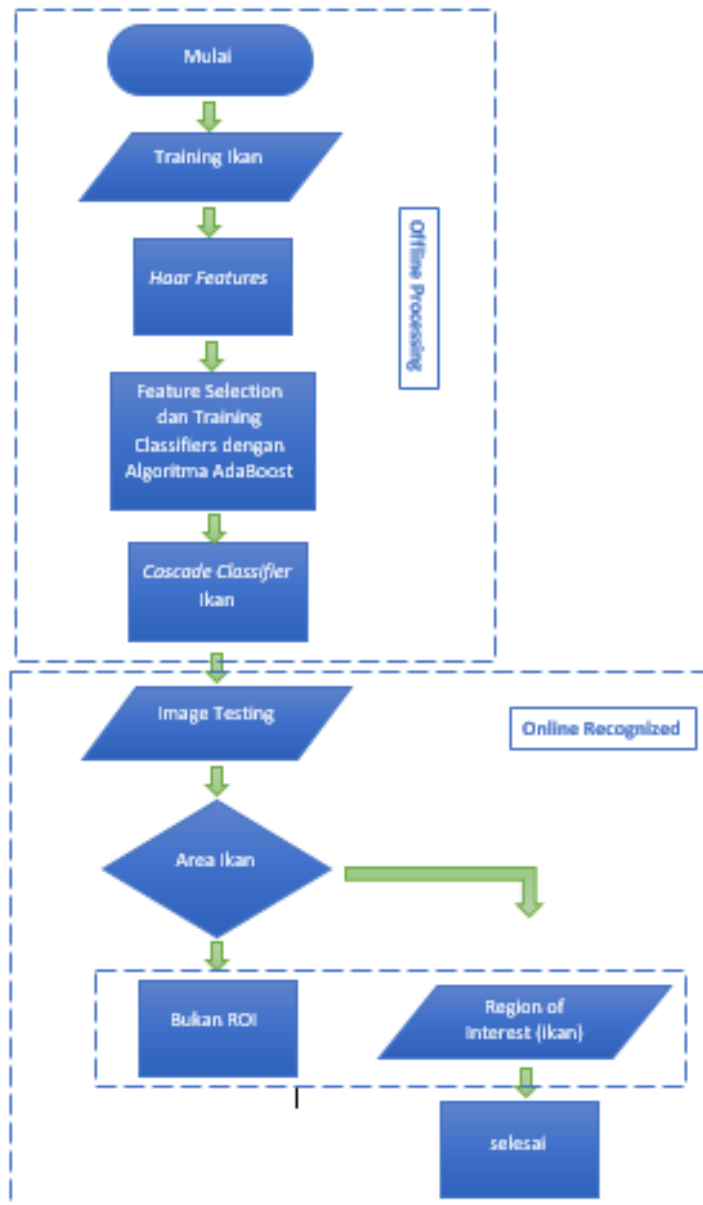
Fields	imageFilename	objectBoundingBoxes
1	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 1.jpg'	[179 291 177 141]
2	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 2.jpg'	[169 280 177 141]
3	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 3.jpg'	[219 297 177 141]
4	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 4.jpg'	[217 296 177 141]
5	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 5.jpg'	[195 335 177 141]
6	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 6.jpg'	[126 328 177 141]
7	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 7.jpg'	[131 330 177 141]
8	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 8.jpg'	[65 340 177 141]
9	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 9.jpg'	[94 305 177 141]
10	'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10\pos 10.jpg'	[149 299 177 141]

Tabel 3.1 Tabel variabel *roi\_10.m*

*Fields* adalah jumlah citra positif yang ditandai, *imageFilename* berisi nama dan letak *directory* atau penyimpanan dari setiap citra positif yang telah atau sudah ditandai, sedangkan *objectBoundingBoxes* berisi koordinat dan dimensi ROI. Contohnya adalah pada *field* 1 yang berisi nama citra positif 1 dengan letak 'F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif\_10\pos1.jpg' dan memiliki koordinat dan dimensi [179,291,177,141], angka 179, 291 adalah skala koordinat nilai  $x$  dan  $y$ , sedangkan 177, 141 adalah nilai lebar dan tinggi ROI.

### 3.3 Pengusulan Model

Untuk skema model yang diusulkan seperti pada Gambar 3.6, langkah pertama adalah melakukan training pada sampel positif dan sampel negatif untuk menghasilkan *cascade classifier*. Proses training menggunakan algoritma AdaBoost dilakukan pada file MATLAB *training\_process.m*, algoritma tersebut digunakan untuk menyeleksi fitur dan melatih *classifiers*, hasil training disimpan dalam file *training.xml*, tahap training ini merupakan tahap *Offline Processing*. Kemudian melakukan *Online Recognized* untuk mengetes citra yang sudah ditraining dengan video pada file *main\_prog.m* [13].

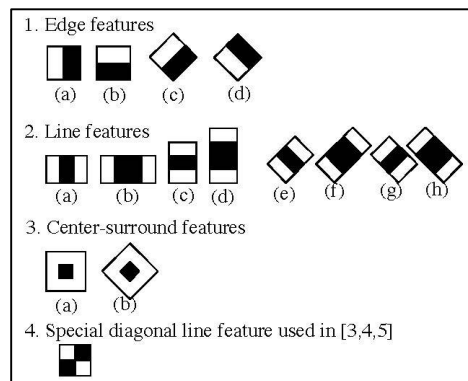


Gambar 3.6 Flow chart atau skema model yang diusulkan

1. Haar-like Feature dan Integral Image

Haar-like Feature memproses gambar dalam bentuk kotak-kotak yang didalamnya mengandung pixel. Dari kotak tadi diproses dan didapatkan perbedaan nilai (*threshold*), nilai tersebut dijadikan dasar dalam *image processing*. Kemudaiian untuk gambar yang bergerak atau video pemrosesan *pixel* tadi terjadi dalam waktu yang lama sehingga digunakan integral. Penggunaan integral menjadikan pemrosesan *pixel* tadi bekerja dengan cepat dan *realtime* pada sebuah benda [14].





Gambar 3.7 Model *Haar Feature* yang dipakai [15]

## 2. Algoritma *AdaBoost*

*Algoritma Adaboost* atau *Adaptive Boosting* merupakan salah satu algoritma yang digunakan untuk mengambil keputusan. Pada dasarnya algoritma ini digunakan untuk melakukan boosting kinerja klasifikasi dari sebuah algoritma pembelajaran yang sederhana, hal ini dilakukan untuk mengkombinasikan sekumpulan fungsi klasifikasi lemah untuk membentuk sebuah *classifier* yang kuat. *Classifier* lemah menentukan fungsi klasifikasi *threshold* yang optimal, sehingga jumlah minimum kesalahan klasifikasi bisa didapatkan.

## 3. *Cascade Classifier*

*Cascade Classifier* membantu detektor untuk meningkatkan kecepatan ketepatan deteksinya. *Cascade Classifier* dapat diibaratkan seperti seleksi bertingkat untuk mendapatkan area objek yang akan dideteksi. Seleksi ditingkat (*stage*) pertama sangat longgar. Kemudian jika berlanjut ke tingkat kedua sudah tidak selonggar tingkat pertama. Begitu seterusnya sehingga diperoleh stage yang pas untuk masalah yang akan dijadikan objek deteksi. Semakin tinggi stage tidak menjamin akan semakin tepat pendeteksiannya, karena semakin ketat proses penyaringan *sub-window* yang dilakukan. Seringkali yang terjadi adalah objek gagal terdeteksi. Oleh karena itu, ketepatan penentuan batas stage sewaktu proses training sangat berpengaruh untuk *classifier* dalam proses deteksi.

## 3.4 Pengujian Model

Pada tahap pengusulan model telah dibahas mengenai training dengan menggunakan sampel positif dan sampel negatif. Kemudian pada tahap pengujian model ini akan dibahas mengenai metode yang dipakai yaitu *Cascade Classifier* serta program yang digunakan untuk menjalankan metode *cascade classifier* ini.

Pada program matlab terdapat *Computer Vision Toolbox* yang menyediakan training menggunakan *cascade classifier* atau disebut *trainCascadeObjectDetector* pada matlab, fungsinya

adalah untuk melatih pengklasifikasi khusus. *Computer Vision Toolbox* matlab dapat mendeteksi kategori objek yang rasio aspeknya tidak bervariasi secara signifikan, objek rasio aspeknya tetap contoh pada skripsi ini ikan koki yang dilihat dari satu sisi. Sistem mendeteksi objek dalam gambar dengan menggeser jendela di atas gambar, kemudian menggunakan *cascade classifier* untuk memutuskan apakah jendela berisi objek yang diinginkan, yang dimaksud jendela adalah frame video.

*Cascade Classifier* terdiri dari beberapa tahapan, dimana setiap tahapan adalah ansambel atau kelompok objek yang lemah. Setiap tahapan dilatih dengan menggunakan teknik yang disebut *boosting*, meningkatkan kemampuan untuk melatih pengklasifikasi yang sangat akurat dengan mengambil rata-rata dari keputusan yang dibuat oleh objek yang lemah. Setiap tahap klasifikasi diberi label wilayah yang ditentukan oleh lokasi jendela geser sebagai positif atau negatif. Positif menunjukkan bahwa suatu objek ditemukan dan negatif menunjukkan tidak ada benda yang ditemukan. Jika labelnya negatif, klasifikasi wilayah ini selesai, dan detektor menggeser jendela ke lokasi berikutnya. Jika labelnya positif klasifikasi akan berlanjut ke tahap berikutnya. Detektor melaporkan objek yang ditemukan dilokasi jendela saat ini.

Untuk menjalankan metode training *cascade classifier* pada matlab adalah dengan menggunakan program *training\_process.m* yang telah dibuat untuk melatih sistem dengan citra masukan berupa positif dan negatif sempel. Dengan program *training\_process.m* seperti pada gambar berikut Gambar 3.8 berikut:

```
1 - make_roi = roi_10; 1
2
3 - citra_positif = fullfile('F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif 10'); 2
4 - addpath(citra_positif);
5 - citra_negatif = fullfile('F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\negatif 200'); 3
6
7 - trainCascadeObjectDetector('training.xml',make_roi, citra_negatif,...
8 - 'NumCascadeStages',50,'FeatureType','Haar'); 4
```

Gambar 3.8 program *training\_process.m*

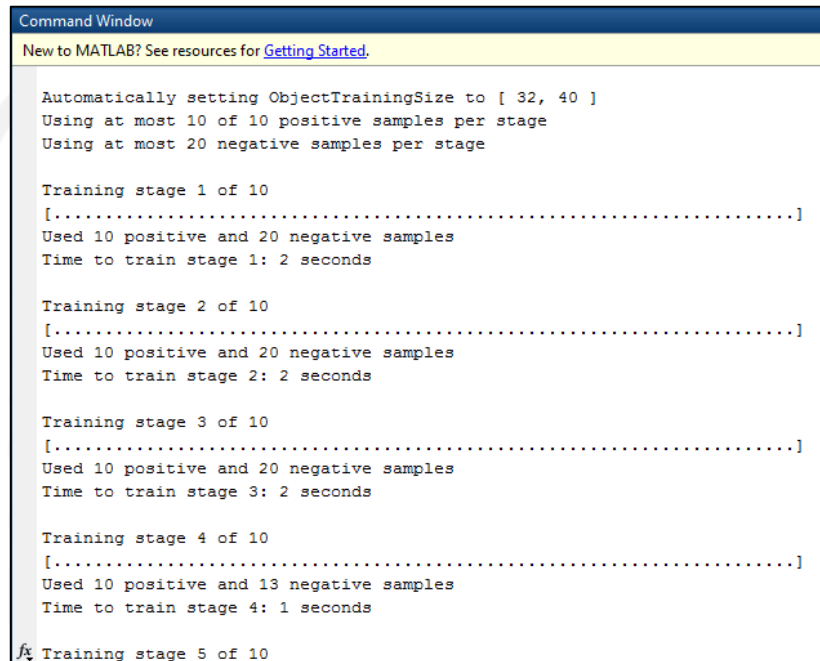
Keterangan:

1. Merupakan citra ikan yang sudah dilakukan proses *cropping*, untuk tahapan *cropping* sudah dibahas pada tahap *preprocessing*.
2. Merupakan *directory* tempat dimana penyimpanan citra positif.
3. Merupakan *directory* tempat dimana penyimpanan citra negatif.
4. *trainingCascadeObjectDetector* merupakan fungsi dari sistem deteksi, *Number Stage* atau banyaknya stage yang akan dilakukan dalam proses training untuk melatih model



citra positif dan citra negatif. *NumCascadeStage* merupakan banyaknya stage training yang akan dilakukan sistem. *FeatureType* Merupakan fitur yang digunakan untuk mengelompokkan citra dengan pengelompokan *Haar*.

Dapat dilihat pada program bahwa penulis memasukan *Number Stage* sebanyak 10, yang berarti nantinya akan ada 10 proses training stage yang akan dijalankan. Dan setelah program *training\_process.m* dijalankan maka akan tampil training stage seperti Gambar 3.9 berikut:



```
Command Window
New to MATLAB? See resources for Getting Started.

Automatically setting ObjectTrainingSize to [ 32, 40 ]
Using at most 10 of 10 positive samples per stage
Using at most 20 negative samples per stage

Training stage 1 of 10
[.....]
Used 10 positive and 20 negative samples
Time to train stage 1: 2 seconds

Training stage 2 of 10
[.....]
Used 10 positive and 20 negative samples
Time to train stage 2: 2 seconds

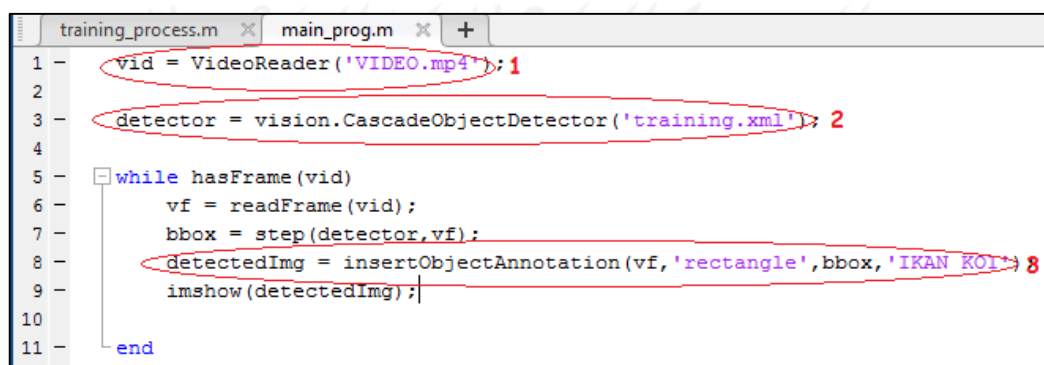
Training stage 3 of 10
[.....]
Used 10 positive and 20 negative samples
Time to train stage 3: 2 seconds

Training stage 4 of 10
[.....]
Used 10 positive and 13 negative samples
Time to train stage 4: 1 seconds

fx Training stage 5 of 10
```

Gambar 3.9 Tahap *training stage*

Setelah tahap training stage selesai, langkah selanjutnya menjalankan program utama atau main program yaitu *main\_prog.m*, dan programnya adalah seperti Gambar 3.10 berikut:



```
training_process.m x main_prog.m x +
1 - vid = VideoReader('VIDEO.mp4'); 1
2
3 - detector = vision.CascadeObjectDetector('training.xml'); 2
4
5 - while hasFrame(vid)
6 -     vf = readFrame(vid);
7 -     bbox = step(detector,vf);
8 -     detectedImg = insertObjectAnnotation(vf,'rectangle',bbox,'IKAN KOL'); 8
9 -     imshow(detectedImg);
10
11 - end
```

Gambar 3.10 Program *main\_prog.m*

Keterangan:

1. Merupakan tempat dimana *directroy* atau tempat video disimpan.

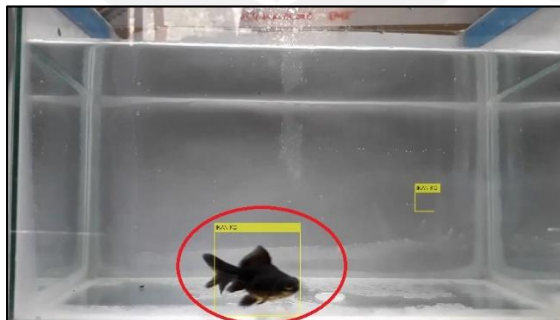
2. Merupakan hasil training yang telah dilakukan pada tahap sebelumnya, yaitu pada *training\_process.m*. File hasil training disimpan dalam file *training.xml*.
3. Merupakan proses penandaan dan labeler pada objek ikan yang nanti akan muncul pada video saat *main\_prog.m* dijalankan.

Pengujian yang dilakukan pada tahap ini adalah dengan training citra ikan positif\_10 dan positif\_30 dengan citra negatif\_10, negatif\_50, negatif\_100, negatif\_150, dan citra negatif\_200, hal ini dilakukan untuk menguji apakah sistem bekerja dengan efektif dengan jumlah citra positif dan negatif yang berbeda jumlahnya. Training yang dilakukan sebanyak lima kali training atau stage, dengan jumlah training persatgenya adalah 10, 20, 30, 40 dan 50 stage.

### 3.5 Analisis Hasil Percobaan

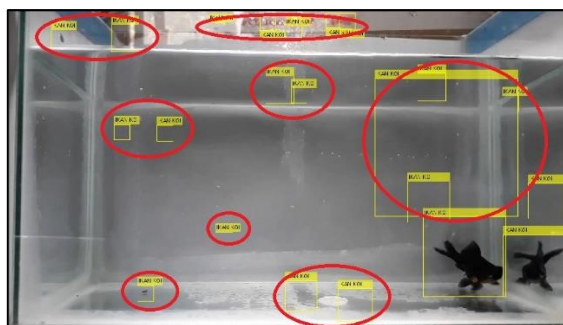
*Cascade Classifier* memerlukan dua sampel yaitu positif image dan negatif image untuk dapat memulai sistem training pada *trainCascadeObjctDetector*, kedua sampel tadi nantinya akan dilatih menjadi beberapa tahapan atau stage. Yang akan menciptakan sebuah hasil dari Sistem pendeteksi ikan koi berupa :

- Positif Benar (PB) yaitu ketika sampel positif diklasifikasi dengan benar.



Gambar 3.11 Positif Benar

- Positif Palsu (PP) yaitu ketika sampel negatif salah diklasifikasi sebagai positif.



Gambar 3.12 Positif Palsu

- Negatif Palsu (NP) yaitu ketika sempel positif keliru diklasifikasikan sebagai negatif.



Gambar 3.13 Negatif Palsu



## BAB 4

### HASIL DAN PEMBAHASAN

Pembahasan akan meliputi dua parameter yang berbeda, parameter ini dipakai untuk menguji tingkat keakuratan sistem dalam mendeteksi pergerakan ikan koki yang berada pada file video. Parameter yang dipakai adalah perbedaan jumlah gambar (citra positif) yang dimasukkan kedalam sistem untuk diolah datanya, seperti pada tahap *preprocessing* pada metodologi penelitian, yaitu 10 gambar ikan koki dan 30 gambar ikan koki. Setelah pengolahan selesai dari masing-masing pengujian yang berbeda tadi ditraining atau dilatih satu persatu dengan variasi gambar yang bukan memuat citra ikan koki (citra negatif). Yang hasil dan pembahasannya seperti pada 4.1 dan 4.2 berikut.

#### 4.1 Pengujian dengan 10 gambar positif ikan koki

Tujuan dari pengujian ini adalah untuk mengetahui kemampuan sistem dalam membaca dan mendeteksi keberadaan ikan koki dalam video. Dengan menggunakan 10 gambar ikan koki sebagai input yang telah diolah dalam program Matlab dengan beberapa tahapan, dari pemisahan atau *cropping* untuk membuat tanda ROI (*Region of Interest*) berupa tanda pada gambar yang nantinya akan dijadikan sebagai area deteksi keberadaan ikan dalam video. Hasil pengujian dengan 10 gambar positif ikan koki dapat dilihat pada Tabel 4.1.

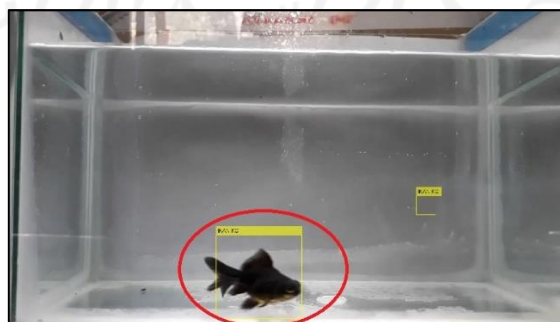
Tabel 4.1 Pengujian dengan 10 gambar positif ikan koki

stage	Citra negatif				
	10	50	100	150	200
10	PB (-)	PB (-)	PB (-)	PB (+)	PB (+)
	PP (+++)	PP (+++)	PP (+++)	PP (++)	PP (++)
	NP (-)	NP (-)	NP (-)	NP (-)	NP (-)
20	PB (-)	PB (-)	PB (-)	PB (+)	PB(+)
	PP (+++)	PP (+++)	PP (++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (+)	NP (+)
30	PB (++)	PB (++)	PB (++)	PB (+++)	PB (+++)
	PP (+++)	PP (+++)	PP (++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP+	NP+
40	PB (++)	PB (++)	PB (++)	PB (+++)	PB (+++)
	PP (+++)	PP (+++)	PP (++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (+)	NP (+)
50	PB (++)	PB (++)	PB (++)	PB (+++)	PB (+++)
	PP (+++)	PP (+++)	PP (++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (+)	NP (+)

Keterangan:

- PB (-), tidak ditemukannya objek ikan berupa sempel positif pada video.
- PB (+), ditemukannya objek ikan berupa sempel positif, akan tetapi jumlahnya kecil.
- PB (++), ditemukannya objek ikan berupa sempel positif agak besar.
- PB (+++), ditemukannya objek ikan berupa sempel positif yang besar.
- PP (+), adanya kekeliruan pembacaan sempel negatif yang dianggap positif, akan tetapi jumlahnya kecil.
- PP (++), adanya kekeliruan pembacaan sempel negatif yang dianggap positif yang agak besar.
- PP (+++), adanya kekeliruan pembacaan sempel negatif dianggap positif yang besar.
- NP (-), tidak adanya kekeliruan pembacaan sempel positif yang dianggap negatif.
- NP (+), adanya kekeliruan pembacaan sempel positif yang dianggap negatif, akan tetapi jumlahnya kecil.

Berdasarkan hasil pengujian pada Tabel 4.1 dapat dilihat bahwa Positif Benar (PB) atau sempel positif diklasifikasikan dengan benar, sudah terlihat atau muncul pada setiap pengujian. Dengan parameter tahapan pelatihan citra (*stage*) dan citra negatif yang bervariasi, mulai dari stage 10, 20, 30, 40, dan 50 dengan citra negatif berupa gambar yang bukan merupakan ikan koki dengan jumlah yang bervariasi mulai dari 10, 50, 100, 150, dan 200 gambar. Tingkat pembacaan keberadaan ikan koki semakin akurat dengan ditambahkan jumlah tahapan (*stage*) yang dimasukkan dalam proses *training*, serta penambahan citra negatif juga sangat mempengaruhi sistem dalam mendeteksi ikan koki, ditandai dengan bertambahnya jumlah positif PB (+) dalam tabel hasil pengujian. Artinya semakin banyak jumlah tahapan dan citra negatif yang diberikan untuk melatih atau mentraining sistem maka semakin akurat sistem dalam mendeteksi keberadaan ikan dalam sebuah video. Akurat dalam pengujian ini ialah hasil pengujian Positif Benar (PB) sudah muncul atau terdeteksi bahwa dalam video sudah ada ikan koki, dan Positif Benar (PB) ditandai dengan penanda persegi berwarna kuning yang akan muncul saat keberadaan ikan koki ada pada video, seperti pada Gambar 4.1.



Gambar 4.1 pengujian Positif Benar (PB)

Artinya Positif Benar (PB) adalah penanda persegi tersebut, dan jika ikan koki tidak muncul dalam video maka penanda tidak ada akan muncul. Tetapi hasil pengujian dalam Tabel 4.1 masih ada nilai PB yang minus atau PB (-), yang berarti pada video ada ikan koki namun sistem tidak mendeteksinya. Hal ini terjadi karena samanya jumlah citra positif dan citra negatif, kurangnya jumlah citra negatif, serta kurangnya jumlah tahapan atau *stage* untuk mentraining citra positif dan citra negatif yang dimasukkan dalam sistem.

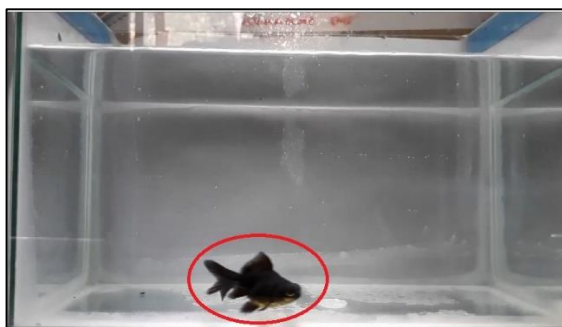
Positif Palsu (PP) atau sempel negatif salah diklasifikasikan sebagai positif, masih muncul pada setiap pengujian dengan berbagai parameter seperti pada Tabel 4.1, akan tetapi berkurang seiring dengan penambahan jumlah tahapan (*stage*) dan citra negatif. Positif Palsu (PP) dalam pengujian ini adalah ketika dalam video tidak terdapat ikan koki namun sistem mendeteksi bahwa dalam area *frame* video ada ikan koki dan cenderung muncul pada area yang bukan terdapat ikan koki, ditandai dengan tanda persegi kuning. Seperti pada Gambar 4.2 berikut.



Gambar 4.2 pengujian Positif Palsu (PP)

Hal ini terjadi karena kurangnya jumlah citra negatif yang dimasukkan dalam sistem, serta kurangnya jumlah tahapan (*stage*) yang dibutuhkan untuk mentraining citra positif dan citra negatif, akan tetapi penambahan jumlah tahapan tidak berpengaruh besar pada pengurangan Positif Palsu (PP).

Negatif Palsu (NP) atau sempel positif keliru diklasifikasikan sebagai sempel negatif, dalam pengujian ini terjadi hampir disetiap pengujian dengan citra negatif yang jumlahnya berada diatas 100 gambar. Negatif Palsu (NP) merupakan kebalikan dari Positif Benar (PB), jika Positif Benar (PB) terdapat ikan koki dalam video maka akan muncul penanda persegi kuning yang berada pada ikan koki, sebaliknya dengan Negatif Palsu (NP) jika ada ikan koi dalam video maka penanda kuning tidak muncul pada ikan koki. Hal ini terjadi karena kekeliruan sistem dalam mengkalifikasikan citra citra positif yang dianggap citra negatif, seperti pada Gambar 4.2.1 berikut.



Gambar 4.3 pengujian Negatif Palsu (NP)

#### 4.2 Pengujian dengan 30 gambar positif ikan koki

Tujuan dari pengujian ini adalah untuk melihat perbedaan hasil pengujian dengan menggunakan 10 gambar positif ikan koi dengan hasil seperti Tabel 4.1 dengan 30 gambar positif ikan koi yang hasilnya seperti pada Tabel 4.2 berikut:

Tabel 4.2 Pengujian dengan 30 gambar positif ikan koki

stage	Citra negatif				
	10	50	100	150	200
10	PB (-)	PB (+)	PB (+)	PB (+)	PB (+)
	PP (+++)	PP (+++)	PP (+++)	PP (++)	PP (++)
	NP (-)	NP (-)	NP (-)	NP (-)	NP (-)
20	PB (-)	PB (+)	PB (+)	PB (++)	PB (++)
	PP (+++)	PP (+++)	PP (+++)	PP (++)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (-)	NP (-)
30	PB (-)	PB (+)	PB (++)	PB (+++)	PB (+++)
	PP (+++)	PP (+++)	PP (++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (-)	NP (-)
40	PB (-)	PB (++)	PB (++)	PB (+++)	PB (+++)
	PP (+++)	PP (+++)	PP (+++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (-)	NP (-)
50	PB (-)	PB (++)	PB (++)	PB (+++)	PB (+++)
	PP (+++)	PP (+++)	PP (+++)	PP (+)	PP (+)
	NP (-)	NP (-)	NP (-)	NP (-)	NP (-)

Dari pengujian pada Tabel 4.2 diperoleh hasil Positif Benar (PB) yang tidak terlalu jauh dengan hasil pada pengujian dengan 10 gambar positif. Jika hasil pengujian PB dengan 10 gambar positif dengan citra negatif 10 sudah terjadi penambahan PB(+) sampai tahapan (*stage*) 50, sedangkan pada hasil pengujian PB dengan 30 gambar positif dengan cita negatif 10 masih minus (PB (-)) sampai tahapan (*stage*) 50. Jadi untuk nilai PB lebih cepat muncul dengan citra positif yang sedikit.



Positif Palsu (PP) hasil dari pengujian dengan 30 gambar positif ikan koki adalah sama dengan 10 gambar ikan koki, atau hasilnya tidak terlalu beda. Hal ini mungkin terjadi karena kurangnya jumlah citra positif yang dimasukkan. Walaupun Positif Palsu (PP) berkurang seiring dengan penambahan jumlah tahapan (*stage*) dan jumlah citra negatif.

Negatif Palsu (NP) pada Tabel 4.2 tidak terjadi sama sekali atau PB (-), berbeda dengan Tabel 4.1 yang masih terdapat PB walupun kecil. Hal ini merupakan pengaruh penambahan jumlah citra positif ikan koi.

Jadi pengujian Sistem pendeteksi ikan koki dengan 10 gambar positif ikan koki dengan 30 gambar positif ikan koki diperoleh hasil terbaik yaitu dapat membaca pergerakan ikan koki dalam sebuah video, walaupun masih ada Positif Palsu (PP) yang muncul saat sistem dijalankan, akan tetapi berkurang seiring penambahan jumlah tahapan (*stage*) dan penambahan citra negatif. Dan untuk Positif Benar (PB) dan Negatif Palsu (NP) hasilnya tidak berbeda baik dengan citra 10 gambar positif dan 30 gambar positif ikan koki.

Karena sistem yang dibuat menggunakan *Cascade Classifier* dibutuhkan sekumpulan sampel positif dan negatif, sampel positif merupakan objek atau gambar yang nantinya akan dicari dalam video, yang pada tahap *preprocessing* sudah dimasukkan pada *Image Labeler* untuk memberi label pada objek gambar positif dengan kotak pembatas berupa *Region Of Interest (ROI)*. Dan sampel negatif merupakan gambar objek yang tidak akan dideteksi yang berguna sebagai pembanding dengan sampel positif, yang akan ditraining atau dilatih dengan fungsi *trainCascadeObjectDetector*.



## **BAB 5**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Dari hasil penelitian yang sudah dilakukan oleh penulis disimpulkan bahwa:

1. Sistem dapat mendeteksi keberadaan ikan koki dalam video dengan semakin akurat, dengan penambahan jumlah tahapan (*stage*) dan jumlah citra negatif.
2. Pengujian dengan 10 gambar positif ikan koki dengan 30 gambar ikan koki menghasilkan nilai Positif Benar (PB), Positif Palsu (PP), dan Negatif Palsu (NP) yang hasilnya tidak terlalu beda.
3. Positif Palsu (PP) masih terjadi disetiap pengujian, akan tetapi berkurang dengan penambahan jumlah tahapan (*stage*) dan jumlah citra negatif.
4. Penambahan citra positif ikan koi menjadi 30 mempengaruhi nilai Negatif Palsu (NP) yaitu mengurangi tingkat kehilangan objek berupa ikan koi saat pendeteksian keberadaan ikan dalam video.
5. Tingkat keberhasilan sistem mendeteksi ikan koki dalam video berdasarkan hasil pengujian dengan *datasheet* berupa citra positif ikan koki sebanyak 10 dan 30 gambar, dan citra negatif dengan jumlah yang bervariasi dari 10, 50, 100, 150, dan 200 dengan berbagai jumlah tahapan training (*stage*) 10, 20, 30, 40, 50. Menghasilkan nilai berupa Positif Benar (PB), Positif Palsu (PP), dan Negatif Palsu (NP) dengan keberhasilan sebesar 85% berdasarkan hasil pengujian.
6. Tingkat keberhasilan yang tinggi didapat pada tahapan (*stage*) dengan jumlah 50 dan citra negatif dengan jumlah 200.

#### **5.2 Saran**

1. Pada penelitian selanjutnya atau pengembangan dari sistem ini diharapkan dapat mendeteksi ikan secara *real time*.
2. Dapat mendeteksi berbagai jenis ikan dalam suatu lingkungan
3. Saat pengambilan citra positif dan negatif diharapkan mengambil kualitas yang baik dan jelas, karena hal ini dapat mempengaruhi hasil dari pendeteksian.
4. Saat pengambilan video diharapkan kamera stabil.

## DAFTAR PUSTAKA

- [1] KOMPAS, "Ikan Hias Air Laut Masih Sulit," *Kompas*. [Online]. Available: <https://regional.kompas.com/read/2010/08/21/08461858/ikan.hias.air.laut.masih.sulit>.
- [2] M. Rinaldi, *Pengolahan Citra Digital dengan pendekatan Algoritmik*. Bandung: Informatika, 2004.
- [3] M. M. Saberioon and P. Cisar, "Automated multiple fish tracking in three-Dimension using a Structured Light Sensor," *Comput. Electron. Agric.*, vol. 121, pp. 215–221, Feb. 2016.
- [4] B. Bendale and A. Karwankar, "Moving Object Tracking in Video Using MATLAB," *Ijecsce.Org*, vol. 2, no. 1, pp. 5–9, 2012.
- [5] M. F. Bulbul, F. Badsha, and R. Islam, "Object Detection by Point Feature Matching using Matlab," *Adv. Image Video Process.*, vol. 5, no. 6, 2018.
- [6] M. R. Kumaseh, L. Latumakulita, and N. Nainggolan, "SEGMENTASI CITRA DIGITAL IKAN MENGGUNAKAN METODE THRESHOLDING," *J. Ilm. SAINS*, vol. 13, no. 1, pp. 74–79, May 2013.
- [7] D. Bee, W. Weku, and A. Rindengan, "Aplikasi Penentuan Tingkat Kesegaran Ikan Selar Berbasis Citra Digital Dengan Metode Kuadrat Terkecil," *Cartes.*, vol. 5, no. 2, pp. 121–130, 2016.
- [8] I. Indrabayu, M. Niswar, and A. A. Aman, "Sistem Pendeteksi Kesegaran Ikan Bandeng Menggunakan Citra," *J. INFOTEL - Inform. Telekomun. Elektron.*, vol. 8, no. 2, pp. 170–179, Nov. 2016.
- [9] Mathworks, "Image Processing Toolbox." [Online]. Available: <https://www.mathworks.com/products/image.html>. [Accessed: 02-Feb-2018].
- [10] S. Madenda, *Pengolahan Citra & Video Digital*. Jakarta: Erlangga, 2015.
- [11] Mathworks, "Train a Cascade Object Detector." [Online]. Available: <https://www.mathworks.com/help/vision/ug/train-a-cascade-object-detector.html>.
- [12] Mathworks, "Train a Cascade Object Detector - MATLAB & Simulink - MathWorks." [Online]. Available: <https://www.mathworks.com/help/vision/ug/train-a-cascade-object-detector.html>. [Accessed: 02-Feb-2018].
- [13] N. N. Putri, "Aplikasi Pendeteksi Objek bergerak pada Image Sequence Dengan Metode Background Substraction," *J. Teknol. Rekayasa Vol.*, vol. 21, no. 3, 2016.

- [14] S. Abidin, “Deteksi Wajah Menggunakan Metode Haar Cascade Classifier Berbasis Webcam Pada Matlab,” *J. Teknol. Elekterika*, vol. 15, no. 1, 2018.
- [15] A. M. Asif Anjum Akash , Abdus Salim Mollah, “Improvement of Haar Feature Based Face Detection in OpenCV Incorporating Human Skin Color Characteristic,” 2016. [Online]. Available: <http://www.symbiosisonlinepublishing.com/computer-science-technology/images/computer-science-information-technology-06-g001.gif>.



## LAMPIRAN

Program sistem pendeteksi ikan koki.

Program proses training sampel positif dan negatif :

```
make_roi = roi_10;

citra_positif = fullfile('F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\positif_10');
addpath(citra_positif);
citra_negatif = fullfile('F:\MAHBUB ESSAY\PROGRAM SISTEM PENDETEKSI IKAN KOI\negatif_200');

trainCascadeObjectDetector('training.xml',make_roi, citra_negatif, ...|
    'NumCascadeStages',50,'FeatureType','Haar');
```

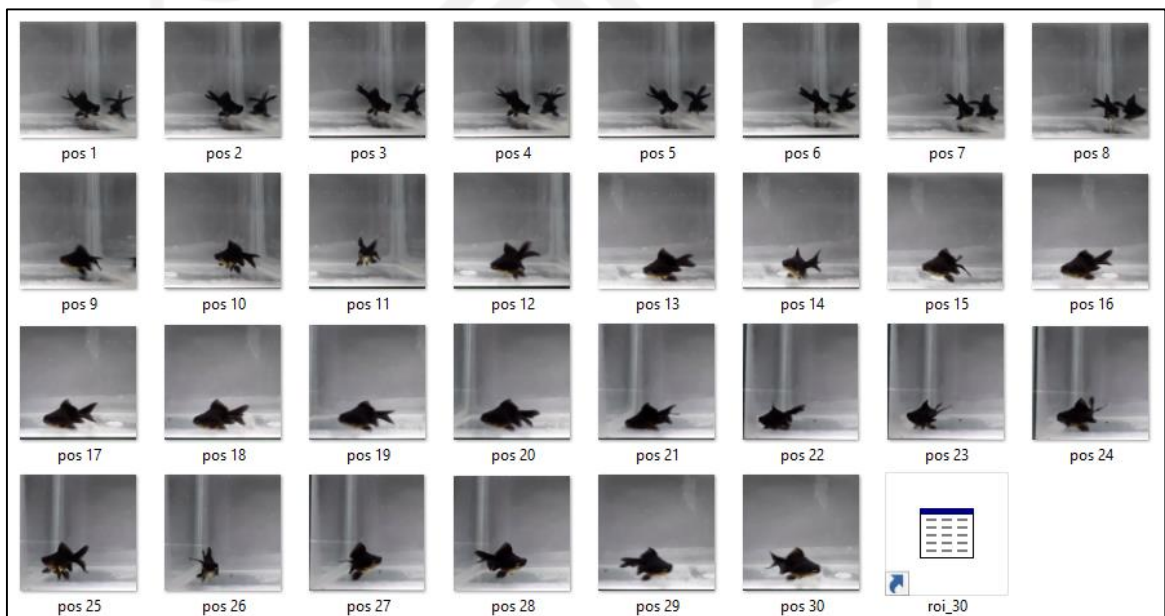
Program utama deteksi ikan dalam video :

```
vid = VideoReader('VIDEO.mp4');

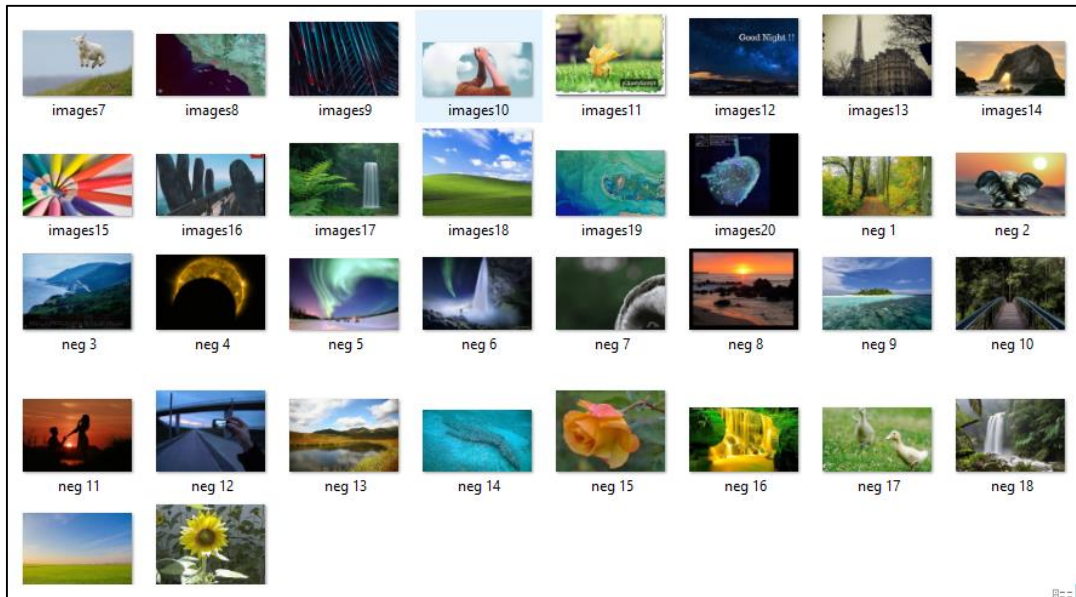
detector = vision.CascadeObjectDetector('training.xml');

while hasFrame(vid)
    vf = readFrame(vid);
    bbox = step(detector,vf);
    detectedImg = insertObjectAnnotation(vf,'rectangle',bbox,'IKAN KOI');
    imshow(detectedImg);
end
```

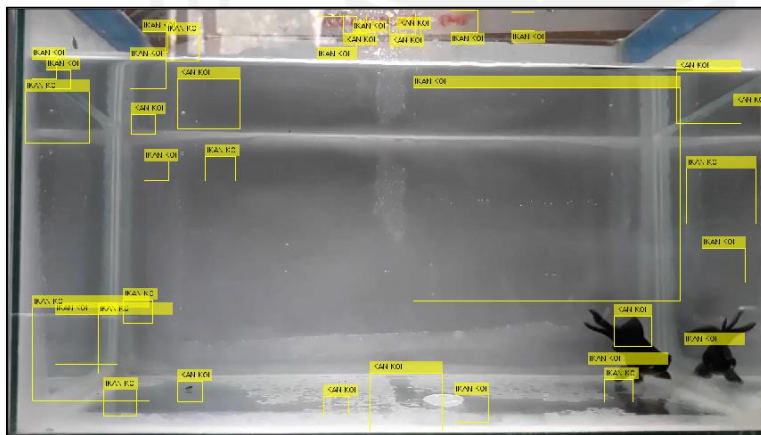
Sempel positif :



Sempel negatif :



Hasil deteksi sistem dari tingkat keakuratan yang kecil :



Hasil deteksi sistem dengan tingkat keakuratan yang tinggi :

