

## BAB III

### PELAKSANAAN KERJA MAGANG

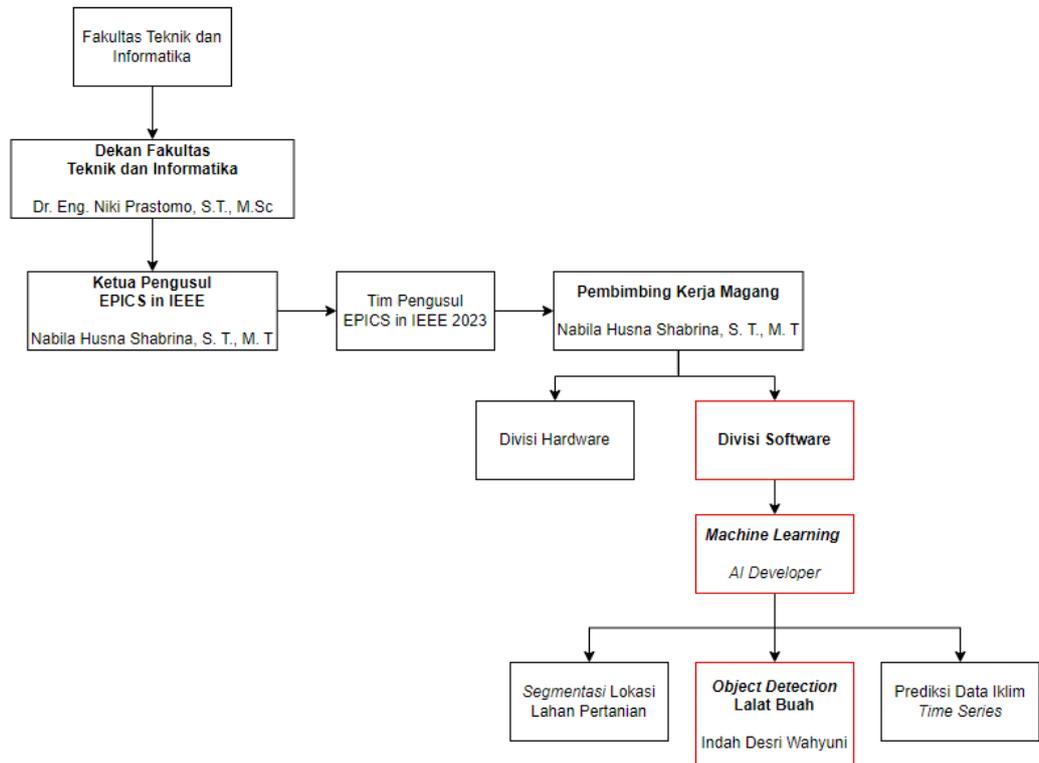
#### 3.1 Kedudukan dan Koordinasi

Dalam pelaksanaan program magang ini, penulis tergabung dalam proyek *EPICS in IEEE* yang berjudul *Sistem Pertanian Pintar dengan Integrasi Artificial Intelligence dan Internet of Things untuk Peramalan dan Pengelolaan Hama*. Proyek ini merupakan hasil kolaborasi antara Universitas Multimedia Nusantara (UMN) dan Universitas Gadjah Mada (UGM), yang memperoleh pendanaan hibah dari *EPICS in IEEE* untuk mengembangkan sistem pertanian cerdas berbasis teknologi AIoT.

Dalam proyek ini, terdapat dua divisi utama, yaitu *Hardware* dan *Software*. Penulis tergabung dalam divisi *Software*, khususnya di bagian *Machine Learning*, yang berfokus pada pengembangan sistem kecerdasan buatan (*Artificial Intelligence*). Divisi ini memiliki tiga peran utama dalam pengembangan AI, yaitu:

1. Deteksi dan perhitungan jumlah lalat buah.
2. Segmentasi lokasi lahan pertanian.
3. Peramalan data sensor.

Penulis berperan sebagai *AI Developer* pada bagian deteksi objek lalat buah, di bawah bimbingan Ibu Nabila Husna Shabrina. Selain itu, dalam pelaksanaan proyek ini, divisi *Software* juga berkolaborasi dengan divisi *Hardware*, terutama dalam pengambilan data primer dari sensor untuk peramalan secara *real-time*. Secara lebih rinci, alur kerja, tanggung jawab, serta posisi penulis sebagai *Machine Learning Developer* dalam proyek ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Struktur Spesifik divisi *Machine Learning Developer*

## 3.2 Tugas dan Uraian Kerja Magang

### 3.2.1. Tugas Kerja Magang

Pelaksanaan kerja magang project EPICS in IEEE Sistem Pertanian Pintar dengan Integrasi *Artificial Intelligence* dan *Internet of Things* untuk Peramalan dan Pengelolaan Hama, penulis ditugaskan untuk dapat merancang sistem *Machine Learning* khususnya pada bagian pengumpulan dan pemrosesan *dataset* lalat buah untuk pengembangan menu perhitungan otomatis di aplikasi *MySalak*.

Dalam rangka pengumpulan dan pemrosesan *dataset* untuk pengembangan fitur pada aplikasi ini, pada minggu pertama dilakukan diskusi, pencarian referensi, serta penyusunan rencana kerja yang akan dijalankan. Selain itu, penulis juga melakukan pendekatan awal dengan perwakilan kelompok tani untuk menjelaskan tujuan

proyek serta langkah-langkah yang akan dilakukan ke depannya. Pada minggu kedua, penulis menyusun rencana program kerja yang lebih terstruktur untuk pengambilan *dataset* lalat buah di lahan pertanian salak, sehingga proses pengumpulan data dapat dilakukan secara sistematis dan efisien.

Pada minggu ketiga, penulis bersama tim melakukan kunjungan kerja lapangan ke Sleman, Yogyakarta, untuk mengumpulkan *dataset* lalat buah (internal), berdiskusi dengan pihak Universitas Gadjah Mada (UGM), serta melakukan survei awal kepada petani guna mengevaluasi perkembangan aplikasi. Pada minggu keempat, melakukan riset mengenai platform yang paling sesuai untuk anotasi gambar serta mengoordinasikan proses anotasi dengan tim guna memastikan sistematika yang jelas dan efisien.

Mulai minggu kelima hingga tahap akhir, kegiatan yang dilakukan mencakup anotasi gambar untuk *dataset* internal, pengumpulan *dataset* eksternal, validasi oleh ahli pertanian, pemrosesan *dataset*.

### 3.2.2. Uraian Kerja Magang

Tabel 3. 1 *Linimasa* Kerja Magang

| Minggu | Rincian Tugas yang Dilakukan   |
|--------|--|
| 1      | <ol style="list-style-type: none"> <li>1. Rapat dan diskusi bersama dengan koordinator <i>project</i> (UMN) terkait <i>project</i> yang akan dilaksanakan.</li> <li>2. <i>Research</i> terkait cara pengumpulan dan pengambilan <i>dataset</i>, platform yang digunakan untuk anotasi gambar, dan langkah <i>pre-processing</i> yang dilakukan.</li> <li>3. Menghubungi perwakilan kelompok tani untuk menjelaskan <i>project</i> dan langkah kedepannya.</li> </ol> |

|     |  |
|-----|--|
| 2   | Penyusunan rencana program kerja yang lebih terstruktur untuk pengambilan <i>dataset</i> lalat buah di lahan pertanian salak, sehingga proses pengumpulan data dapat dilakukan secara sistematis dan efisien, seperti susunan kegiatan, lokasi, dan target selama 4 hari.  |
| 3   | <ol style="list-style-type: none"> <li>1. Kunjungan kerja lapangan ke Sleman, Yogyakarta untuk pengumpulan <i>dataset</i> internal lalat buah.</li> <li>2. Rapat dan diskusi bersama dengan pihak Universitas Gadjah Mada (UGM) mengenai <i>progress project</i> yang telah dilaksanakan.</li> <li>3. Melakukan survei awal kepada petani guna mengevaluasi perkembangan aplikasi</li> </ol> |
| 4   | <ol style="list-style-type: none"> <li>1. Riset mengenai platform untuk anotasi gambar yang digunakan.</li> <li>2. Mengumpulkan gambar lalat buah dari hasil tangkapan tim lainnya.</li> <li>3. Koordinasi proses anotasi dengan tim guna memastikan sistematika yang jelas dan efisien dengan membuat 10 <i>workspace</i> yang berbeda pada platform yang digunakan.</li> </ol>             |
| 5   | Ubah file yang memiliki format .HEIC ke .JPG agar gambar, karena platform yang digunakan hanya menerima gambar dengan format .JPG, .PNG  |
| 6-9 | <ol style="list-style-type: none"> <li>1. Anotasi gambar untuk <i>dataset</i> internal</li> <li>2. Menghubungi perwakilan kelompok tani untuk laporan mengenai jumlah tangkapan lalat buah yang disertai dengan gambar, sehingga digunakan untuk <i>dataset</i> eksternal.</li> </ol>  |
| 10  | <ol style="list-style-type: none"> <li>1. Anotasi gambar untuk <i>dataset</i> eksternal</li> </ol>   |

|       |   |
|-------|---|
|       | <ol style="list-style-type: none"> <li>2. Diskusi dengan <i>expert</i> pertanian untuk melakukan validasi mengenai ketepatan pada label/anotasi lalat buah.</li> <li>3. Melakukan <i>pre-processing dataset</i> sebagai tahap persiapan sebelum pelatihan model dalam sistem perhitungan otomatis, guna memastikan data yang digunakan memiliki kualitas optimal dan sesuai untuk pengolahan lebih lanjut.</li> </ol> |
| 11-13 | <ol style="list-style-type: none"> <li>1. Merapikan <i>dataset</i> lalat buah pada <i>roboflow</i> di setiap <i>workspaceny</i></li> <li>2. Berdiskusi dengan <i>expertise</i> pertanian untuk melakukan validasi pada <i>dataset</i> internal maupun eksternal</li> </ol>  |

### 3.2.3. Pengumpulan *Dataset* Lalat Buah

*Dataset* yang dikumpulkan dalam *project* ini ada 2 yaitu *dataset* internal dan *dataset* eksternal. Perbedaan utama antara keduanya terletak pada metode pengumpulannya, di mana *dataset* internal dikumpulkan secara langsung di lahan pertanian, sedangkan *dataset* eksternal berasal dari perantara kelompok tani.

- *Dataset* Internal

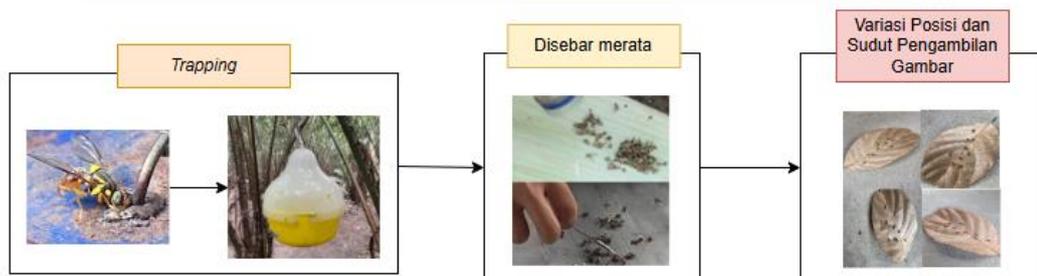
*Dataset* internal lalat buah dikumpulkan secara langsung di lahan pertanian kelompok tani Mitra Turindo selama satu minggu, tepatnya di Kelompok Tani Kusuma Mulya, Sido Makmur, Si Cantik, dan Sedyo Makmur, dengan koordinat yang ditunjukkan pada Tabel 3.2. Dalam tabel tersebut, hanya tercantum empat kelompok tani yang terlibat dalam proses pengumpulan *dataset* internal. Hal ini disebabkan oleh keterbatasan ketersediaan serta kesediaan kelompok tani tersebut untuk memberikan sampel lalat buah yang terdapat di lahan pertanian salak mereka. Proses pengumpulan ini menghasilkan total 6.008 gambar, yang kemudian digunakan sebagai data utama dalam tahap pelatihan model untuk sistem perhitungan otomatis.

Tabel 3. 2 Koordinat Lokasi Pengumpulan *Dataset*

| Kelompok Tani | Koordinat Lokasi                          |
|---------------|---|
| Kusuma Mulya  | -7.6201011106408,<br>110.40155469528769   |
| Sido Makmur   | -7.626041647523953,<br>110.38289835111135 |
| Si Cantik     | -7.6421667, 110.3593611                   |
| Sedyo Makmur  | -7.609593, 110.377292                     |

- *Dataset* Eksternal

*Dataset* eksternal dikumpulkan dari foto laporan tangkapan petani yang dikirim setiap minggu melalui *WhatsApp*, dengan total 224 gambar. *Dataset* ini digunakan dalam proses pengujian (*testing*) model untuk mengevaluasi kinerjanya dalam kondisi nyata di lapangan.

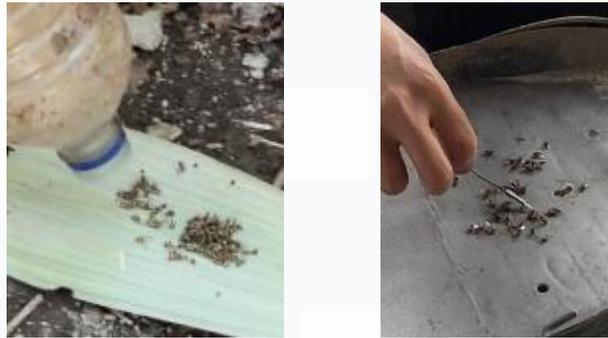


Gambar 3. 2 *Flowchart* Pengumpulan *Dataset*

*Dataset* gambar lalat buah dikumpulkan dari hasil tangkapan lalat buah di perangkap (*trapping*) yang mengandung zat pemikat *Methyl Eugenol*, seperti yang ditunjukkan pada Gambar 3.2. Perangkap ini dipasang oleh petani di sekitar lahan perkebunan salak sebagai bagian dari upaya pemantauan populasi hama. Secara umum, setiap kelompok tani memasang sekitar 10 perangkap yang tersebar di

berbagai titik di lahan, guna memastikan cakupan pengambilan sampel yang lebih representatif.

Langkah pengambilan foto untuk *dataset* dilakukan dengan cara menyebar dan menata lalat buah yang tertangkap di perangkap secara berjarak di atas media latar belakang, seperti yang ditunjukkan pada Gambar 3.3. Pada Gambar 3.3 bagian (a), lalat yang tertangkap di perangkap pertama-tama dituangkan. Kemudian disebar atau ditata dengan jarak tertentu, ditunjukkan pada Gambar 3.3 bagian (b). Media yang digunakan sebagai latar belakang bervariasi, termasuk ubin, daun, kayu, dan lainnya, sebagaimana diperlihatkan pada Gambar 3.4. Pada Gambar 3.4, terlihat berbagai jenis media yang digunakan sebagai latar belakang untuk pengambilan foto lalat buah. Bagian (a) menunjukkan penggunaan daun, (b) ubin, dan (c) kayu, sementara bagian (d) menampilkan ubin dalam kondisi basah. Keberagaman media ini bertujuan untuk meningkatkan variasi *dataset* agar model yang dikembangkan lebih adaptif terhadap berbagai kondisi latar belakang. Media yang dipilih tidak hanya diperhatikan agar objek lalat buah terlihat jelas dalam gambar, tetapi juga dipilih berdasarkan kemudahan akses dan ketersediaannya di lokasi pengumpulan data. Selain itu, proses pengumpulan *dataset* juga memperhitungkan berbagai kondisi lalat buah, termasuk lalat dalam keadaan kering, terlihat pada Gambar 3.4 bagian (a), (b), dan (c), serta lalat dalam kondisi basah yang ditunjukkan pada bagian (d). Variasi ini bertujuan untuk meningkatkan ketahanan model dalam mengenali lalat buah di berbagai kondisi lingkungan nyata.



(a)

(b)

Gambar 3. 3 Proses Pengambilan Dataset: (a) Dituang dari Perangkap dan (b) Disebar Merata



(a)

(b)

(c)

(d)

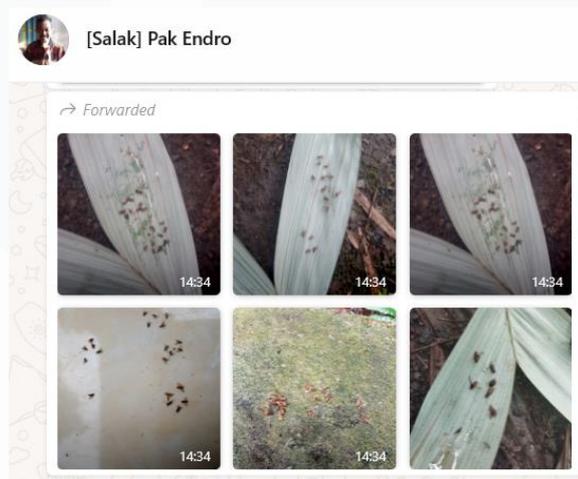
Gambar 3. 4 Variasi *Dataset* Lalat Buah: (a) Variasi Media Daun, (b) Media Ubin, (c) Media Kayu, dan (d) Media Ubin dengan Kondisi Basah

Kemudian, variasi posisi dan sudut pengambilan gambar diterapkan selama proses dokumentasi lalat buah guna memperkaya keberagaman *dataset*, seperti yang ditunjukkan pada Gambar 3.5. Hal ini dilakukan agar model yang dikembangkan mampu mengenali lalat buah dalam berbagai perspektif dan kondisi pencahayaan. Dengan variasi ini, *dataset* menjadi lebih representatif terhadap situasi nyata di lapangan, sehingga model dapat lebih akurat dalam mendeteksi dan menghitung jumlah lalat buah pada berbagai latar belakang dan sudut pandang.



Gambar 3. 5 Variasi Pengambilan Dataset

Pengumpulan *dataset* eksternal dilakukan melalui *WhatsApp*, di mana perwakilan kelompok tani mengirimkan laporan setiap minggu. Laporan ini berisi jumlah lalat buah yang tertangkap serta dilengkapi dengan gambar dokumentasi, ditunjukkan pada Gambar 3.6. Setiap kelompok tani memiliki jadwal pelaporan yang berbeda dalam rentang waktu satu minggu, sehingga gambar yang diterima berasal dari berbagai hari dan kondisi yang beragam. Data gambar yang diperoleh dari laporan ini kemudian digunakan sebagai *dataset* untuk menguji (*testing*) model, guna mengevaluasi performanya dalam kondisi nyata di lapangan.



Gambar 3. 6 Proses Pengumpulan *Dataset* Eksternal

#### 3.2.4. *Pre-Processing Dataset*

- **Menyeragamkan Format ke .JPG**

Dalam proses pengumpulan *dataset*, berbagai jenis perangkat dan kamera digunakan, ditunjukkan pada Gambar 3.7. Penggunaan perangkat yang beragam bertujuan untuk menghasilkan *dataset* dengan variasi dalam aspek kontras, pencahayaan, serta karakteristik visual lainnya. Dengan adanya perbedaan kualitas dan spesifikasi kamera, *dataset* yang dikumpulkan menjadi lebih beragam, sehingga model yang dikembangkan dapat lebih adaptif dan mampu mengenali lalat buah dalam berbagai kondisi pencahayaan dan kualitas gambar di lapangan.



Gambar 3. 7 Keberagam Kamera yang Digunakan

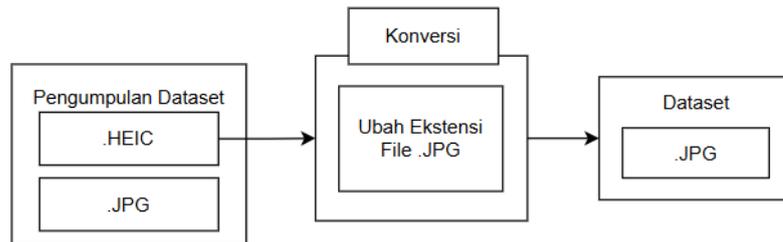
Berikut jenis kamera dan spesifikasinya yang digunakan dalam proses pengambilan *dataset*:

Tabel 3. 3 Spesifikasi Perangkat

| Perangkat         | Resolusi Gambar          | Format |
|-------------------|--------------------------|--------|
| Iphone 12 Mini    | 3024 x 4032              | HEIC   |
| Iphone 12 Pro     | 3024 x 4032              | HEIC   |
| Oppo reno7 z 5g   | 3468 x 4624              | JPG    |
| Redmi K20         | 2448 x 3264              | JPG    |
| Xiaomi 11T        | 2233 x 3968              | JPG    |
| Samsung S23 fe    | 3000x4000                | JPG    |
| Sony Cybershot 16 | 1920 x 1080<br>640 x 480 | JPG    |

Dapat dilihat pada Tabel 3.3 bahwa terdapat beberapa gambar dalam *dataset* yang memiliki format file HEIC. Diperoleh 159 gambar dengan format .HEIC. Karena format HEIC tidak didukung oleh platform anotasi *Roboflow*, gambar dikonversi ke format JPG. Konversi ini diperlukan karena *Roboflow* hanya mendukung gambar dengan format JPG atau PNG. Untuk memastikan proses berjalan lebih cepat dan efisien, konversi dilakukan menggunakan bahasa pemrograman *Python*,

ditunjukkan Gambar 3.8. Pada Gambar 3.8 dapat dilihat *flow general* untuk melakukan konversi.



Gambar 3. 8 *Flow* Diagram Konversi

Langkah pertama adalah membaca *file* HEIC untuk membuat objek gambar dalam format lain. Seperti yang ditunjukkan pada Gambar 3.9, *file* HEIC dibaca menggunakan pustaka *pyheif*, yang kemudian menghasilkan objek gambar dari data mentah. Objek ini dapat dimanipulasi atau disimpan ke dalam format yang diinginkan.

```
image = Image.frombytes(  
    heif_file.mode,  
    heif_file.size,  
    heif_file.data,  
    "raw",  
    heif_file.mode,  
    heif_file.stride,  
)
```

Gambar 3. 9 Potongan kode untuk Membuat Objek Gambar dari Data HEIC

Selanjutnya, seperti yang diperlihatkan pada Gambar 3.10, nama *file output* dibuat dengan mengganti ekstensi *file* dari *.heic* menjadi *.jpg*. Proses ini memastikan *file* keluaran memiliki format yang sesuai dengan kebutuhan platform anotasi.

```
base_name = os.path.splitext(os.path.basename(heic_path))[0] + ".jpg"
output_path = os.path.join(output_folder, base_name)
```

Gambar 3. 10 Potongan kode untuk Menentukan Nama *File Output* JPG

Terakhir, seperti yang ditunjukkan pada Gambar 3.11, objek gambar yang telah dimuat disimpan dalam format JPG menggunakan metode *save* dari pustaka *Pillow*. Metode ini memungkinkan gambar disimpan di folder *output* dengan nama dan format yang sesuai.

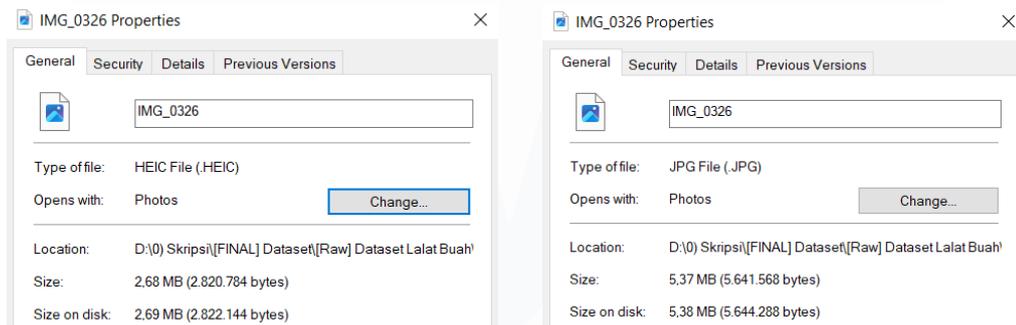
```
image.save(output_path, "JPEG")
print(f"Konversi selesai: {output_path}")
```

Gambar 3. 11 Potongan kode untuk Menyimpan Gambar Format JPG

Dengan langkah-langkah ini, semua gambar dalam *dataset* berhasil dikonversi ke format JPG dapat dilihat pada Gambar 3.12, sehingga dapat diunggah dan diproses pada platform anotasi *Roboflow*.

Sebelum

Sesudah





```
➔ Konversi selesai: /content/output/IMG_0384.jpg
Konversi selesai: /content/output/IMG_0382.jpg
Konversi selesai: /content/output/IMG_0358.jpg
Konversi selesai: /content/output/IMG_0338.jpg
Konversi selesai: /content/output/IMG_0335.jpg
Konversi selesai: /content/output/IMG_0385.jpg
Konversi selesai: /content/output/IMG_0333.jpg
Konversi selesai: /content/output/IMG_0386.jpg
Konversi selesai: /content/output/IMG_0387.jpg
Konversi selesai: /content/output/IMG_0368.jpg
```

Gambar 3.12 Hasil *Konvert Heic ke JPG*

- ***Resize***

Ukuran gambar yang berbeda-beda dari masing-masing *dataset* karena diambil dengan menggunakan berbagai jenis kamera yang resolusi beragam. Agar pada saat proses pelatihan dapat bekerja secara maksimal terutama untuk model YOLO (*You Only Look Once*), gambar dilakukan *resize* menjadi ukuran 448x448 piksel. Ukuran ini, mampu membuat model dapat memproses gambar dengan lebih ringan dan cepat, sehingga dapat mempercepat proses pelatihan tanpa ada membebani sumber daya komputasi secara berlebihan [6]. *Resize* gambar menggunakan bahasa pemrograman *Python*, seperti yang ditunjukkan pada Gambar 3.13. Hal ini karena fleksibilitas *Python* dalam mengolah data dalam skala besar. Dengan bantuan *library OpenCV*, proses *resize* dapat dilakukan secara otomatis untuk semua gambar dalam *dataset*, sebagaimana yang ditunjukkan pada Gambar 3.14. Hal ini sangat efisien mengingat jumlah gambar dalam *dataset* cukup banyak, sehingga sulit dilakukan secara manual.

```

def resize_images_in_dataset(dataset_path, new_size=(448, 448)):
    image_folder = os.path.join(dataset_path, 'images')
    #...baca file
    resized_image = cv2.resize(image, new_size, interpolation=cv2.INTER_LINEAR)
    cv2.imwrite(image_path, resized_image)

    print(f"Ubah ukuran gambar: {image_file} -> {new_size}")

dataset_path = '/content/lalat'

resize_images_in_dataset(dataset_path, new_size=(448, 448))

```

Gambar 3. 13 Potongan Kode *Resize* Gambar

20240808\_143911\_jpg.rf.7561123d4feff797ee294f5931662d96.jpg  
 Ukuran: 448 x 448 pixels



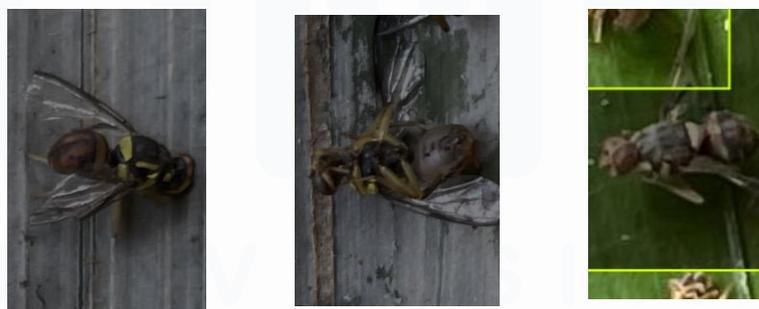
Gambar 3. 14 Output *Resize* Gambar

### 3.2.5. Anotasi Gambar

Proses anotasi gambar adalah tahap penting dalam pemrosesan *dataset*, terutama dalam pengembangan model deteksi berbasis *machine learning*. Proses anotasi melibatkan penandaan objek dengan menggunakan kotak pembatas (*bounding boxes*), yaitu garis persegi yang mengelilingi objek target dalam gambar. Kotak ini membantu model dalam memahami posisi, ukuran, serta bentuk objek yang akan dideteksi. Dalam penelitian ini, proses anotasi terdapat 2 label yaitu “lalat” dan “0”. *framework* yang digunakan yaitu *Roboflow*. *Roboflow* merupakan platform yang dirancang khusus untuk pengelolaan *dataset* dan pelabelan gambar, yang memungkinkan anotasi dilakukan dengan lebih efisien dan terorganisir [7]. Kualitas

anotasi sangat berpengaruh terhadap performa model, karena anotasi yang akurat memungkinkan model membedakan antara objek utama (lalat buah) dan elemen latar belakang atau objek lain yang tidak relevan.

Dalam proses anotasi dataset lalat buah, terdapat aturan yang harus diikuti. Aturan ini ditetapkan berdasarkan arahan dari koordinator proyek EPICS in IEEE Universitas Gadjah Mada (UGM), yaitu Dr. Suputa. Dapat dilihat pada Gambar 3.15 menunjukkan label “lalat” dengan kondisi utuh sehingga dapat diidentifikasi sebagai individu lalat. Namun, tidak hanya lalat buah yang utuh yang diidentifikasi sebagai objek, tetapi bagian toraks lalat buah juga dianggap relevan, ditunjukkan pada Gambar 3.16 bagian (✓). Namun, jika hanya bagian kepala atau abdomen (perut) yang terlihat dalam gambar, bagian tersebut tidak akan diklasifikasikan sebagai lalat buah. Dikarenakan bagian tubuh tersebut mungkin terbawa oleh lalat buah lain, sehingga tidak dapat diidentifikasi sebagai individu lalat buah yang terpisah, ditunjukkan pada Gambar 3.8 bagian (✗). Pendekatan ini diterapkan untuk meningkatkan akurasi model dalam mendeteksi dan menghitung jumlah lalat buah dengan lebih tepat, serta meminimalkan kesalahan akibat objek yang tidak utuh. Meskipun kasus ini jarang terjadi, penerapannya tidak memberikan dampak signifikan terhadap performa model.



Gambar 3. 15 Label “Lalat”



Gambar 3. 16 Bagian Toraks Lalat Buah

Kemudian, terdapat label "0". Label ini digunakan dalam kondisi lalat buah berkumpul dalam jumlah besar, sehingga tidak memungkinkan untuk mengidentifikasi setiap individu secara terpisah. Contoh penerapan label "0" dapat dilihat pada Gambar 3.17, yang memperlihatkan kondisi di mana pengelompokan lalat membuat pendeteksian individu menjadi sulit.

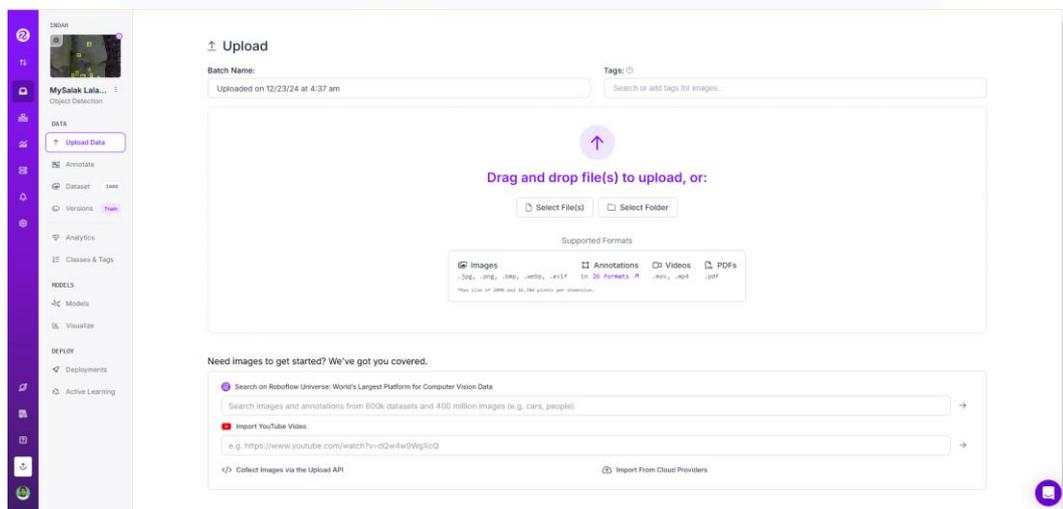


Gambar 3. 17 Label "0"

Proses anotasi *dataset* dilakukan menggunakan platform *Roboflow* karena memiliki antarmuka yang intuitif dan mudah digunakan, sehingga dapat mempercepat proses anotasi. *Dataset* yang dianotasi terdiri dari *dataset* internal sebanyak 6.008 foto dan *dataset* eksternal sebanyak 224 foto. Sebelum memulai proses anotasi, dibuat 10

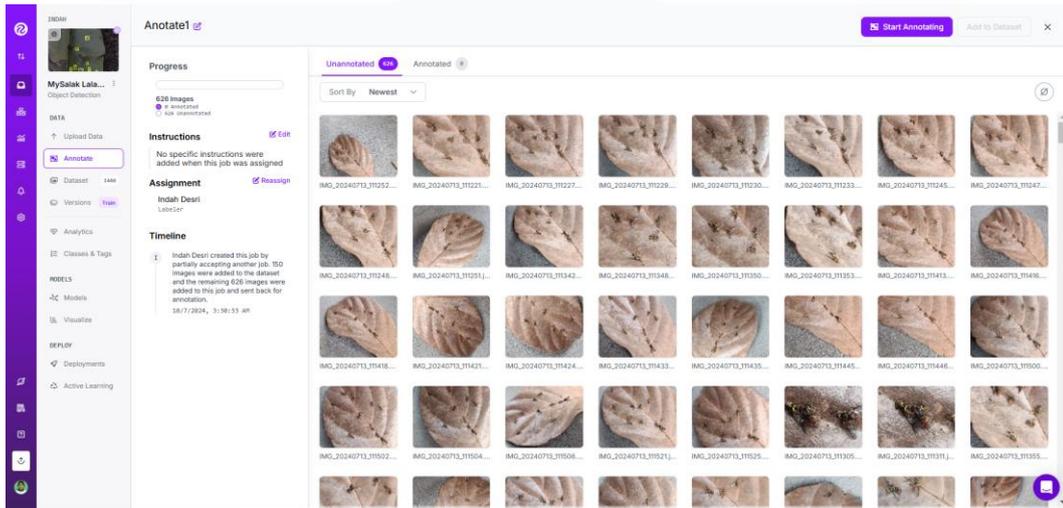
*workspace* untuk membagi tugas di antara anggota tim. Tim terdiri dari 20 orang, di mana setiap *workspace* dikerjakan oleh 1–2 orang. Pembagian ini dilakukan karena adanya batasan pada *Roboflow*, yang hanya mengizinkan maksimal 3 orang untuk mengakses satu *workspace* secara bersamaan. Selain mengatasi batasan teknis, pembagian *workspace* ini juga bertujuan untuk memudahkan koordinasi dalam proses anotasi.

Pertama, unggah *dataset* dengan “*drag and drop*” atau klik “*select file*” atau “*select folder*” seperti pada tampilan Gambar 3.18. *File* gambar yang bisa diunggah hanya yang berformat *jpg* dan *png*.



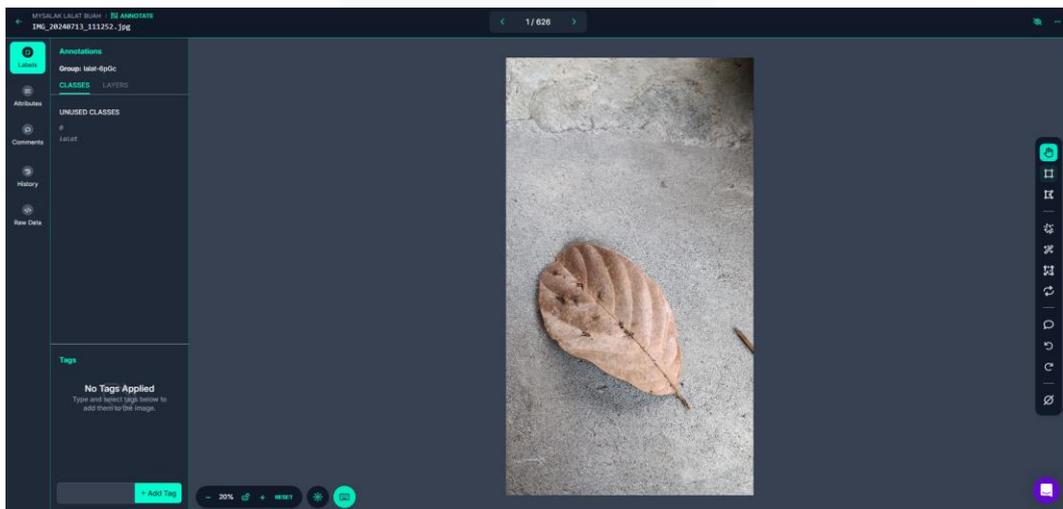
Gambar 3. 18 Tampilan untuk Mengunggah *Dataset*

Selanjutnya, setelah *dataset* berhasil diunggah, tampilan akan terlihat seperti pada Gambar 3.19. Untuk memulai proses anotasi, klik tombol “*Start Annotating*”.



Gambar 3. 19 Tampilan *Dataset* pada *Roboflow*

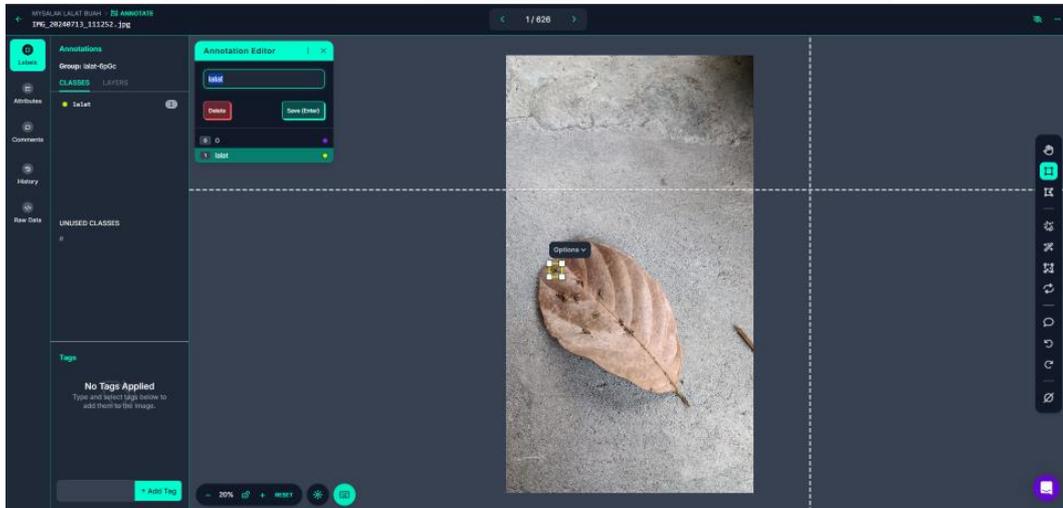
Tampilan untuk mengerjakan anotasi akan berubah seperti yang ditunjukkan pada Gambar 3.20. Anotasi dimulai dengan menggambar kotak pada lokasi yang terdapat objeknya.



Gambar 3. 20 Tampilan *Workspace* untuk Anotasi

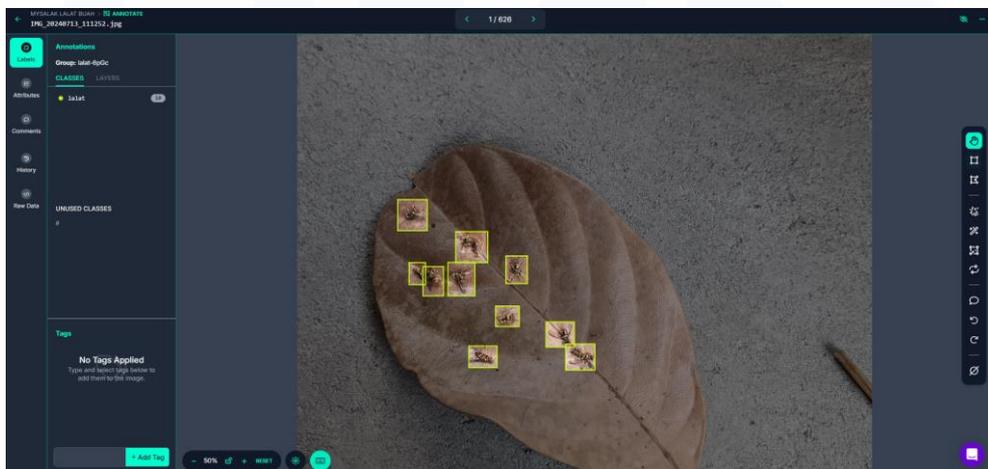
Setelah menggambar *bounding box* pada lokasi yang terdapat objek di gambar maka secara otomatis akan muncul pilihan label kelas seperti pada Gambar 3.21. Label kelas pada penelitian ini 2 yaitu 0 dan lalat. Label 0 merepresentasikan objek

dengan kondisi alat yang menggumpal sehingga sulit untuk diidentifikasi satu per satu. Sedangkan, label alat yaitu objek alat. Klik “*enter*” untuk mengidentifikasi kelas label pada objek yang sudah digambar *bounding box*.



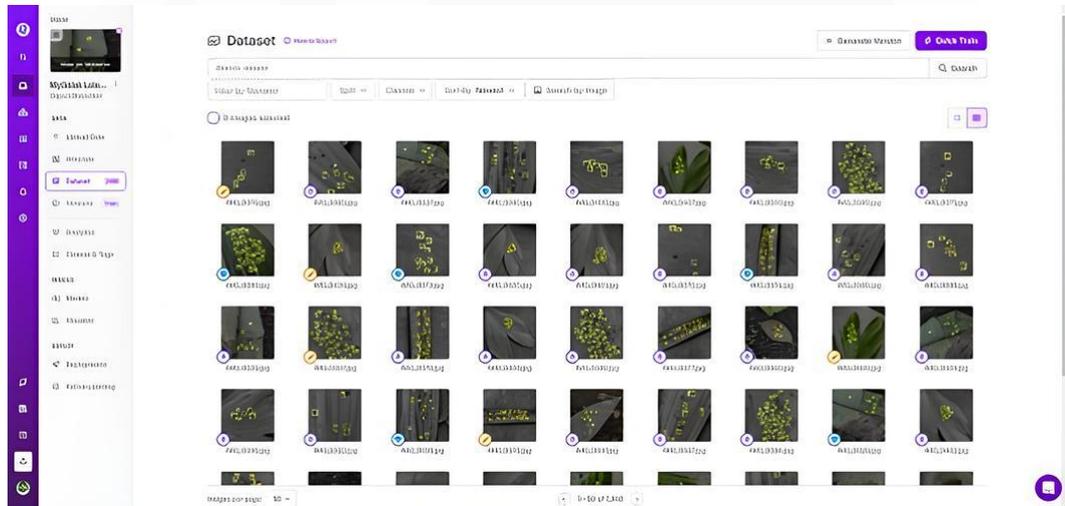
Gambar 3. 21 Proses Anotasi Gambar

Semua objek yang ada sudah dianotasi, maka tampilan gambar akan seperti Gambar 3.22. Pada kolom kiri terdapat informasi kelas dan jumlah objek pada gambar. Jika objek terlalu kecil digambar maka dapat melakukan *zoom in* agar saat menggambar Lokasi *bounding box* menjadi lebih akurat.



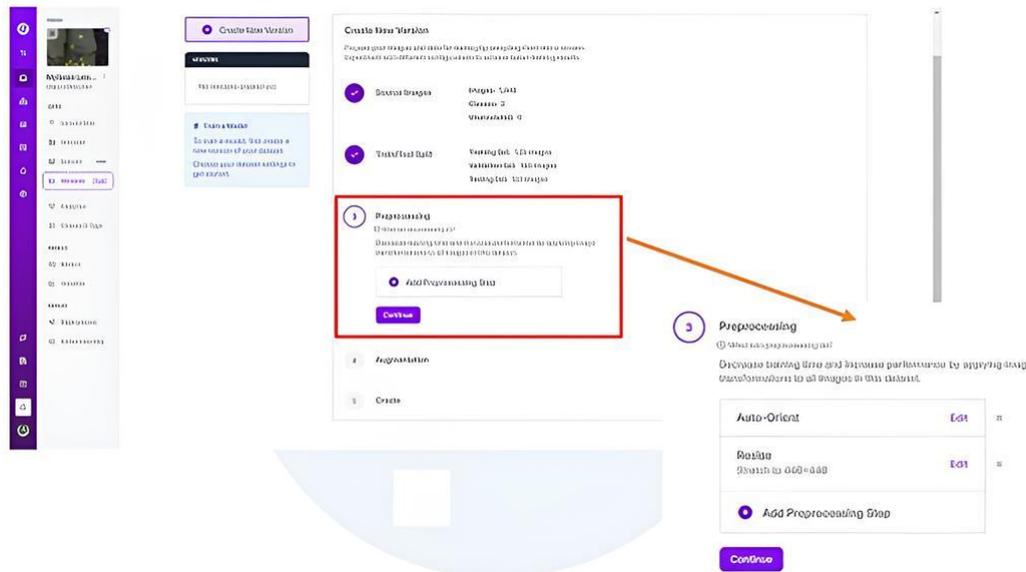
Gambar 3. 22 Tampilan Gambar Hasil Anotasi

Jika semua *dataset* sudah dilakukan anotasi maka tampilan akan seperti Gambar 3.23. Pada menu “*dataset*” ini juga dapat melakukan *review* hasil anotasi, jika ada yang salah maka dapat dilakukan *re-Anotate* pada gambar yang dipilih.



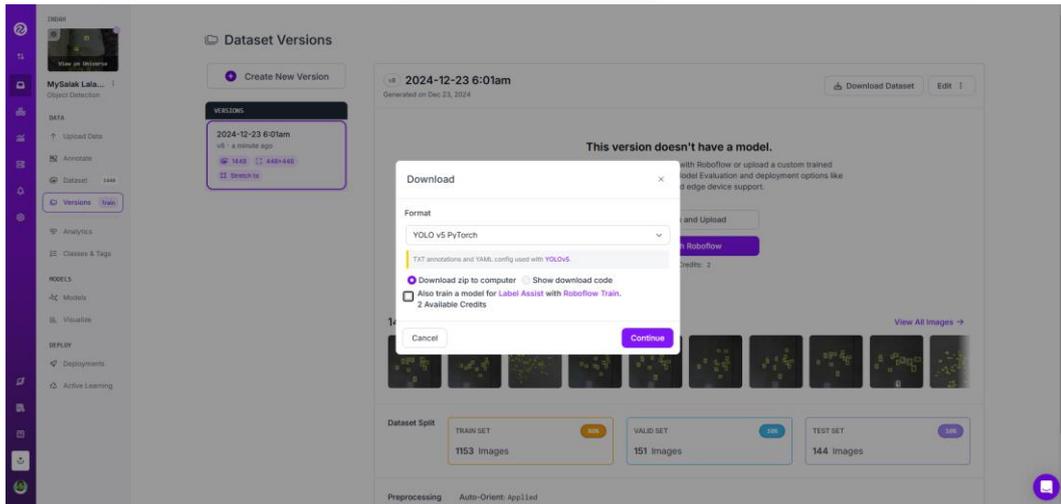
Gambar 3. 23 Hasil Anotasi *Dataset*

Tahap terakhir adalah pengunduhan *dataset*. Pengunduhan *dataset* dengan membuat “*version*” pada sistem *roboflow* seperti pada Gambar 3.24. Pada tahap *preprocessing* dapat menambahkan step “*auto-rotation*” dan “*resize*”. Jika tidak melakukan tahap tersebut maka *dataset* rusak ditandai dengan ketidaksesuaian *bounding box* dengan lokasi objek yang ada. Pada penelitian ini, *setting* nilai *resize* menjadi 448x448.



Gambar 3. 24 *Setting Dataset*

Jika *version dataset* sudah terbuat, maka tampilan akan berubah menjadi seperti pada Gambar 3.25. Klik “*download dataset*” pada tombol kanan atas, maka akan muncul pop-up untuk memilih mengunduh dalam format yang dibutuhkan. Pada penelitian ini menggunakan format *YoloV5 PyTorch*. Sehingga, *dataset* anotasi akan berupa gambar dan *file (.txt)* yang berisi lokasi *bounding box*. *Dataset* yang sudah *terunduh* selanjutnya dapat digunakan untuk *training* model.



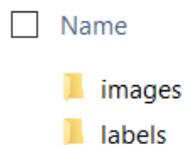
Gambar 3. 25 Download Dataset

Dataset yang telah dianotasi dan diunduh disimpan secara lokal oleh penulis dalam dua folder terpisah, yaitu dataset internal dan dataset eksternal. Meskipun disimpan dalam folder yang berbeda, kedua dataset memiliki struktur direktori yang sama yang dapat dilihat pada Gambar 3.26 dan Gambar 3.27, yaitu terdiri dari dua subfolder utama:

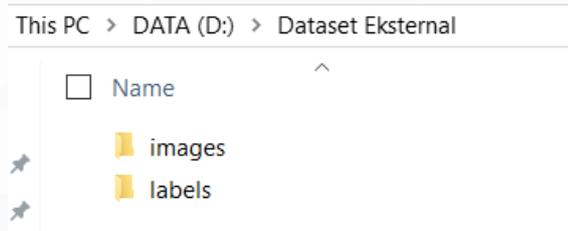
1. images/ → berisi file gambar lalat buah yang digunakan untuk pelatihan atau pengujian.
2. labels/ → berisi file .txt yang berisi informasi bounding box dari setiap objek lalat buah yang telah dianotasi.

Struktur ini memastikan bahwa dataset internal dan eksternal tersimpan secara terpisah, namun tetap memiliki tata letak yang seragam untuk memudahkan akses dalam proses pelatihan dan pengujian model deteksi lalat buah.

This PC > DATA (D:) > Dataset Internal

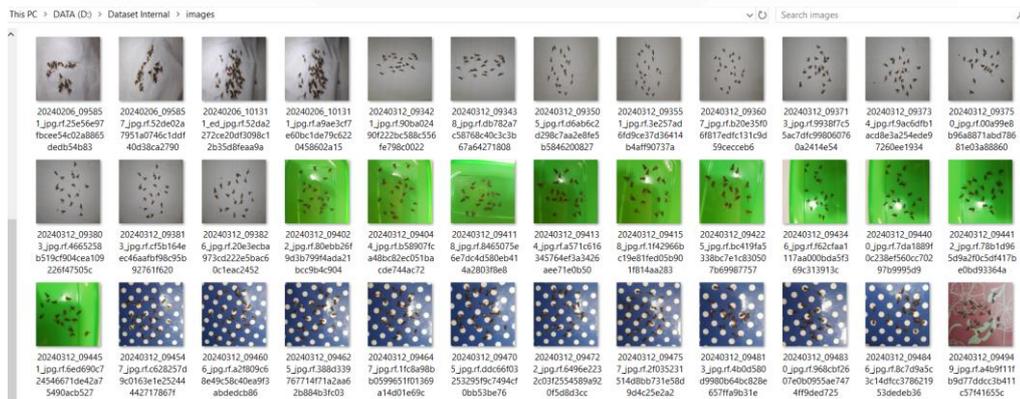


Gambar 3. 26 Struktur Folder Dataset Internal

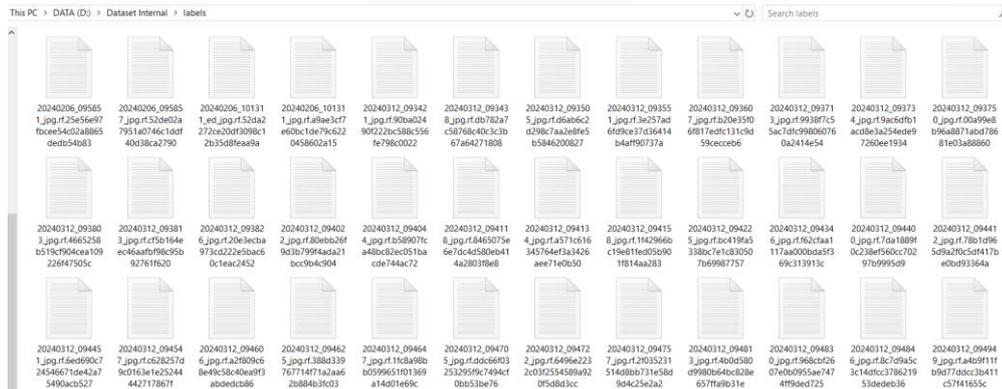


Gambar 3. 27 Struktur Folder Dataset Eksternal

Penamaan file pada masing-masing folder /images dan /labels ditampilkan secara konsisten pada Gambar 3.28 untuk dataset internal dan Gambar 3.29 untuk dataset eksternal. Gambar 3.28(a) menunjukkan penamaan file gambar dalam folder /images, sedangkan Gambar 3.28(b) menunjukkan penamaan file .txt dalam folder /labels untuk dataset internal. Pola penamaan yang sama juga diterapkan pada dataset eksternal, di mana Gambar 3.29(a) menampilkan penamaan file gambar dalam folder /images, dan Gambar 3.29(b) menampilkan penamaan file .txt dalam folder /labels.

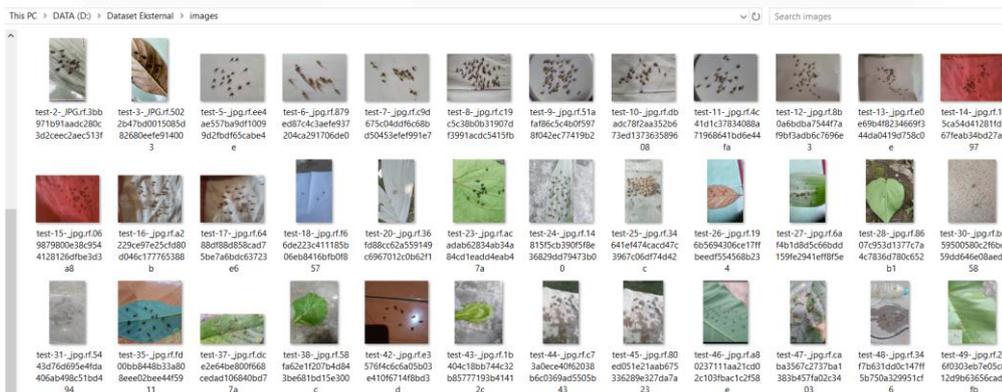


(a)

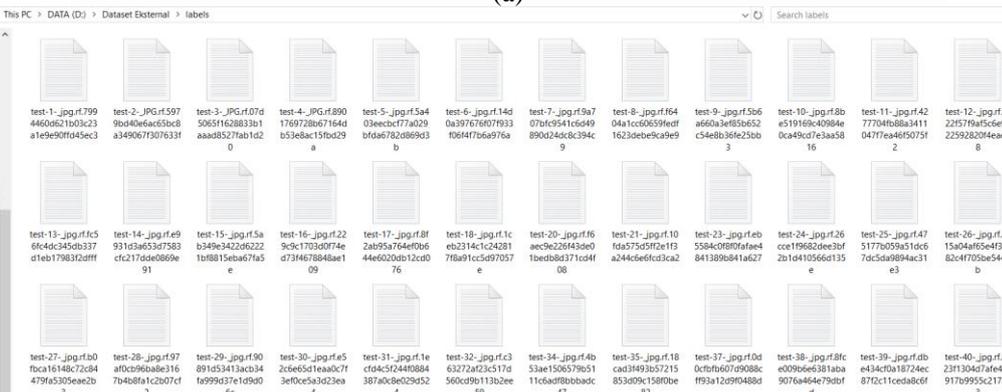


(b)

Gambar 3. 28 File Dataset Internal: (a)File Gambar dan (b) File .txt



(a)



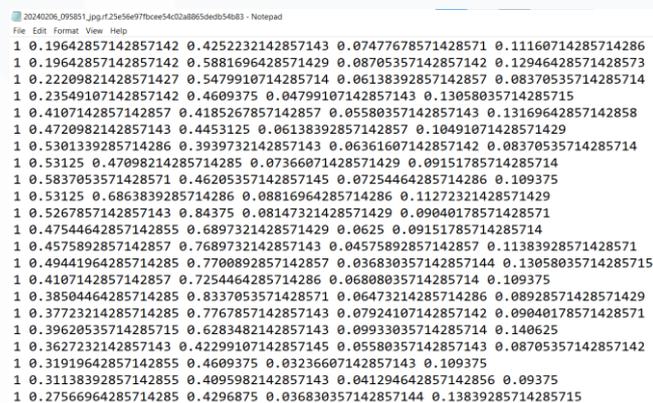
(b)

Gambar 3. 29 File Dataset Eksternal: (a) File Gambar dan (b) File .txt

Penamaan file dalam kedua dataset ini tampak lebih panjang dibandingkan nama file aslinya. Hal ini disebabkan oleh platform Roboflow, yang secara otomatis menambahkan hash unik ke dalam nama file. Penambahan hash ini bertujuan untuk

memastikan bahwa setiap file dalam dataset memiliki identitas unik, meskipun terdapat beberapa file dengan nama asli yang sama.

File .txt dalam dataset internal maupun eksternal berisi informasi bounding box, yang dapat dilihat pada Gambar 3.30. File ini mencakup indeks label kelas objek serta koordinat bounding box untuk setiap objek yang terdeteksi dalam gambar. Dalam dataset ini, label "0" digunakan untuk merepresentasikan kelas "0", sedangkan label "1" digunakan untuk merepresentasikan kelas "lalat". Selain itu, file .txt juga berisi empat nilai numerik yang menggambarkan koordinat bounding box, yaitu posisi tengah objek serta lebar dan tinggi dalam skala relatif terhadap ukuran gambar.



```
20240206_095851.jpg_r125e5e977bce54c02a8b65ed54b83 - Notepad
File Edit Format View Help
1 0.19642857142857142 0.4252232142857143 0.07477678571428571 0.11160714285714286
1 0.19642857142857142 0.5881696428571429 0.08705357142857142 0.12946428571428573
1 0.22209821428571427 0.5479910714285714 0.06138392857142857 0.08370535714285714
1 0.23549107142857142 0.4609375 0.04799107142857143 0.13058035714285715
1 0.4107142857142857 0.4185267857142857 0.05580357142857143 0.13169642857142858
1 0.4720982142857143 0.4453125 0.06138392857142857 0.10491071428571429
1 0.5301339285714286 0.3939732142857143 0.06361607142857142 0.08370535714285714
1 0.53125 0.47098214285714285 0.07366071428571429 0.09151785714285714
1 0.5837053571428571 0.46205357142857145 0.07254464285714286 0.109375
1 0.53125 0.6863839285714286 0.08816964285714286 0.11272321428571429
1 0.5267857142857143 0.84375 0.08147321428571429 0.09040178571428571
1 0.47544642857142855 0.6897321428571429 0.0625 0.09151785714285714
1 0.4575892857142857 0.7689732142857143 0.04575892857142857 0.11383928571428571
1 0.49441964285714285 0.7700892857142857 0.036830357142857144 0.13058035714285715
1 0.4107142857142857 0.7254464285714286 0.06808035714285714 0.109375
1 0.38504464285714285 0.8337053571428571 0.06473214285714286 0.08928571428571429
1 0.37723214285714285 0.7767857142857143 0.07924107142857142 0.09040178571428571
1 0.39620535714285715 0.6283482142857143 0.09933035714285714 0.140625
1 0.3627232142857143 0.42299107142857145 0.05580357142857143 0.08705357142857142
1 0.31919642857142855 0.4609375 0.03236607142857143 0.109375
1 0.31138392857142855 0.4095982142857143 0.041294642857142856 0.09375
1 0.27566964285714285 0.4296875 0.036830357142857144 0.13839285714285715
```

Gambar 3. 30 Label yang Berisi Lokasi Bounding Box

### 3.2.6. Validasi oleh *Expert* Pertanian

Selanjutnya, dilakukan validasi label oleh ahli di bidang pertanian untuk memastikan model dapat mengenali individu "lalat" dengan akurat dan menghindari kesalahan identifikasi. Proses validasi ini dilakukan dengan memeriksa setiap gambar yang telah diberi label "lalat" guna memastikan bahwa hanya lalat buah yang relevan yang masuk dalam *dataset*. Validasi ini dibantu oleh Riya Fatma Sari, *Research Assistant* Laboratorium Hama Tanaman Bagian Invertebrata Hama, Departemen Hama dan Penyakit Tumbuhan,

Fakultas Pertanian Universitas *Gadja* Mada (UGM). Proses ini sangat penting karena dalam beberapa kasus, ditemukan lalat yang bukan termasuk lalat buah yang menyerang salak.

Prosedur validasi dilakukan dengan mengamati setiap individu lalat buah dalam foto dan mencocokkan morfologinya dengan ciri khas spesies lalat buah salak. Karakteristik morfologi yang diperhatikan meliputi pola sayap dan warna tubuh. Proses validasi pada pola sayap dari lalat buah diidentifikasi dari beberapa bagian penting seperti yang ditunjukkan pada Gambar 3.31. Proses ini memerhatikan beberapa bagian yaitu basal costa (bagian pangkal dari tepi sayap), costa (tepi utama sayap yang membentang dari pangkal hingga ujung), microtrichia (struktur kecil seperti rambut halus di permukaan sayap), costal band (pola kosta sayap) yaitu pola garis gelap di sepanjang tepi sayap, dan anal streak (pola garis yang terletak di bagian belakang sayap) [8].



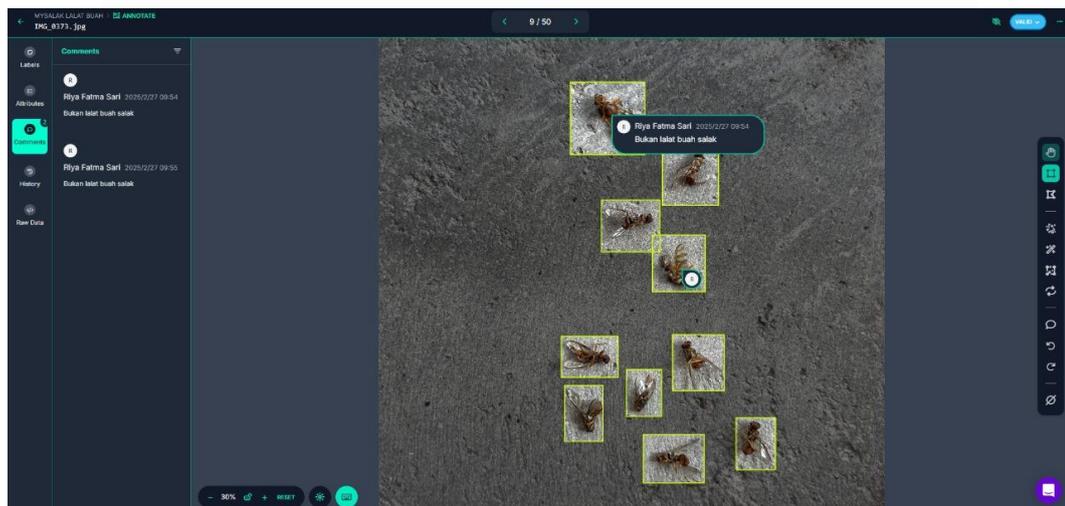
Gambar 3. 31 Pola Sayap Lalat Buah Salak [8]

Selain itu, lalat buah yang menyerang salak memiliki karakteristik unik, yaitu adanya siluet berwarna kuning keemasan pada bagian tubuhnya. Ciri khas ini dapat dilihat pada Gambar 3.32, yang ditandai dengan kotak merah. Meskipun kasus kesalahan identifikasi ini jarang terjadi, validasi yang teliti dapat meningkatkan akurasi model dalam mengenali dan membedakan lalat buah dengan lebih baik.



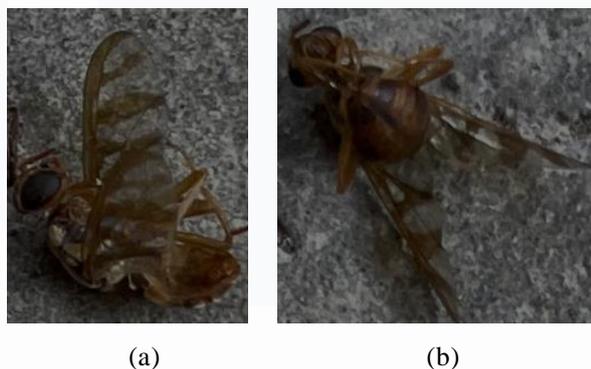
Gambar 3. 32 Lalat Buah

Validasi dilakukan hanya pada *workspace* “Indah” karena telah merepresentasikan *dataset* dari *workspace* lainnya, dengan total 1.448 gambar (*dataset* internal) dan 224 gambar (*dataset* eksternal). Oleh karena itu, *expert* hanya perlu melakukan pengecekan pada *workspace* tersebut. Sisa *workspace* yang ada akan diperiksa secara mandiri oleh penulis melalui laporan mingguan. Pengecekan dilakukan menggunakan fitur komentar pada platform yang tersedia. Komentar ini digunakan untuk menandai spesies lalat lain yang bukan termasuk lalat buah, sehingga tidak perlu diberikan label "lalat", seperti yang ditunjukkan pada Gambar 3.33.



Gambar 3. 33 Validasi Anotasi

Jenis lalat yang sering ditemukan di dalam perangkap dapat dilihat pada Gambar 3.34. Lalat-lalat tersebut bukan termasuk spesies lalat buah yang menyerang salak, karena memiliki karakteristik fisik yang berbeda dan umumnya sangat kecil kemungkinannya untuk menyerang salak dapat dilihat pada Gambar 3.20 bagian (a) dan (b). Meskipun lalat tersebut merupakan lalat buah, spesies ini lebih umum menyerang tanaman lain seperti mangga, cabai, jambu, dan tomat. Keberadaannya di perangkap dapat dimaklumi, mengingat lahan pertanian salak sering kali berdekatan dengan berbagai jenis tanaman lain yang juga menjadi habitat potensial bagi lalat buah.



Gambar 3. 34 Spesies Lain Lalat Buah: (a) Tampak Samping dan (b) Tampak Atas

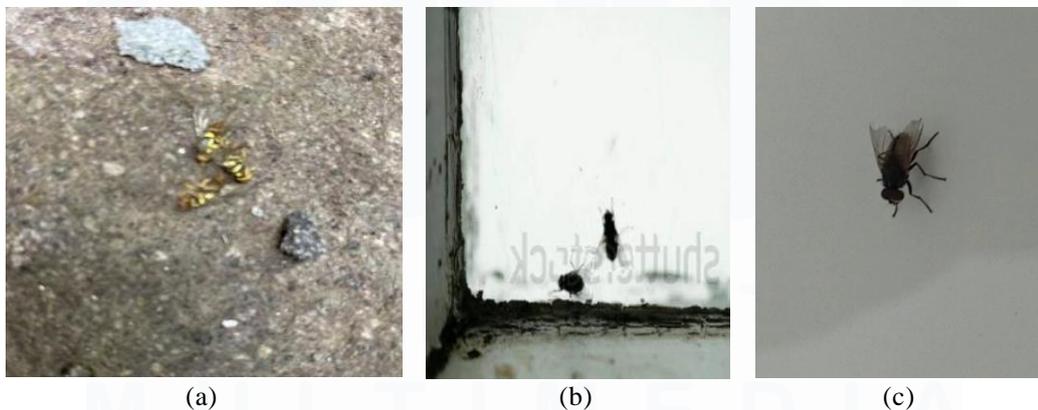
Berdasarkan hasil validasi oleh *expertise*, diperoleh hasil seperti yang ditunjukkan pada Tabel 3.4. Setelah melalui proses validasi, jumlah *dataset* internal berkurang menjadi 1.448, sedangkan *dataset* eksternal mengalami penurunan menjadi 148. Penurunan jumlah *dataset* ini disebabkan oleh beberapa faktor, terutama kualitas gambar yang kurang jelas, seperti gambar yang buram, pencahayaan yang buruk, atau objek yang tidak terlihat dengan jelas. Akibatnya, objek dalam gambar tidak dapat diidentifikasi dengan baik.

Tabel 3. 4 Hasil Validasi

| <i>Dataset</i> | <b>Jumlah</b> |
|----------------|---------------|
| Internal       | 1448          |
| Eksternal      | 148           |

### 3.2.7. Perbandingan *Dataset Online* dan *Dataset Lalat Buah Salak*

Saat ini, terdapat beberapa *dataset* lalat yang tersedia secara *online*. Namun, *dataset* tersebut berasal dari spesies lalat yang berbeda dengan lalat buah yang menyerang salak, seperti yang ditunjukkan pada Gambar 3.35 [9][10][11]. Pada Gambar 3.25 bagian (a), dapat dilihat bahwa secara morfologi, lalat buah yang menyerang salak memiliki karakteristik khusus, seperti warna tubuh siluet kuning keemasan dan pola sayap yang berbeda dibandingkan dengan spesies lalat lain yang umum ditemukan dalam *dataset online*, sebagaimana ditampilkan pada Gambar 3.35 bagian (b) dan (c). Pada Gambar 3.35 bagian (b) merupakan spesies lalat hitam dan bagian (c) lalat rumah yang menunjukkan spesies lalat dengan ciri khas warna tubuh yang cenderung dominan hitam serta pola sayap yang memanjang. Bukan hanya perbedaan morfologi saja, namun berikut pada Tabel 3.5 ditampilkan perbedaan *dataset online* dengan *dataset* lalat buah salak.



Gambar 3. 35 Gambar Lalat: (a) Lalat Buah Salak, (b) Lalat Hitam, (c) Lalat Rumah

Tabel 3. 5 Perbandingan *Dataset* Online dengan *Dataset* Lalat Buah Salak

| Parameter              | Dataset Online   | Dataset Lalat Buah Salak                              |
|------------------------|--|---|
| Jumlah Gambar          | 2789 gambar  | 6008 gambar   |
| Jenis Spesies          | Lalat rumah ( <i>Musca domestica</i> )<br>Lalat Hitam ( <i>Hermetia illucens</i> ) | Lalat Buah Salak ( <i>Bactrocera Dorsalis</i> )       |
| Lokasi Pengambilan     | Lalat ini umumnya ditemukan di sekitar area permukiman atau di dalam rumah         | Lahan pertanian salak Sleman, Yogyakarta              |
| Resolusi Gambar        | Bervariasi (128x128, 416x416, dan 640x640)   | 448x448 (diseragamkan)                                |
| Variasi latar belakang | Latar belakang berwarna putih  | Kayu, daun, dan ubin                                  |
| Kondisi Dataset        | Individu lalat sempurna  | Basah, kering, individu utuh, hanya toraks, dan rusak |

Perbedaan yang ditampilkan pada Tabel 3.5 ini menyebabkan *dataset online* yang ada tidak dapat digunakan secara langsung untuk mendeteksi lalat buah pada salak. Sebaliknya, *dataset* lalat buah yang dikumpulkan secara langsung dari lahan pertanian salak sesuai dengan kondisi nyata. *Dataset* ini mencerminkan berbagai variasi lingkungan dan kondisi lalat buah yang sesungguhnya di lahan pertanian salak. Oleh karena itu, meskipun terdapat *dataset* lalat yang tersedia secara *online*, penggunaan *dataset* lalat buah yang dikumpulkan sendiri menjadi solusi yang lebih optimal dalam pengembangan sistem deteksi lalat buah pada salak.

### 3.3 Kendala yang Ditemukan

Kendala yang dihadapi selama melakukan pengumpulan dan pelabelan *dataset* untuk pengembangan menu sistem otomatis untuk lalat buah adalah:

1. Ketidakseimbangan *dataset* yang memiliki label “0” dengan “lalat”. Sehingga membuat hasil *metric* evaluasi yang dihasilkan tidak optimal dan deteksi menjadi kurang akurat. Meskipun sudah dilakukan beragam cara *pre-processing*.
2. Ukuran *dataset* yang besar saat anotasi.

### 3.4 Solusi atas Kendala yang Ditemukan

Adapun solusi untuk menghadapi kendala yang telah ditemukan, antara lain:

1. Pengumpulan *dataset* untuk *project* selanjutnya cukup berfokus pada gambar “lalat” saja.
2. Jaringan yang stabil agar anotasi yang dilakukan lancar.