BAB III METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dimulai pada November 2022 sampai dengan April 2023. Penelitian ini dilakukan Laboratorium Terpadu Jurusan Teknik Elektro Universitas Muhammadiyah Malang.

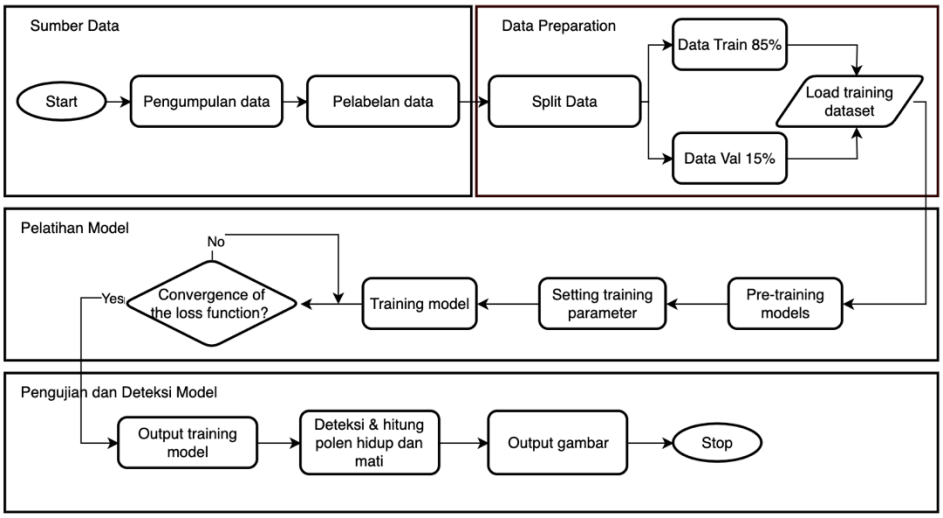
3.2 Alat dan Bahan

Adapun alat dan bahan untuk mendukung penelitian ini, berikut perangkat serta bahan yang digunkaan:

1. Satu unit laptop dengan system operasi macOS Monterey dengan spesifikasi CPU *dual-core* Intel i5 2,7 GHz, SSD 500 GB, RAM 8 GB. Sebagai media untuk menunjang proses pelabelan dan uji model menggunakan perangkat keras laptop.
2. Super komputer dari DIKTI AI-Center dengan spesifikasi CPU 16 *core* AMD EPYC 7742 dan 2 × 40 GB, GPU NVIDIA GDX-A100 dengan RAM 128 GB dengan sistem operasi yang berjalan menggunakan Ubuntu 20.04.
3. *Software* yang digunakan pada penelitian
4. ini yaitu, *Visual Studio Code* sebagai tempat untuk menulis dan mengedit kode program, *LabelImg* digunakan untuk anotasi dataset, dan *Jupyter Notebook* digunakan untuk penulisan dan menjalankan kode program *python* dengan format.ipynb, program ini ditulis menggunakan versi *python* 3.9.16.
5. *Terminal* sebagai *running script* code program.

3.3 Perancangan Program Pengolahan Gambar

Perancangan program ini bertujuan untuk mengolah gambar yang dapat dilihat pada diagram alir seperti gambar 3.1 berikut



Gambar 3.1 Tahapan Perancangan Program pengolahan Gambar

3.3.1 Pengumpulan Data

Data yang diambil berupa citra gambar polen bunga jantan kelapa sawit dengan format JPG (Joint Photographic Group) yang ditangkap melalui kamera mikroskop sejumlah 415 data termasuk data yang sudah augmentasi dan akan di resize sebesar 640x640 pixels.

3.3.2 Pelabelan Data

Setelah data terkumpul, selanjutnya dilakukan proses pelabelan data menggunakan anotasi *bounding box* yaitu proses pelabelan objek dengan *rectangular boxes* melalui aplikasi *labelimg*. Proses pelabelan data ini mengklasifikasikan objek kedalam dua kelas yaitu hidup dan mati, dan diseting menggunakan format anotasi YOLO pada aplikasi.

3.3.3 Preparasi Data

Proses ini bertujuan untuk membagi data kedalam dua bagian yaitu data latih, data validasi dan data uji. Dimana data latih merupakan data yang digunakan untuk melatih model dalam deep learning, data validasi merupakan data uang digunakan untuk mengukur kemampuan model yang telah dilatih dalam memprediksi suatu objek, data latih dan data validasi ini digunakan pada saat proses pelatihan model sedangkan data uji merupakan data yang digunakan

untuk menguji model yang telah dilatih apakah memiliki akurasi yang baik atau belum dalam mengenali objek.

3.3.4 Pelatihan Model

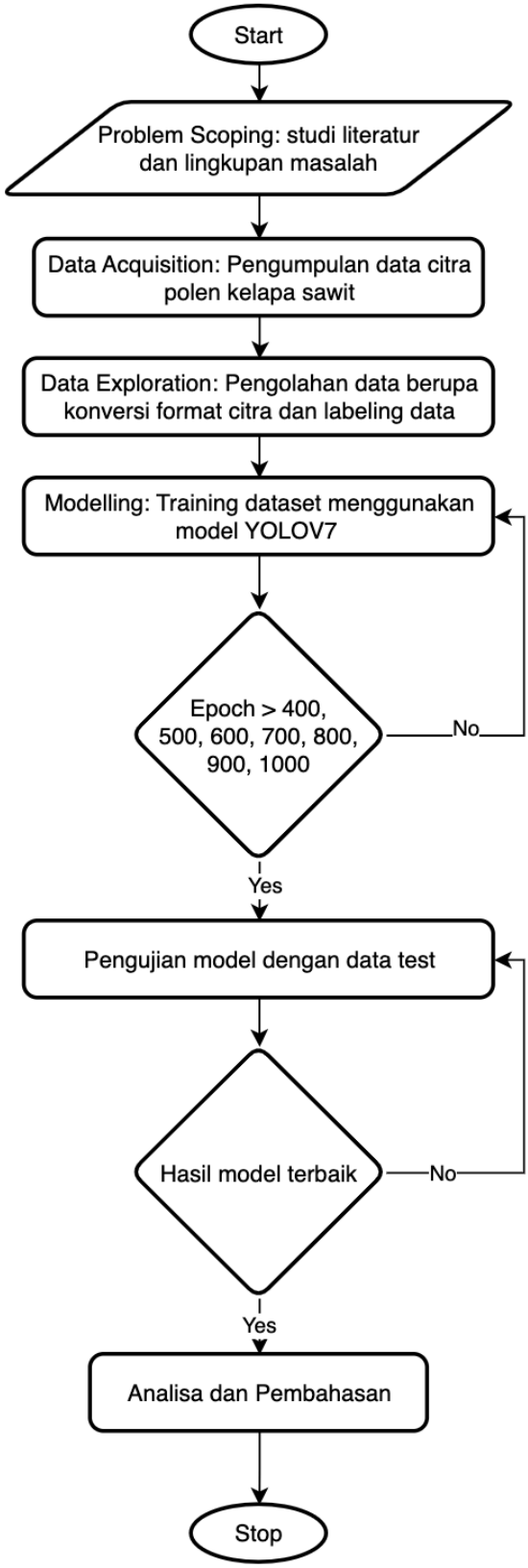
Pelatihan model merupakan proses dimana untuk mengajarkan sistem dan mengenalkan mesin agar dapat mendeteksi objek dan menghitungnya sesuai dengan data latih dan data validasi yang sudah diberi label dengan kelas yang telah ditentukan, yaitu polen hidup dan mati. Pada saat proses pelatihan model ini menggunakan model YOLOV7x dengan *resize* ukuran gambar sebesar 640 x 640 pixel.

3.3.5 Pengujian Model

Hasil dari pelatihan model (*training model*) ini berupa bobot (*weight*) selanjutnya dilakukan pengujian dengan data *test*. Proses deteksi didapatkan setelah proses *training* data menggunakan model YOLOV7x agar dapat mengklasifikasikan polen hidup dan mati secara otomatis. Selanjutnya sistem dapat menghitung polen berdasarkan deteksi dari *bounding box* yang dilakukan sistem setelah dilakukan *training* data, yang kemudian hasil penghitungan tersebut ditampilkan dalam citra polen untuk memudahkan user mengetahui viabilitasnya, proses ini dilakukan dengan menggunakan aplikasi *web streamlit* dengan minimal *confidence threshold* 0,5 agar menghindari terjadinya deteksi ganda pada objek. Apabila belum memiliki hasil yang akurat maka dilakukan evaluasi dan dilakukan pelatihan data kembali hingga mendapatkan hasil yang akurat.

3.4 Tahapan Penelitian

Alur kerja dan metode yang digunakan dalam penelitian ini digambarkan dalam diagram alir pada gambar 3.2 berikut.



Gambar 3.2 Tahapan Penelitian

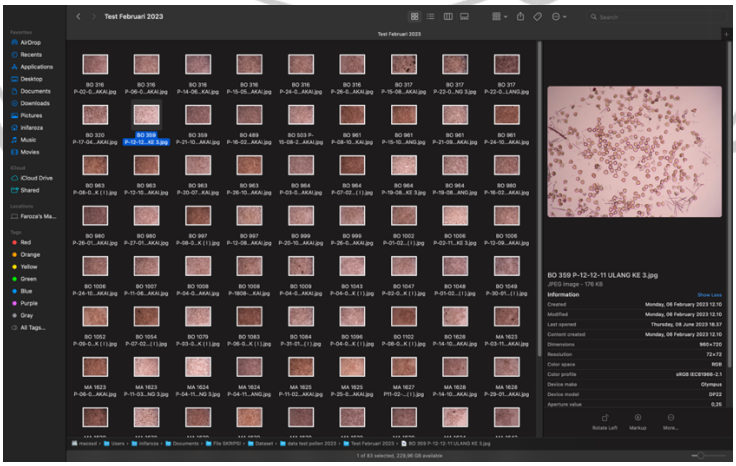
Tahapan penelitian yang dilakukan pada penelitian yang berjudul "*Palm Oil Pollen Detection and Counting System Using YOLOv7 Algorithm*" atau implementasi algoritma YOLOv7 untuk deteksi dan menghitung pollen (serbuk sari) kelapa sawit adalah sebagai berikut:

3.4.1 *Problem Scoping*

Proses ini dilakukan untuk mengidentifikasi permasalah yang ada dan membuat solusi dengan pendekatan AI. Pada tahapan awal sebelum penelitian, untuk mempelajari materi, memehami materi dari berbagai sumber literature, memahami pustaka dan dasar teori, serta referensi terkait sistem yang akan digunakan untuk menunjang penelitian. Hal ini dilakukan untuk mendapatkan pemahaman konsep, metode, penelitian terdahulu, teknologi dan informasi yang dapat mendukung penelitian ini, dan juga yang berkaitan dengan *image processing*, *object detection* antara lain kebutuhan data diawal agar dapat menunjang pengolahan, *tools* yang dibutuhkan, bagaimana mengimplementasikan metode *image processing*, sampai bagaimana data dapat diproses ke dalam metode. Penelitian ini menggunakan dasar teori berupa buku/ *e-book*, jurnal, dan penelitian terdahulu

3.4.2 *Data Acquistion*

Tahapan ini dilakukan pengumpulan dataset pollen kelapa sawit, data diambil dari hasil citra mikroskopik pollen bunga jantan kelapa sawit yang diperoleh dari PT. RPN Pusat Penelitian Kelapa Sawit Medan dengan izin perusahaan.

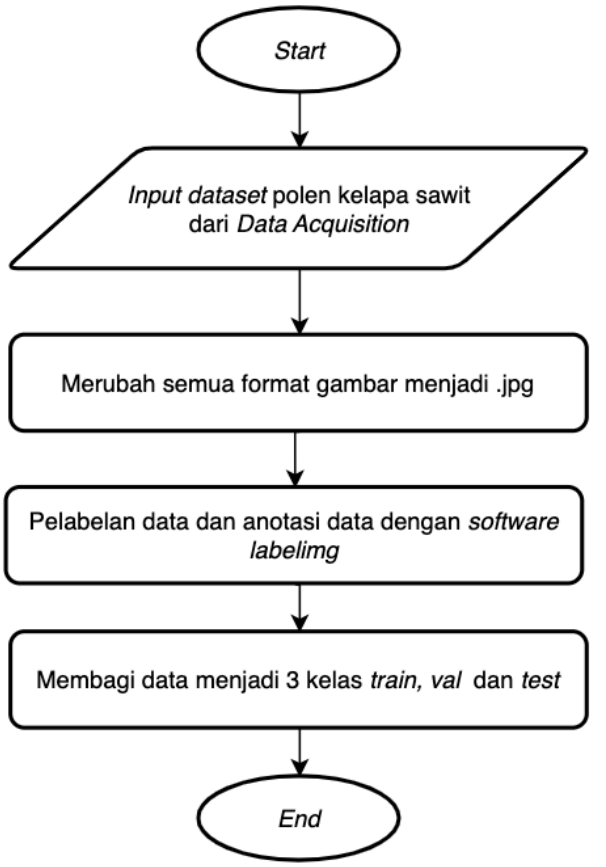


Gambar 3.3 Dataset Pollen Kelapa Sawit

Format *citra* polen menggunakan *Joint Photographic Group* (JPG) yang diambil menggunakan kamera *mikroskop OLYMPUS CX31* dengan dimensi yang di *capture* 960 × 720, 1600 × 1200, dan 1920 × 1440 pixel. Total foto yang terkumpul ±1200 gambar polen tampak seperti gambar 3.2.

3.4.3 *Data Exploration*

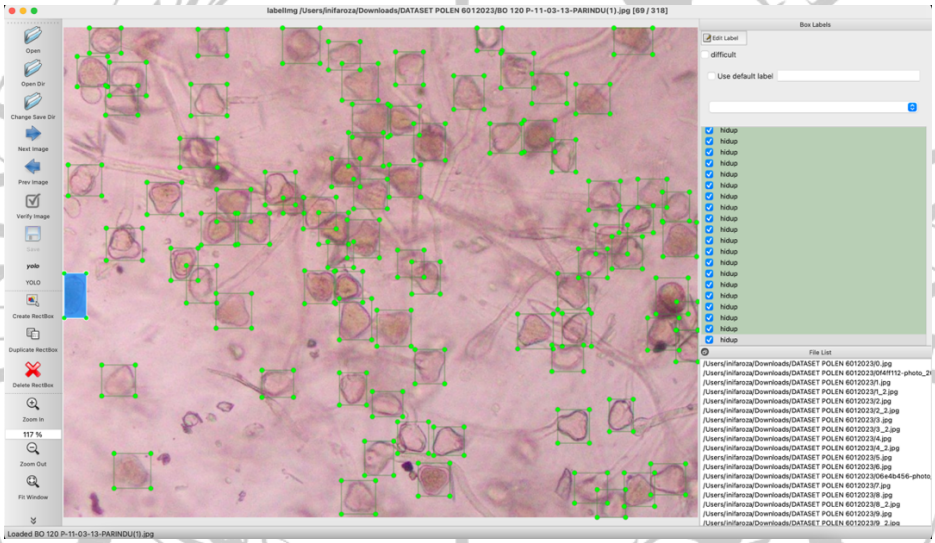
Pada tahapan ini dataset yang telah terkumpul dan sudah dipilih sesuai kebutuhan akan diolah terlebih dahulu. Dilakukanya pengolahan foto ini agar *format* setiap citra sesuai dengan syarat algoritma pendeteksian yang dibuat. Gambar 3.3 berikut menjelaskan alur kerja untuk tahap *Data Exploration*.



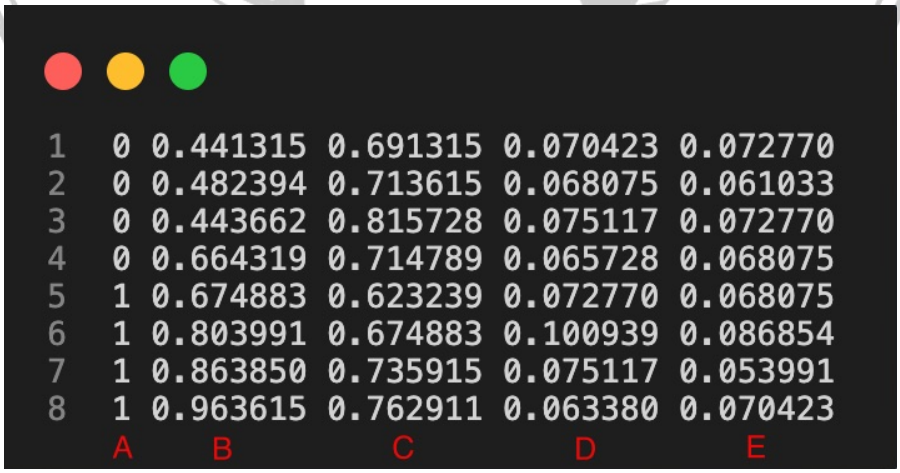
Gambar 3.4 *Flowchart Data Exploration*

Berdasarkan gambar 3.4 ketika *dataset* polen yang berupa foto gambar mikroskop terkumpul, maka data akan diolah. Data yang terkumpul tersebut akan dirubah format gambar secara keseluruhan. Berdasarkan gambar yang diperoleh terdapat beberapa gambar dengan format '.png'. Gambar akan di *convert* menjadi format '.jpg' agar ukuran file gambar menjadi kecil tanpa

merubah kualitas gambar. Selanjutnya gambar akan dilakukan proses *labeling* dan *anotasi* menggunakan *software labelimg*. Proses ini bertujuan memberikan keterangan berupa nama atau kata kunci pada suatu objek. *Anotasi* gambar dilakukan dengan cara memberikan pembatas pada suatu objek, proses ini menggunakan *anotasi rectangular boxes* merupakan sebuah anotasi dengan memberikan kotak pembatas pada objek tertentu secara manual, yang nantinya akan mudah dikenali oleh mesin dengan dua kelas yaitu polen hidup dan polen mati. Pada penelitian ini gambar polen diklasifikasikan menjadi dua jenis kelas yaitu kelas pollen hidup dengan kode kelas (0) dan pollen mati untuk polen yang berwarna hitam dengan kode kelas (1). Dataset polen ini menggunakan format dataset YOLO yang sudah tersedia pada *software labelimg,* file *anotasi* disimpan dalam format berbentuk teks (.txt).



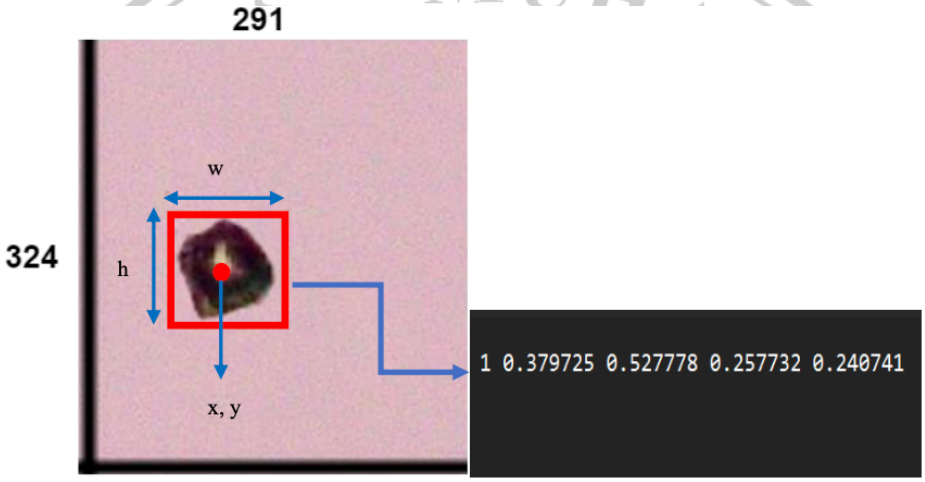
Gambar 3.5 Pelabelan gambar polen



```
1  0 0.441315 0.691315 0.070423 0.072770
2  0 0.482394 0.713615 0.068075 0.061033
3  0 0.443662 0.815728 0.075117 0.072770
4  0 0.664319 0.714789 0.065728 0.068075
5  1 0.674883 0.623239 0.072770 0.068075
6  1 0.803991 0.674883 0.100939 0.086854
7  1 0.863850 0.735915 0.075117 0.053991
8  1 0.963615 0.762911 0.063380 0.070423
   A    B        C        D        E
```

Gambar 3.6 Contoh hasil file *txt* pelabelan gambar

Gambar 3.6 menunjukan keluaran hasil pelabelan data adalah hasil anotasi gambar yang menggunakan aplikasi *labelimg.* Pada bagian A adalah kode dari kelas dalam pelabelan data, yang dimana 0 adalah kode kelas polen hidup dan 1 adalah kode polen mati. Bagian B adalah *x_center,* sedangkan C *y_center,* D adalah *width* lebar anotasi sedangkan E adalah *height* tinggi dari *anotasi* objek. Koordinat pada kotak *anotasi* harus dinormalisasi dengan dimensi gambar yaitu memiliki nilai antara 0 dan 1, nilai pada kelas berindeks nol atau dimulai dari 0. Untuk menentukan nilai *x_center, y_center, width* dan *height* ditampilkan dalam gambar 3.3.



Gambar 3.7 Kotak Pembatas Pelabelan Objek

Dapat diketahui pada Gambar 3.7 diatas, bahwa terdapat sebuah citra dengan resolusi 291 x 324 piksel yang terdapat kotak pembatas pada objek dengan kode nilai kelas 1 yang berarti polen mati, x.*center* 0.379725, y.*center* 0.527778, *width* 0.257732, dan *height* 0.240741. Untuk menentuan nilai x dan y *center* ini berdasarkan titik tengah objek, sedangkan nilai w dan h berdasarkan nilai dari panjang dan lebar kotak pembatas dibandingkan ukuran citra. Nilai-nilai tersebut sudah dilakukan normalisasi agar dapat menghindari masalah dengan resolusi gambar yang berbeda-beda. Resolusi gambar dapat bervariasi tergantung pada perangkat dan sumber gambar. Dengan normalisasi, koordinat kotak pembatas dapat tetap relatif terhadap gambar aslinya. Untuk dapat mengetahui nilai koordinat yang sebenarnya dapat dilakukan dengan perhitungan sebagai berikut.

x = x.*center* × lebar citra

x = 0.379725 × 291 = 110.49

y = y.*center* × tinggi citra

y = 0.527778× 324 = 171

w = *width* × lebar citra

w = 0.257732 × 291 = 75

h = *height* × tinggi citra

h = *0.240741* × 324 = 78

Tabel 3.1 Pembagian dataset citra pollen

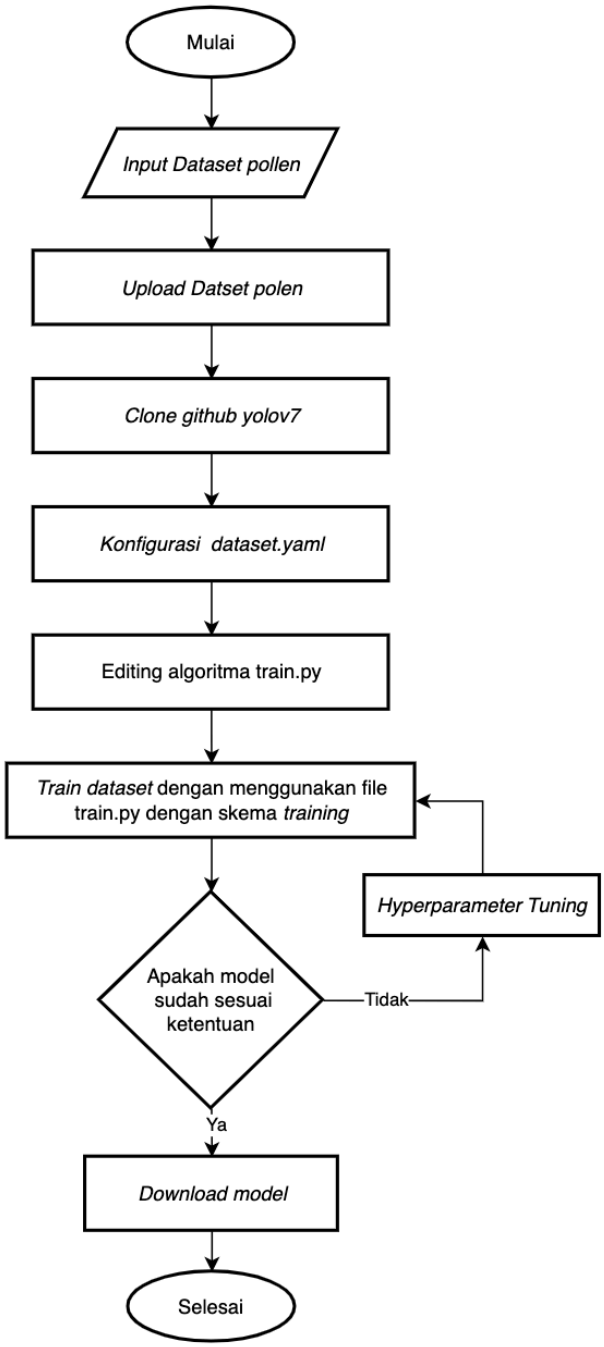
Type Data	**Jumlah**	**%**
Train	350	85%
Valid	71	15%
Test	6	5%
Total data	422	100%

Sebanyak 422 gambar data yang terkumpul akan dikelompokan dalam kelas *training, valid,* dan *test* sesuai kebutuhan dataset *training* seperti data tabel 3.1 pembagian dataset citra pollen.

3.4.4 *Modelling*

Pada tahapan *modelling* ini, dataset yang sudah sesuai akan dilatih (*training*) untuk membuat sebuah model baru. *Training Dataset* menggunakan *Visual code* yang sudah terinstall *extension Jupyter Notebook.* Proses modeling ini dapat dilihat pada gambar 3.8 sebagai berikut. Gambar 3.8 menunjukan bahwa pada tahap *modelling* ini dimulai dengan *dataset* polen yang ada akan di unggah ke delam *folder* data pada *folder* YOLOv7 pada *server* yang akan digunakan untuk preses *training model.* Selanjutnya *clone file* yolov7 dari *repository Github* YOLOv7. Setelah file YOLOv7 dilakukan *clone* kemudian konfigurasi file *dataset.yaml* untuk menetukan dimana letak *dataset file original* YOLOv7 berada. Selanjutnya melakukan konfigurasi dan editing pada file *train.py* untuk menentukan hasil *output* seperti desain sistem yang ada. Tahap selanjutnya *dataset* akan melakuakan proses *training* dengan cara menjalankan *file script train.py* pada terminal, dengan beberapa skema yang sudah ditentukan.

Parameter yang diubah antara lain *image size, batch size,* dan *epoch.* Tahapan ini dilakukan metode '*trial and error'* saat pelatihan model agar menghasilkan hasil akurasi yang tinggi. Untuk menganalisis dam mengamati performa dari model menggunakan *metrix evaluasi, metrix evaluasi* ini antara lain *Confusion Matrix, Precision, Recall, AP, mAP, Loss, dan Accuration.* Setelah melakukan berbagai kombinasi parameter, didapatkan model terbaik untuk konfigurasi *image size* 640, *batch* 16, dan 900 epoch.



Gambar 3.8 *Flowchart Modelling*

3.4.5 *Evaluation*

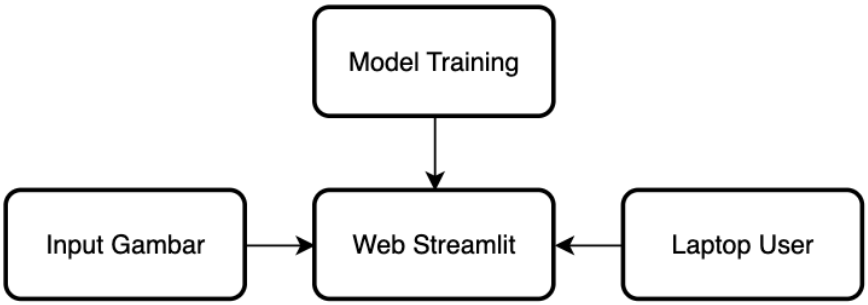
Pada tahapan evaluasi terdapat dua tahapan evaluasi, pertama evaluasi ketika pada tahap *modelling* yang dimana model akan di evaluasi kinerja skema konfigurasi saat proses *training dataset.* Setelah proses *training dataset* selesai akan dilakukan evaluasi terhadap hasil model beberapa kali sampai dianggap sesaui maka didapatkanlah hasil pelatihan model terbaik. Hasil *training* dari model, kemudian dilanjutkan uji deteksi model terhadap *datatest test*. Saat proses uji deteksi objek juga dilakukan beberapa konfigurasi khusus dengan parameter tertentu berupa *image size, dan confidence score.* Tahap ini juga dilakukan metode *(trial and error)* pada setiap konfigurasi yang dilakukan selama skema pelatihan, sehingga akan didapatkan konfigurasi skor akurasi tertinggi yang dapat dijalankan untuk semua *dataset*.

3.4.6 Analisa data dan Pembahasan

Tahap ini merupakan proses untuk memastikan hasil akurasi apakah sudah sesuai dengan tujuan dari penelitian ini yaitu dapat mendeteksi dan menghitung polen yang kemudian dilanjutkan ke tahap *deployment model*.

3.5 Perancangan Sistem Aplikasi Web *Streamlit*

Perancangan sistem aplikasi web ini untuk melakukan *deployment* dari hasil pelatihan model dengan aplikasi web *streamlit*. Pengguna dapat mengakses aplikasi web tersebut dan memasukkan citra polen kemudian di proses oleh aplikasi web streamlit yang sudah dilakukan konfigurasi pada laptop *user* dan output akan tersimpan secara otomatis di laptop *user*. Berikut Gambar 3.9 merupakan diagram blok perancangan sistem.



Gambar 3.9 Flowchart Perancangan Sistem

Gambar 3.9 menujukan bahwa, selama proses awal yang dilakukan saat melakukan deployment, sistem mengambil bobot (*weight*) dari hasil *training* yang telah dilakukan dan menghubungkannya dengan program *streamlit* yang sudah diprogram dan dikonfigurasi pada laptop *user*. Setelah bobot dari hasil pelatihan model terkonfigurasi dengan *streamlit*, maka *user* dapat memasukkan gambar polen kelapa sawit kemudian akan diproses oleh *streamlit* dan hasilnya akan muncul di laptop *user*. Laptop *user* yang sudah disetting akan mengambil data dari input citra, selanjutnya akan memproses gambar polen tersebut menggunakan *streamlit* untuk menampilkan hasil deteksi dan penghitungan polen.

UNIVERSITAS MUHAMMADIYAH MALANG