



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB 3 METODOLOGI PENELITIAN

Dalam penelitian penerapan yolov5 untuk mendeteksi jenis kendaraan dilakukan metode penelitian yang dibagi menjadi beberapa tahap yaitu, tinjauan pustaka, identifikasi masalah, pengumpulan data, perancangan dataset, pengujian dataset, dan implementasi sistem deteksi

3.1 Tinjauan Pustaka

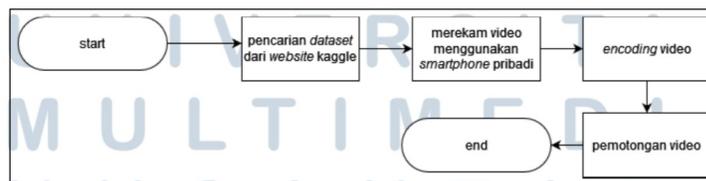
Pada tahap tinjauan pustaka yang dilakukan adalah mencari dan mempelajari penelitian serta teori yang berhubungan dengan deteksi jenis kendaraan menggunakan Yolov5.

3.2 Identifikasi Masalah

Pada tahap ini, dilakukan identifikasi masalah yang muncul sesuai data yang ada. Setelah itu, dilanjutkan dengan mencari solusi dari masalah tersebut. Masalah yang sudah dicarikan solusinya akan ditinjau kembali agar sesuai dengan hasil akhir yang diharapkan.

3.3 Pengumpulan data

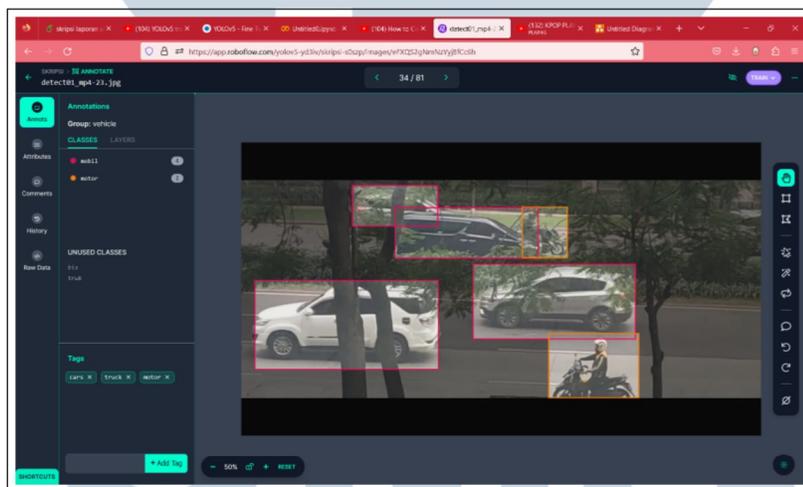
Pada tahap ini, setelah identifikasi masalah selesai, dimulai mencari dataset sesuai dengan penelitian ini, yaitu pencarian berupa dataset kendaraan dan jenis kendaraan pada *website* kaggle dan melakukan perekaman video di daerah Alam sutera dan perumahan puri dewata indah.



Gambar 3.1. Alur Pengumpulan data

Dalam Gambar 3.1 dilakukan pengumpulan data-data berupa video dan *install software* yang diperlukan. Data yang dikumpulkan berupa video dengan

format mp4, video kurang lebih sebanyak 12 video beserta 980 gambar. Setelah mencari video pada website kaggle dilanjutkan dengan perekaman menggunakan *smartphone* pribadi. setelah semua data didapamua tkan video tersebut akan di-*encoding* dikarenakan sistem *browser* Mozilla firefox tidak dapat menerima hasil video dengan *encode* H-265, maka dari itu dilakukanlah proses *encoding* pada video yang sudah dikumpulkan untuk keperluan skripsi ini. Setelah dilakukannya proses *encoding*, dilakukan proses pemotongan video agar saat nanti di *upload* ke *website* Roboflow tidak memakan waktu yang terlalu lama. Setelah semua video dilakukan proses *encoding* dan pemotongan video. Pada tahap perancangan *dataset* ini video yang telah melalui tahapan *encoding* dan pemotongan di-*upload* ke *website* Roboflow satu-persatu, lalu setelah di-*upload*, *website* Roboflow akan memproses video satu-persatu menjadi format.jpg dengan *settingan* 1frame/second. Setelah itu dilakukanlah anotasi gambar dengan tujuan untuk memberikan *boundary box* pada gambar dan nama kelas pada setiap masing-masing objek.



Gambar 3.2. Pemberian *Boundary box* pada objek dan nama kelas

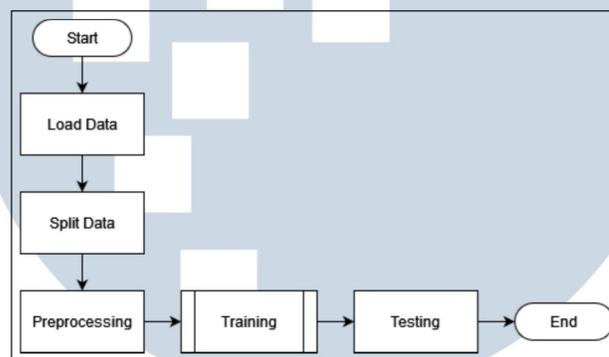
Setelah pemberian *Boundary box* proses selanjutnya adalah melakukan *train/split test* disini proses dibagi menjadi tiga bagian yaitu *training set*, *validation set*, dan *testing set*. Lalu dilanjutkan dengan *preprocessing data* disini dilakukan bagaimana hasil foto tersebut akan di implementasikan dengan banyak pilihan orientasi dan beberapa *filter* pada gambar yang dipilih adalah *auto-orient*, untuk *scaling-fit* dipilih 640x640. Pada tahap selanjutnya dilakukan *augmentation data*, disini penulis tidak banyak menambahkan pada *augmentation data*, hanya menampilkan *outputs per training* saja. setelah itu hasil akan di-generate dan

menunggu hasil untuk hasil skor mAP, *precision*, dan *recall*-nya.

3.4 Tahap Perancangan

Dalam tahap perancangan sistem deteksi untuk jenis kendaraan direalisasikan dalam bentuk *flowchart*.

3.4.1 Flowchart Utama



Gambar 3.3. *Flowchart* utama

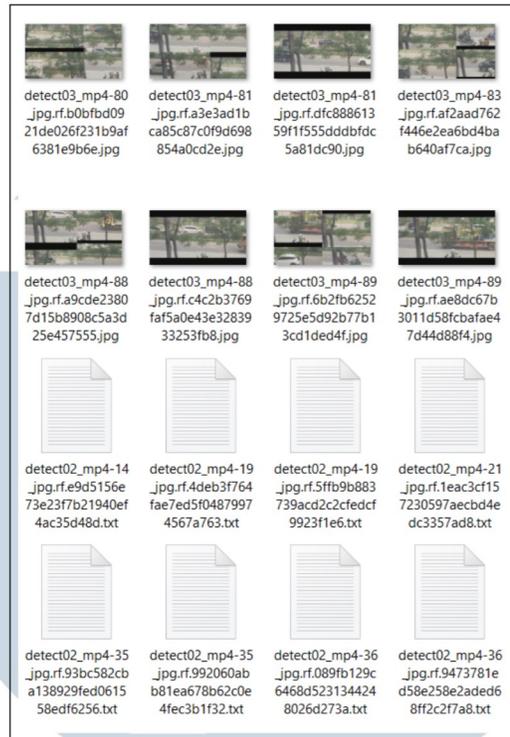
Pada Gambar 3.3 dilakukan *load data* berupa *file* dilakukan *load data* berupa *file* yang ada pada *device* pribadi berupa laptop dan *website roboflow* dimana kedua *data* tersebut sudah dilakukan anotasi dan pemberian label sebelumnya.

U
M
I
N

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 3.4. file yang digunakan

Pada tahap *load data* pada laptop penulis, yang dilakukan langsung mengeksportnya ke google colab, lalu sisa *data* yang ada pada *website roboflow* di-load ke google colab.

```

!pip install roboflow

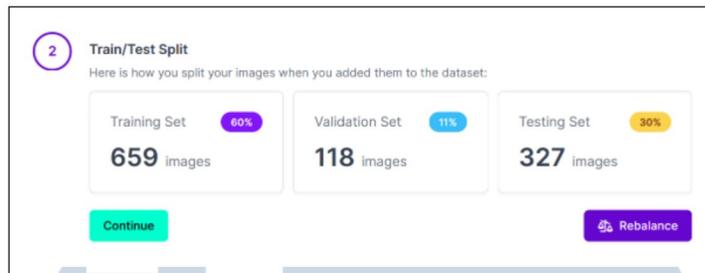
from roboflow import Roboflow
rf = Roboflow(api_key="PEDowgIKLwd9CTqY3Mqx")
project = rf.workspace("yolov5-yd3iv").project("skripsi-s0szp")
dataset = project.version(5).download("yolov5")

```

Gambar 3.5. code untuk load data pada website roboflow

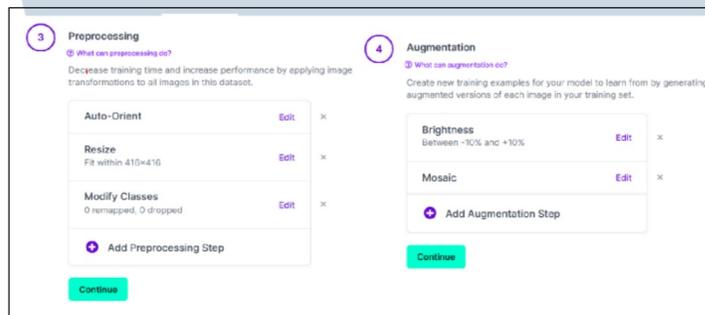
setelah dilakukan *load data* ke google colab dilanjutkan dengan *split data*





Gambar 3.6. *splitting data*

Pada tahap *splitting data* penulis membaginya menjadi 3 bagian yaitu *training set*, *validation set*, dan *testing set*. Setelah dilakukan *splitting data* dilakukan *preprocessing* dan *augmentation*.

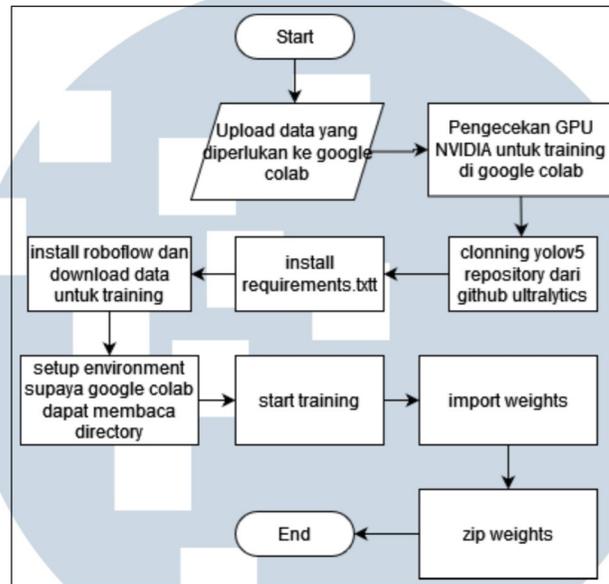


Gambar 3.7. *Preprocessing dan Augmentation*

Pada Gambar 3.7 dilakukan *preprocessing* dengan tujuan untuk merubah resolusi gambar dari ukuran 640x360 *pixel* menjadi 460x460 *pixel* dengan tujuan agar saat dilakukannya proses *training* gambar yang sudah di-*resize* akan lebih mudah men-*training*-nya dan proses *training*-nya akan lebih cepat dan tidak memakan waktu lama. Pada tahap *augmentation* penulis hanya merubah *brightness*-nya saja menjadi 10% supaya hasil yang digunakan untuk *training* lebih cerah dan dapat mendeteksi kendaraan lebih jelas.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.4.2 Training Flowchart



Gambar 3.8. Alur proses *training*

Pada Gambar 3.8 untuk proses *training* dimulai dari upload seluruh *data* yang diperlukan ke google colab, dilanjutkan dengan pengecekan *GPU* NVIDIA sudah tersedia pada google colab, tujuan pengecekan *GPU* adalah supaya saat melakukan *training* tidak memakan waktu yang lama. Dilanjutkan dengan melakukan *cloning repository* Yolov5 dari *github* ultralytics, lalu dilanjutkan dengan melakukan *install requirements.txt* karena dalam *requirements.txt* tersimpan seluruh *libraries*, *modules*, dan *packages* yang nantinya akan dipergunakan ketika mengerjakan sebuah projek. Selanjutnya yang dilakukan adalah *set up environment* dikarenakan google colab *data* yang sudah tersimpan didalamnya bersifat *temporary* atau sementara sehingga diperlukan *set up environment*. Dilanjutkan dengan memulai *training*, dikarenakan kita sudah melakukan anotasi dari awal dan pemberian *bounding box data* yang sudah tersplit nantinya akan menjadi *training set* dan langsung akan dijalankan dan akan menghasilkan *folder train* yang berisi *weights*, *result.xls*, hasil kurva *loss*, kurva *precision*, kurva *recall*, dan kurva *confidence F1-score*.

3.4.3 Testing

Pada bagian *testing* dilakukan dengan cara memanggil

```
1 %%time
2 !python detect.py --weights runs/train/exp/weights/best.pt --img
  416 --conf 0.5 --source {dataset.location}/test/images
```

Kode 3.1: potongan kode untuk *testing*

pada tahap *testing* seluruh data yang ada pada *folder test* dan *sub-folder images* akan dibaca dan diklasifikasikan berdasarkan *class* yang sudah ditentukan sebelumnya.

3.4.4 Perancangan Model

Dalam melakukan training, model YOLO yang akan digunakan adalah yolov5s. Model yolov5s yang akan digunakan dalam deteksi jenis kendaraan akan menggunakan *framework* PyTorch dengan *hyperparameter* bawaan (*default*), berikut adalah detailnya.

1. Image size: 416x416
2. Batch: 16
3. Epoch: 100
4. Learning rate: 0.01

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.4.5 Pengujian Dataset

Pada tahap ini, dilakukan pengujian terhadap hasil *split dataset* jenis kendaraan yang akan digunakan sebagai bahan penelitian ini. Dengan tujuan supaya hasil yang didapatkan akan sesuai dengan harapan dan meminimalisir tingkat *error* saat diaplikasikan pada saat penelitian. Pengujian hasil *split dataset* ini dilakukan pada *website* roboflow.



Gambar 3.9. contoh hasil *split dataset*

3.5 Implementasi

Pengimplementasian deteksi jenis kendaraan menggunakan bahasa pemrograman *python* dan digunakan pada gambar yang sudah di-*training* dan *testing* sehingga muncul nilai *confidence* dan *bounding box*-nya.

U M I N
UNIVERSITAS
MULTIMEDIA
NUSANTARA

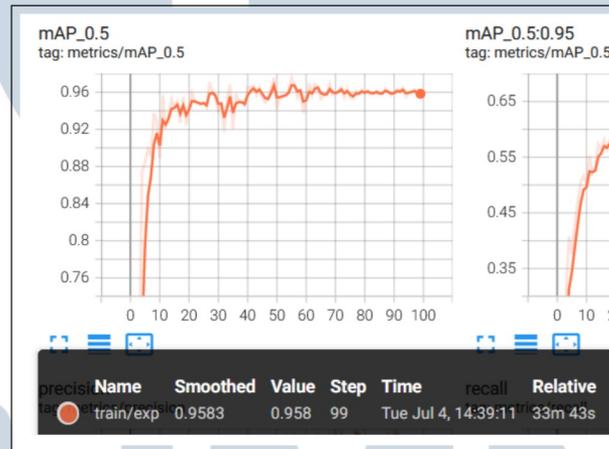
3.6 Evaluasi

Pada tahap evaluasi ini setelah dilakukannya *training* kita akan mendapatkan hasil berupa *metrics*

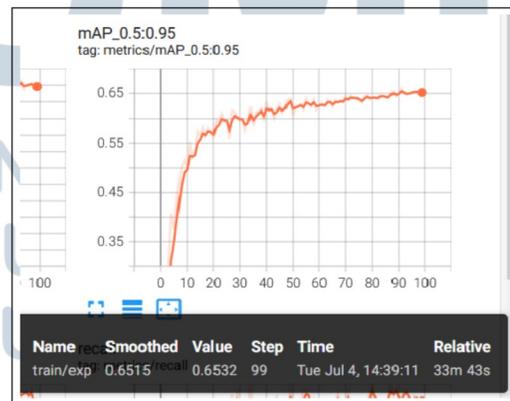
3.6.1 Metrics

A mAP Metrics

Setelah dilakukan *training* pada 100 kali pengulangan didapatkan kurva mAP@0.5 dan mAP@0.5:0.95. Berikut gambar ditampilkan pada Gambar 3.10 dan Gambar 3.11



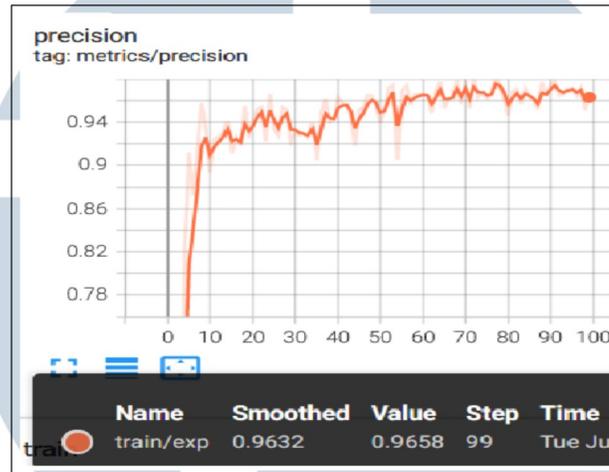
Gambar 3.10. mAP@0.5



Gambar 3.11. mAP@0.5:0.95

B Precision Metrics

Precision metrics didapatkan setelah kita men-*train dataset* kita. berikut gambar *precision metrics* ditunjukkan pada Gambar 3.12



Gambar 3.12. *precision metrics*

C Recall Metrics

Recall metrics didapatkan dari *weights* pada *best .pt* dan digunakan untuk membuat sebuah prediksi. *Recall metrics* ditunjukkan pada Gambar 3.13



Gambar 3.13. *Recall metrics*

D Accuracy

Accuracy didapatkan setelah kita melakukan uji coba sebanyak 5 kali yang nanti akan dijelaskan lebih detail pada bab IV.

3.7 Penulisan Laporan

3.7.1 Sistematika Penulisan

Berisikan uraian singkat mengenai struktur isi penulisan laporan penelitian, dimulai dari Pendahuluan hingga Simpulan dan Saran.

Sistematika penulisan laporan adalah sebagai berikut:

- **Bab 1 PENDAHULUAN**

Bab ini berisikan latar belakang masalah yang melandasi penelitian yang dikerjakan. Yang meliputi:

1. Latar belakang masalah
2. Rumusan masalah
3. Tujuan penelitian
4. Manfaat penelitian
5. Sistematika penulisan

- **Bab 2 LANDASAN TEORI**

Bab ini berisikan teori yang menjadi referensi dalam melakukan penelitian yang diperoleh dari beberapa sumber.

- **Bab 3 METODOLOGI PENELITIAN**

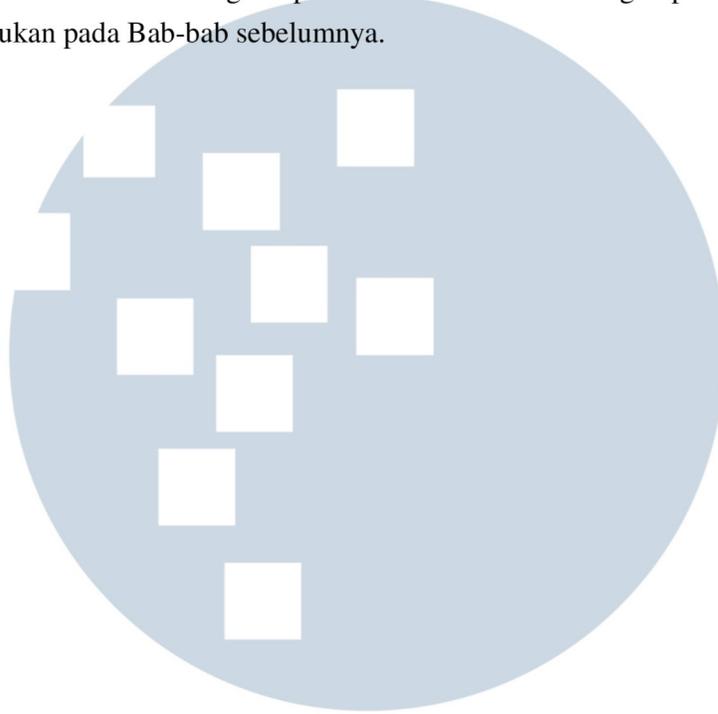
Bab ini berisikan langkah-langkah yang dilakukan selama penelitian agar mencapai titik akhir berupa solusi yang ingin dicapai dalam penelitian ini.

- **Bab 4 HASIL DAN DISKUSI**

Bab ini berisikan tentang hasil penelitian yang sudah dicapai beserta penjabarannya.

- Bab 5 SIMPULAN DAN SARAN

Bab ini berisikan tentang simpulan dan saran terkait dengan penelitian yang dilakukan pada Bab-bab sebelumnya.



UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA