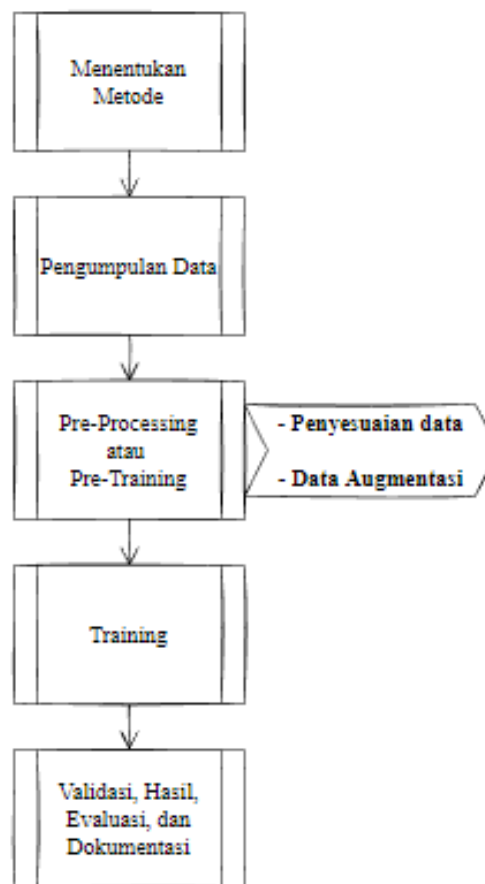


BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Metode Penelitian

Untuk pelaksanaan penelitian ini, ada beberapa tahap yang harus dilakukan diantaranya adalah seperti pada penjelasan di bawah ini atau dapat dilihat pada gambar 3.1 di bawah ini.



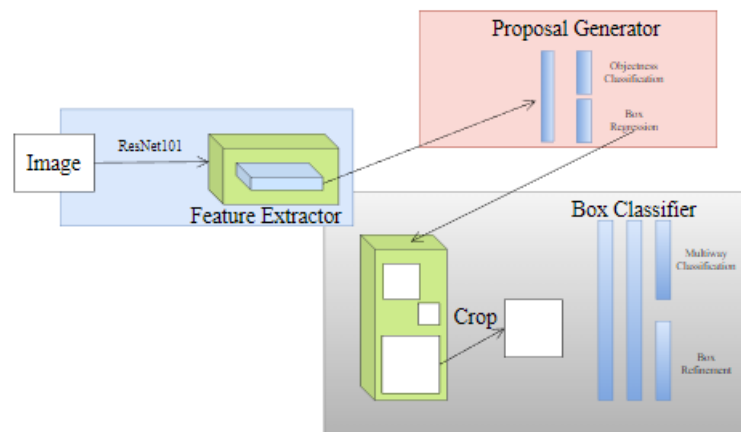
Gambar 3. 1 Tahapan Pelaksanaan Metode Penelitian

3.1.1 Metode Implementasi

Setelah melakukan studi literatur, penulis menentukan metode yang akan digunakan dalam penelitian ini. penelitian ini akan menggunakan metode pendeteksian yang berbasis jaringan konvolusi

atau CNN khususnya Faster R-CNN ResNet101 yaitu Faster Region-Based Convolutional Neural Network dengan ResNet101 sebagai metode klasifikasinya. R-CNN sendiri awalnya pada tahun 2015 dibuat sebagai metode pendeteksian objek yang mengkombinasikan algoritma Region Proposal Network (RPN) dan Convolutional Neural Network (CNN) (Girshick et al., 2014). Setelah itu dikembangkan lagi menjadi Fast R-CNN namun masih memiliki kekurangan, salah satunya adalah adanya bottleneck pada RPN sehingga tidak dapat menyamai kecepatan CNN. Oleh karena itu Faster R-CNN merupakan solusi dari kelemahan Fast R-CNN dengan mempercepat proses RPN sehingga mengurangi terjadinya bottleneck [15].

Arsitektur Faster R-CNN ini dikombinasikan dengan ResNet 101 sebagai feature extractor. Dapat dilihat melalui gambar 3.2 di bawah ini.



Gambar 3. 2 Faster R-CNN dengan ResNet101 Feature Extractor

Seperti yang terlihat pada gambar diatas, dapat terlihat bahwa Faster R-CNN sebagai meta-architecture dengan banyak pilihan feature extractor. Dalam penelitian ini, penulis memilih menggunakan ResNet101 dengan pertimbangan runtime yang tidak terlalu lama dan hasil presisi dan akurasi yang cukup baik.



Gambar 3. 3 Arsitektur yang digunakan

Gambar 3.3 diatas menggambarkan lapisan dari arsitektur model Faster R-CNN ResNet101 yang digunakan, cara kerja model adalah saat menerima input gambar dengan ukuran tertentu, gambar selanjutnya diproses oleh 4 layer konvolusi dengan keterangan kernel masing-masing layer seperti terlihat pada gambar, total dari ke-4 layer konvolusi berjumlah 101 layer. Feature map yang didapatkan selanjutnya digunakan untuk mencari region of interest atau objek-objek pada gambar, ROI juga didapatkan dari region proposal network yang menerima feature map sebagai input untuk memprediksi bounding box pada gambar, selanjutnya kedua hasil akan di-pooling dan diberikan kepada fully connected layer untuk mendapatkan bounding box prediction beserta hasil klasifikasi dari setiap bounding box yang diprediksi. Sehingga secara keseluruhan model dapat mendeteksi dan mengklasifikasi objek dari gambar.

3.1.2 Pengumpulan Data (Dataset)

Untuk pengumpulan dataset guna kepentingan penelitian ini, dataset yang akan digunakan merupakan dataset baru yaitu menggunakan data nematoda dari Indonesia dengan bantuan dari Bagian Nematologi, Laboratorium Ilmu Hama Tumbuhan, Departemen Proteksi Tanaman, Fakultas Pertanian UGM. Dari dataset yang diberikan, yaitu berupa gambar satu sampai sekumpulan nematoda dalam satu gambar. Dataset masih belum termasuk bounding box dan masih berupa gambar mikroskopis. Berikut contoh dataset terdapat pada gambar 3.4 dan gambar 3.5 di bawah ini.



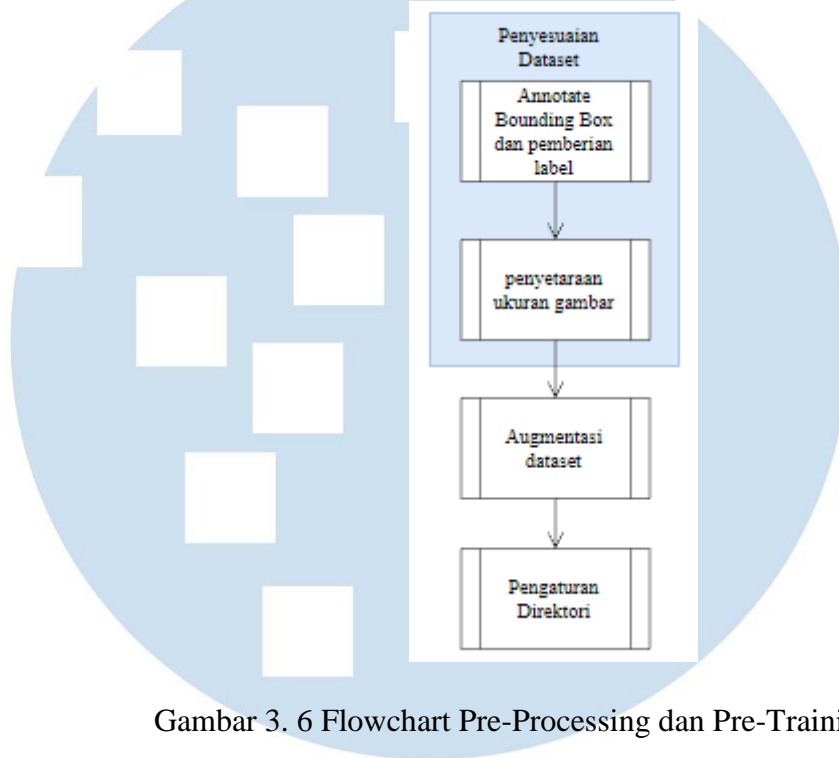
Gambar 3. 4 kiri parasit, kanan non parasit



Gambar 3. 5 dataset mix parasit dan non parasit

3.1.3 Pre-Processing atau Pre-Training

Pada proses pre-processing atau pre-training, dilakukan dengan Roboflow dengan beberapa tahap yaitu:



Gambar 3. 6 Flowchart Pre-Processing dan Pre-Training

3.1.3.1 Penyesuaian dataset

Dari dataset yang sudah ada, pertama akan dilakukan pemasangan bounding box dan labeling karena dataset masih berupa gambar yang belum diberi box dan label. Kemudian dilakukan penyesuaian terhadap parameter yang sesuai dan sudah ditentukan seperti dilakukan scaling atau penyetaraan ukuran menjadi 416x416. Berikut di bawah ini merupakan contoh annotate bounding box dan labeling terdapat pada gambar 3.7.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3. 7 Contoh Annotate Box dan Labeling [18]

Dari total 80 gambar data yang di anotasi terbagi menjadi 69 gambar data training dan 11 gambar data testing yang memiliki persebaran seperti pada tabel 3.1 di bawah ini

Tabel 3. 1 Persebaran Dataset Original

	parasit	non parasit
jumlah gambar Training	69	
jumlah gambar Testing	11	
jumlah label training	143	167
jumlah label testing	34	23

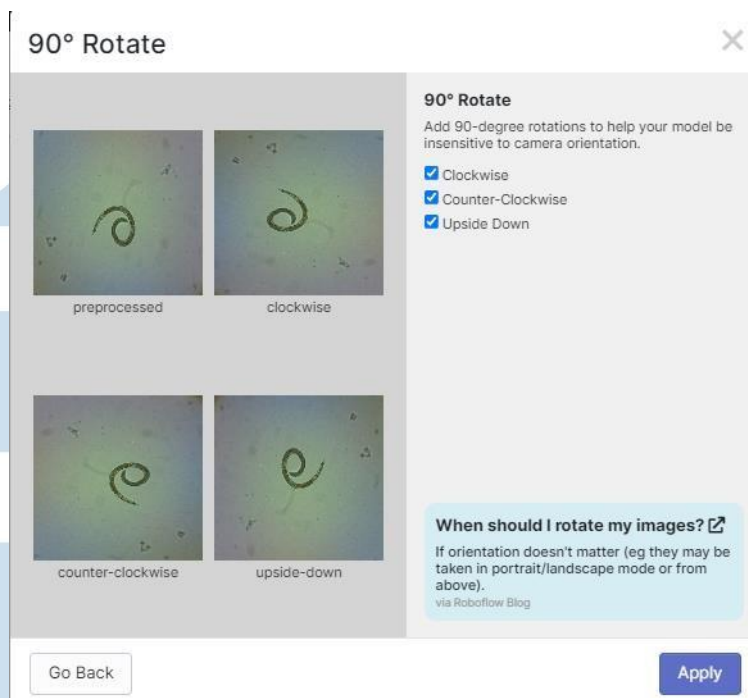
Seperti yang terlihat pada tabel diatas, data asli atau original yang digunakan dalam penelitian ini sebanyak 80 gambar dengan variasi setiap gambarnya. Dalam satu gambar bisa memiliki isi yang berbeda beda seperti objek parasit atau non parasit satuan, banyak objek parasit atau non parasit dalam satu gambar, dan adapula gambar yang berisi banyak objek parasit dan non parasit dalam satu gambar. Maka dari jumlah gambar sebanyak 80, memiliki box atau objek sebanyak 177 objek parasit (143 untuk training, 34 untuk testing) dan 190 objek non parasit (167 untuk training, 23 untuk testing).

3.1.3.2 Data Augmentation

Selain penyesuaian dataset, pada pre-processing atau pre-training juga akan dilakukan data augmentasi dengan metode tradisional atau geometrical data augmentation yaitu hanya melakukan transformasi dengan brightness, exposure, rotasi, flip, shear, dan lain-lain. Data augmentasi ini diperlukan untuk memperbanyak dataset dengan dataset yang sudah ada diperbanyak namun tidak menduplikasinya. Fungsi data augmentasi ini adalah agar hasil dari training lebih baik sehingga menghindari adanya overfitting.

Sama seperti step penyesuaian dataset, augmentasi tradisional ini juga dilakukan dengan Roboflow. Augmentasi dilakukan dengan mencoba masing-masing satu metode augmentasi dan diperbanyak sebanyak 3x. cara ini digunakan pada 5 metode augmentasi yaitu brightness, exposure, rotate, flip, dan shear secara masing-masing. Kelima metode augmentasi ini dipilih dengan pertimbangan augmentasi yang tidak mengubah bentuk yang akan mengubah karakteristik dari nematoda tersebut. Augmentasi dengan 5 metode ini dilakukan untuk melihat performa masing-masing metode augmentasi dan kemudian dibandingkan dan dipilih 3 metode augmentasi dengan hasil terbaik untuk dilakukan augmentasi campuran.

Augmentasi campuran ini dibuat melalui 3 tahap yaitu pertama dilakukan augmentasi dengan brightness yang diperbanyak sebanyak 3 kali. Kemudian diaugmentasi dengan rotate 90° yang juga sebanyak 3 kali. Dan terakhir dilakukan augmentasi dengan shear yang juga diperbanyak sebanyak 3 kali. Berikut merupakan contoh augmentasi dengan roboflow pada gambar 3.8 di bawah ini.



Gambar 3. 8 Contoh Augmentasi dengan Roboflow [18]

Berdasarkan persebaran dataset dari tabel 3.1 yang berisi data original atau data yang tidak diaugmentasi, dilakukan augmentasi yang mendapatkan hasil secara acak dengan menggunakan roboflow sehingga menghasilkan persebaran data seperti yang terdapat pada tabel 3.2 di bawah.

Dari total 1625 gambar data yang di anotasi dan telah dilakukan augmentasi terbagi menjadi 1415 gambar data training dan 210 gambar data testing yang memiliki persebaran seperti pada tabel 3.2 di bawah ini.

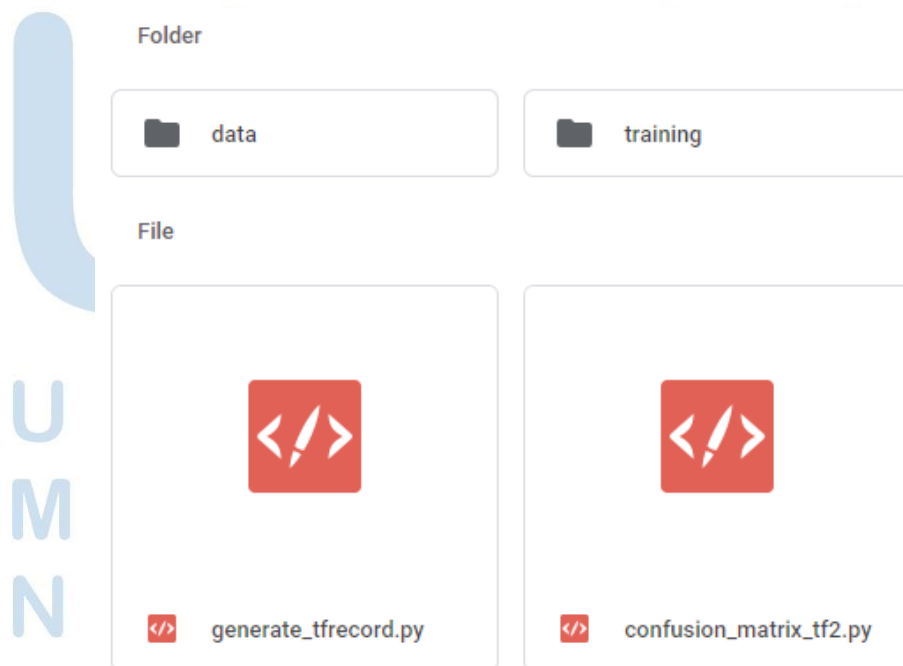
Tabel 3. 2 Persebaran Dataset Augmentasi

	parasit	non parasit
jumlah gambar training	1415	
jumlah gambar testing	210	
jumlah label training	2899	3412
jumlah label testing	651	321

Seperti yang terlihat pada tabel diatas, dataset yang diaugmentasi yang digunakan dalam penelitian ini sebanyak 1625 gambar dengan variasi setiap gambarnya. Dalam satu gambar bisa memiliki isi yang berbeda beda seperti objek parasit atau non parasit satuan, banyak objek parasit atau non parasit dalam satu gambar, dan adapula gambar yang berisi banyak objek parasit dan non parasit dalam satu gambar. Maka dari jumlah gambar sebanyak 1625, memiliki box atau objek sebanyak 3550 objek parasit (2899 untuk training, 651 untuk testing) dan 3733 objek non parasit (3412 untuk training, 321 untuk testing).

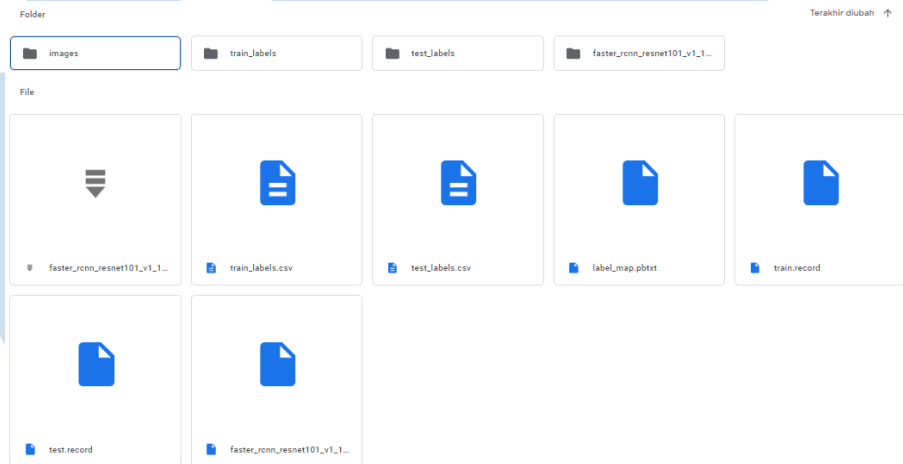
3.1.3.3 Susunan Direktori Drive

Dalam melakukan penelitian ini diperlukan penyimpanan pada google drive yang akan diakses oleh google colab. Secara keseluruhan, data akan berada pada sebuah folder yang didalamnya terdapat folder 'data', dan folder 'training'. Berikut merupakan gambar isi dari folder utama pada gambar 3.9 di bawah ini.



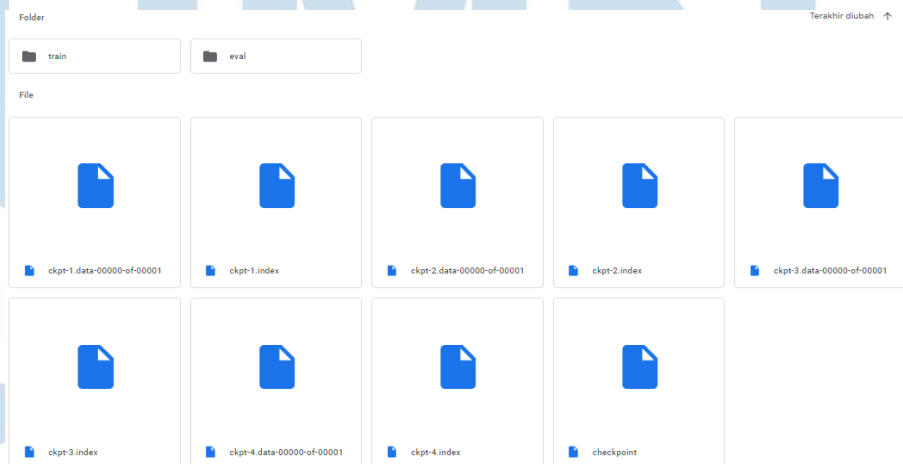
Gambar 3. 9 Isi dari Folder Utama

Pada folder ‘data’, berisi folder ‘images’, folder ‘train_labels’, folder ‘test_labels’. Selain folder-folder tersebut adapula beberapa file yang nantinya akan masuk ke folder ‘data’ ini seperti konfigurasi, label_map, file-file csv, dan file-file record. Berikut merupakan gambar isi dari folder data pada gambar 3.10 di bawah ini.



Gambar 3. 10 Isi dari Folder Data

Pada folder training akan terisi pada saat training dimulai yang terdiri dari folder ‘train’, folder ‘eval’ dan beberapa file checkpoint. Berikut merupakan gambar isi dari folder training pada gambar 3.11 di bawah ini.



Gambar 3. 11 Isi dari Folder Training

3.1.4 Training

Pada fase training ini, dataset yang digunakan adalah dataset yang berupa gambar dan lengkap beserta bounding box nya yang terdapat dalam bentuk file xml yang memiliki nama file sama seperti file gambar. File xml ini berisi koordinat box di setiap gambarnya.

Pada tahap training ini, dataset yang sudah ada nantinya akan dibagi berdasarkan objek, dari total 100% data original atau data yang tidak diaugmentasi (tabel 3.1) menjadi sekitar 80% untuk training yang diperbanyak dengan melakukan data augmentation, 20% untuk validasi dan untuk pengujian. Setelah dilakukan pembagian 80% dan 20% objek dari 80 gambar, maka akan dilakukan augmentasi berdasarkan pembagian dari data asli dengan mendapatkan hasil secara acak menggunakan roboflow.

Untuk porsi training dibuat lebih banyak karena pada metode ini diperlukan lebih banyak pelatihan untuk mencapai hasil uji dan validasi dengan akurasi yang lebih baik. Dataset akan dipisahkan lebih dulu untuk training, validasi, dan pengujian. Data original sebanyak 80 gambar.

Training akan dilakukan dengan Google Colaboratory karena berbentuk cloud sehingga dapat lebih mudah di jalankan menggunakan browser, dan mempermudah atau mengurangi proses instalasi dan setup karena keperluan seperti setting dan adjustment akan diatur atau diserahkan ke cloud, sehingga lebih mudah dan praktis digunakan dibanding dengan menggunakan Jupyter Notebook yang perlu instalasi [19].

Oleh karena training akan dilakukan dengan Google Colab, maka bahasa yang akan digunakan untuk proses training ini adalah bahasa pemrograman Python dengan framework deep learning yang digunakan adalah tensorflow.

3.1.5 Hasil, Validasi, Evaluasi, dan dokumentasi

Dari proses yang sudah dilakukan sebelumnya, mulai dari pengumpulan data, pre-processing, training, kemudian dilakukan pengujian dan validasi. Diharapkan hasil uji dan validasi ini mendapatkan hasil akurasi yang cukup baik (sekitar 70-80%). Selain menggunakan akurasi, penulis juga menggunakan average precision dan average recall sebagai hasil evaluasi dari detection. Pengujian dan validasi akan menggunakan dataset yang sudah dipisah sebelumnya diluar dataset yang di training. Hasil dan validasi ini didapatkan dari perbandingan antara hasil yang dikeluarkan sebagai output dari sistem dengan prediksi label asli dari nematoda atau data yang sudah ada. Hasil evaluasi untuk *average precision*, *mean average precision*, dan *average recall* dilihat dengan *coco metrics*. Sedangkan hasil akurasi akan ditampilkan dalam bentuk *confusion matrix* 3x3 yang kemudian akan dihitung secara manual. Contoh tabel *confusion matrix* seperti pada tabel 3.3 di bawah ini.

Tabel 3. 3 *Confusion Matrix*

	Actual	Prediction	Evaluation
TP	Non parasit	Non parasit	OK
FP	parasit	Non parasit	Not OK
FN	Non parasit	parasit	Not OK
TN	parasit	parasit	OK
GT P	Non parasit	-	Not OK
GT N	parasit	-	Not OK
FD P	-	Non parasit	Not OK
FD N	-	parasit	Not OK

Keterangan:

- TP = True positif yang artinya objek tersebut adalah objek positif dan dideteksi sebagai objek positif.

- FP = False positif yang artinya objek tersebut adalah objek negatif yang dideteksi sebagai objek positif.
- FN = False negatif yang artinya objek tersebut adalah objek positif yang dideteksi sebagai objek negatif.
- TN = True negatif yang artinya objek tersebut adalah objek negatif dan dideteksi sebagai objek negatif.
- GT P = Ground truth positif yang artinya objek tersebut adalah objek positif namun tidak terdeteksi.
- GT N = Ground truth negatif yang artinya objek tersebut adalah objek negatif namun tidak terdeteksi.
- FD P = False detection positif yang artinya objek tersebut tidak ada namun dideteksi sebagai objek positif.
- FD N = False detection negatif yang artinya objek tersebut tidak ada namun dideteksi sebagai objek negatif.

Menurut hasil yang sudah ada dengan akurasi yang didapatkan akan dilakukan evaluasi yang mengakibatkan pergantian parameter untuk trial mendapatkan hasil yang lebih baik. Setelah itu dilakukan dokumentasi berdasarkan trial dan error yang dilakukan dan sesuai tahap yang dilakukan.

