

[ABDITEK][2][2023][42-52]

ISSN : 2964-5379 (Online)

Journal homepage: <http://ojs.uninus.ac.id/index.php/Abditek>



ABDITEK NUSANTARA

Jurnal Pengabdian Kepada Masyarakat
Fakultas Teknik UNINUS

ISSN : 2964-5379 (Online)

Journal homepage: <http://ojs.uninus.ac.id/index.php/Abditek>

PROTOTYPE SISTEM DETEKSI DAN PENGENALAN PLAT NOMOR SEPEDA MOTOR DI INDONESIA MENGGUNAKAN YOLOv4

Muhama Ikmal Wiawan¹

Fakultas Teknik, Universitas Islam Nusantara,
ikmal.wiawan@uninus.ac.id

Siti Nur²

Fakultas Teknik, Universitas Islam Nusantara,

wahyu satya gumelar³

Universitas Islam Nusantara

ABSTRAK

Perkembangan kendaraan bermotor di Indonesia setiap tahun semakin bertambah , dilansir dari (Badan Pusat Statistik, 2018) menyimpulkan bahwa jumlah sepeda motor yang terhitung hingga tahun 2018 di Indonesia mencapai 120 juta unit. plat nomor merupakan identitas utama dari sebuah kendaraan bermotor yang beroperasi di jalan raya , sehingga plat nomor umumnya diletakan pada tempat atau posisi yang mudah dilihat. (Mellolo, 2012). Proses pencatatan plat nomor menjadi sebuah urgensi tersendiri pada beberapa kondisi, seperti ketika proses penindakan pelanggaran lalu lintas yang dilakukan oleh polisi , ketika proses parkir untuk menjalankan manajemen proses parkir , juga proses pemantauan masuk dan keluarnya kendaraan ke dalam wilayah khusus yang umumnya dilakukan oleh pihak keamanan sebagai bagian dari standar keamanan wilayah. Pencatatan plat nomor tersebut umumnya dilakukan secara manual oleh manusia, hal tersebut sangat membutuhkan sumber daya manusia yang cukup banyak dan biaya yang cukup banyak untuk biaya jasa keamanann. Pendeteksial Plat nomor secara manual memerlukan resource yang besar dan tidak efisien sehingga kami mengusulkan pemanfaatan *Artificial Intelligence* atau kecerdasan buatan di bidang computer vision. Metode Pendeteksian Visual menggunakan YOLOv4 dan menggunakan algoritma *You Only Look Once* (YOLO) sebagai penerapan kecerdasan buatan.

Keywords: *Plat Nomor, Artificial Intelligence, YOLO, Computer Vision*

PENDAHULUAN

Kendaraan merupakan hal yang bisa sangat dibutuhkan oleh masyarakat dalam menjalani kegiatan sehari-hari, terutama untuk kegiatan mobilitas dari satu tempat ke tempat lainnya. Perkembangan kendaraan bermotor di Indonesia setiap tahun semakin bertambah, dilansir dari (Badan Pusat Statistik, 2018) menyimpulkan bahwa jumlah sepeda motor yang terhitung hingga tahun 2018 di Indonesia mencapai 120 juta unit.

Salah satu bagian yang cukup unik dari kendaraan bermotor adalah plat nomor polis kendaraan tersebut yang secara resmi disebut sebagai Tanda Nomor Kendaraan Bermotor (TNKB), plat nomor merupakan identitas utama dari sebuah kendaraan bermotor yang beroperasi di jalan raya, sehingga plat nomor umumnya diletakan pada tempat atau posisi yang mudah dilihat. (Mellolo, 2012)

Proses pencatatan plat nomor menjadi sebuah urgensi tersendiri pada beberapa kondisi, seperti ketika proses penindakan pelanggaran lalu lintas yang dilakukan oleh polisi, ketika proses parkir untuk menjalankan manajemen proses parkir, juga proses pemantauan masuk dan keluarnya kendaraan ke dalam wilayah khusus yang umumnya dilakukan oleh pihak keamanan sebagai bagian dari standar keamanan wilayah. Pencatatan plat nomor tersebut umumnya dilakukan secara manual oleh manusia, hal tersebut sangat membutuhkan sumber daya manusia yang cukup banyak dan biaya yang cukup banyak untuk biaya jasa keamanann.

Manusia berusaha untuk mengadopsi bentuk kecerdasan kedalam sistem komputer. Kecerdasan komputer inilah yang seringkali disebut Artificial Intelligence atau kecerdasan buatan, seperti mengenali sesuatu, membedakan objek satu dengan objek lainnya, bahkan mengenali karakter yang terdapat dalam citra.

Kecerdasan buatan tersebut dapat diterapkan kepada proses pencatatan plat nomor agar mengurangi jumlah sumber daya manusia yang terlibat dalam proses pencatatan plat nomor.

Salah satu metode penerapan kecerdasan buatan pada pencatatan plat nomor adalah sistem deteksi dan pengenalan plat nomor kendaraan menggunakan YOLOv4 sebagai judul yang Peneliti ambil. Sistem tersebut terfokus kepada plat nomor sepeda motor di wilayah Indonesia, dan menggunakan algoritma You Only Look Once (YOLO) sebagai penerapan kecerdasan buatan.

METODE PENELITIAN

Metode Penelitian ini adalah Sistematisa Peneliti dalam menjalankan Penelitian. Penelitian ini Menggunakan Metode Kualitatif Sehingga Hasil Penelitian dapat dibuktikan secara kualitatif dari mulai Proses hingga pengujian akhir. Metode Pengembangan Sistem Menggunakan Metode Waterfall yang terdiri dari

Analisis

Sistem yang dibuat merupakan sistem deteksi dan pengenalan plat nomor menggunakan YOLOv4 yang berfokus pada plat nomor sepeda motor di wilayah Indonesia. Selain itu, sistem ini pun dirancang untuk dapat mengenali setiap karakter pada plat nomor menggunakan tesseract-ocr sebagai fitur tambahan dari sistem deteksi dan pengenalan plat nomor sepeda motor di Indonesia . Sistem ini dapat mendeteksi dan mengenali plat nomor sepeda motor di Indonesia melalui gambar , juga melalui video secara realtime menggunakan webcam

Perencanaan

Perencanaan Merupakan tahapan Penentuan Rencana Detail dalam mengembangkan sebuah perangkat lunak. Di dalam perencanaan terdiri dari Rencana Pengembangan dan rencana Pengujian Prototipe. Perencanaan terdiri dari rencana implementasi dan rencana pengujian.

Rencana Implementasi

1. Pembuatan dataset dengan gambar yang dikumpulkan dari internet .
2. Implementasi training terhadap dataset untuk menjadi sebuah model deteksi objek.
3. Implementasi instalasi beberapa software pendukung.
4. Implementasi pengujian dengan gambar dan video secara realtime.

Rencana Pengujian

1. Aktivasi environment anaconda untuk menjalankan yolov4 dengan CPU.
2. Penyiapan model menjadi format model tensorflow untuk melakukan deteksi & pengenalan plat nomor.
3. Pengujian menggunakan gambar dan video secara realtime.

Implementasi

Tahap Implementasi adalah Proses eksekusi dari seluruh hasil analisa kebutuhan dan Perencanaan yang matang. Tahap pembuatan aplikasi ada di dalam proses ini

Pengujian

Pengujian merupakan tahapan akhir guna menguji hasil akhir penelitian yang diharapkan. Dalam pelaksanaan prototipe maka hasil akhir harus dapat menunjukkan keberhasilan alat / sistem / software dalam pendeteksian Plat nomor.

HASIL

Pembuatan Dataset Plat Nomor Sepeda Motor Indonesia

Pada Tahap ini , Peneliti menjelaskan pembuatan *dataset* untuk menjadi model dalam proses deteksi dan pengenalan plat nomor sepeda motor di Indonesia

a. Mengumpulkan Data Gambar Untuk Dataset

Peneliti mengumpulkan data gambar plat nomor sepeda motor di Indonesia dari internet (seperti google images, duckduckgo images, situs jual beli sepeda motor , dan sebagainya)

Peneliti mengumpulkan 883 gambar plat nomor sepeda motor indonesia . Sebagian besar merupakan plat bagian depan dikarenakan nantinya saat penerapan *real* di lapangan sebagai verifikasi parkir , bagian termudah untuk dideteksi oleh kamera adalah bagian depan.

b. Cloning *Labelimg*

```
Cloning into 'labelImg'...
remote: Enumerating objects: 18, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (14/14), done.
Receiving objects: 79% (1369/1718), 31.48 MiB | 7.00 KiB/s
```

c. Instalasi Paket Dan *Resource* Pendukung Untuk *Labelimg*

d. instalasi paket *pyqt5* dan paket *lxml* sebagai paket pendukung untuk menjalankan *LabelImg* Peneliti juga melakukan instalasi *resource*

e. Pembuatan Label

Pada tahap ini Peneliti membuat label untuk anotasi setiap gambar plat nomor sepeda motor di Indonesia dengan file text yang bernama *classes.txt* yang berisi nama label yaitu *plat_nomor*

f. Pembuatan Anotasi

Pembuatan anotasi pada setiap gambar yang telah diambil dari internet yang digunakan untuk *dataset* plat nomor sepeda motor di Indonesia.

Pembukaan tool *labelImg* untuk membuat anotasi dengan menggunakan perintah *python labelImg.py ../New\folder/ classes.txt*

```
ls
__init__.py      data/           labelImg.py*   MANIFEST.in
build-tools/    demo/           libs/          README.rst
classes.txt      HISTORY.rst     LICENSE        requirements
CONTRIBUTING.rst issue_template.md Makefile       resources/

Raysan Sanja@DESKTOP-JPESJ4T MINGW64 ~/TA/labelImg (master)
$ python labelImg.py ../New\ folder/ classes.txt
Warning: QT_DEVICE_PIXEL_RATIO is deprecated. Instead use:
  QT_AUTO_SCREEN_SCALE_FACTOR to enable platform plugin control
  actors.
  QT_SCREEN_SCALE_FACTORS to set per-screen DPI.
  QT_SCALE_FACTOR to set the application global scale factor.
```

Setelah menjalankan perintah tersebut , akan langsung terlihat *pop-up* membuka tool *LabelImg* dengan tampilan awal.

Untuk Pembuatan anotasi agar dapat digunakan pada algoritma YOLO, Peneliti mengubah format untuk anotasi menjadi YOLO

Pembuatan anotasi dengan membuat *Box* sebagai anotasi pada posisi plat nomor yang terdapat pada data gambar dengan memilih menu *RectBox* di posisi sebelah kiri jendela *LabelImg*

Setelah anotasi ditentukan , Selanjutnya adalah label pada anotasi yang telah dibuat dengan label yang sudah dibuat sebelumnya

Anotasi selesai dibuat dengan labelnya , maka data gambar yang telah diberi anotasi+label tampak

g. Hasil Anotasi

[ABDITEK][2][2023][42-52]

ISSN : 2964-5379 (Online)

Journal homepage: <http://ojs.uninus.ac.id/index.php/Abditek>

Setelah Pembuatan Anotasi pada setiap gambar yang dikumpulkan sebagai *dataset*, maka setiap gambar akan memiliki file text tersendiri dengan nama yang sama yang berisi kode anotasi yang telah dibuat.

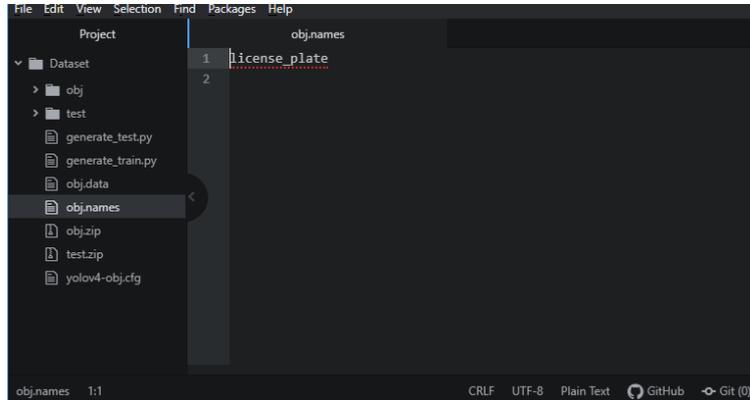
h. Melakukan Konfigurasi *Dataset* beserta *file-file* pendukung konfigurasi *Dataset* beserta beberapa *file-file* pendukung sehingga *dataset* siap untuk di *training*.

Peneliti memindahkan *dataset* gambar plat nomor sepeda motor di Indonesia beserta anotasinya ke dalam *file obj* sebanyak 80% dari total gambar(883) yaitu 707 gambar dan ke dalam *file test* sebanyak 20% dari total gambar(883) yaitu 176 gambar

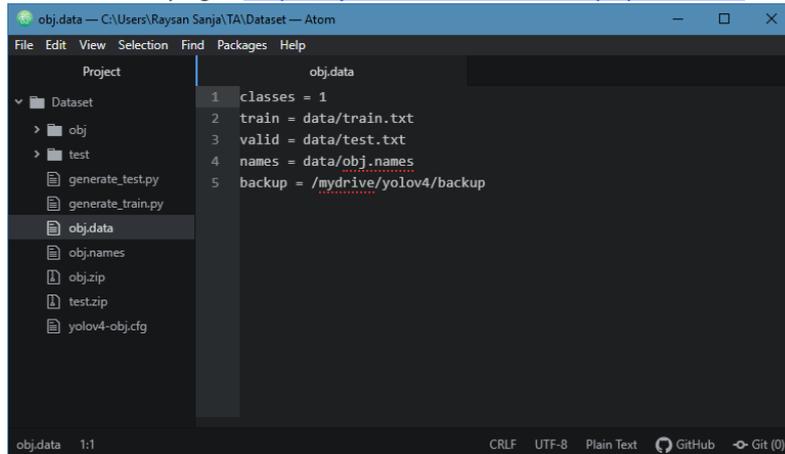
Peneliti mengkompres folder *obj* dan *test* menjadi format *.zip* seperti pada gambar 15 agar kualitas data tetap terjaga saat akan proses *training*.

Name	Date modified	Type	Size
obj	29/01/2021 00.01	File folder	
test	29/01/2021 00.01	File folder	
yolov4-obj.cfg	27/11/2020 16.46	CFG File	14 KB
obj.data	27/11/2020 16.46	DATA File	1 KB
obj.names	27/11/2020 16.46	NAMES File	1 KB
generate_test.py	27/11/2020 16.46	Python File	1 KB
generate_train.py	27/11/2020 16.46	Python File	1 KB
obj.zip	29/01/2021 00.15	WinRAR ZIP archive	102.519 KB
test.zip	29/01/2021 00.15	WinRAR ZIP archive	20.795 KB

konfigurasi file *obj.names* sebagai file *class* atau label dari objek yang akan dideteksi



Peneliti melakukan konfigurasi file *obj.data* yang berisi konfigurasi data-data objek yang akan dideteksi untuk melakukan proses *training*



Penjelasan filenya sebagai berikut :

- a. *classes* : menyatakan jumlah class pada *dataset* yang dibuat yaitu 1 (plat nomor)
- b. *train* : menunjukan direktori tempat disimpannya file *train.txt* yang merupakan file *dataset* untuk *training*
- c. *valid* : menunjukan direktori tempat disimpannya file *test.txt* yang merupakan file *dataset* untuk validasi
- d. *names* : menunjukan direktori tempat disimpannya file *obj.name* yang merupakan file *class* dari objek yang akan dideteksi
- e. *backup* : menunjukan direktori tempat disimpannya hasil *training*

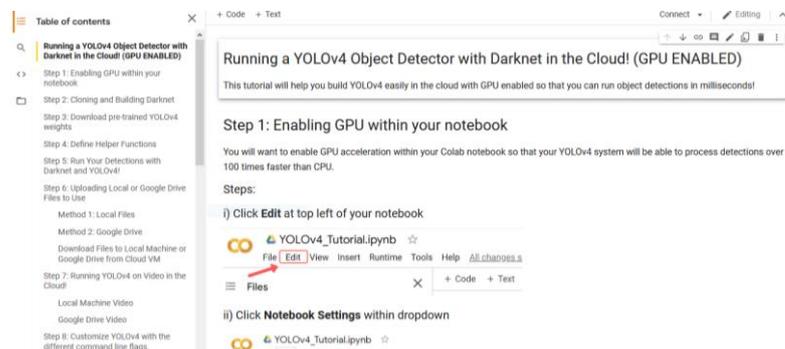
Training Dataset

Peneliti melakukan *training dataset* plat nomor sepeda motor Indonesia menggunakan *Google Colab* agar mendapat hasil yang maksimal dibanding dengan menggunakan perangkat komputer milik Peneliti.

- a. Membuka file *notebook Google Colab*

Pada tahap training , Peneliti menggunakan file *notebook Google Colab* dengan nama *YOLOv4_Training.ipynb* yang sudah siap untuk melakukan training. File tersebut Peneliti dapatkan dari video *youtube* dengan nama channel *TheAIGuys*(<https://www.youtube.com/channel/UCrydcKaojc44XnuXrfhIV8Q>).

Berikut merupakan tampilan awal *Google Colab* dalam kondisi membuka file untuk melakukan *training*.



b. Mengaktifkan GPU *Google Colab*

Peneliti mengaktifkan GPU untuk Notebook agar *training* pada *Google Colab* dilakukan menggunakan GPU.

Peneliti memilih menu edit > notebook setting untuk konfigurasi *notebook Google Colab* agar menggunakan GPU

Peneliti memilih GPU sebagai akselerator untuk berjalanya *notebook* pada *Google Colab* Koneksi *notebook* ke *cloud Google Colab*

Untuk dapat menggunakan GPU yang Peneliti konfigurasi kan dan menggunakan *reosurce* dari *Google* untuk melakukan proses *training* , Peneliti mengkoneksikan *notebook* dengan *Cloud Google Colab* dengan cara memilih/mengklik bagian “*connect*”

Setelah terkoneksi dengan *Cloud Google Colab* , maka bagian “*connect*” akan berubah

c. *cloning* dan *building darknet*

Peneliti melakukan *cloning Darknet* dari *repository AlexeyAB's* dengan menjalankan perintah *git clone* yang akan digunakan sebagai *framework* untuk YOLOv4

Peneliti melakukan perpindahan direktori kerja *notebook* pada *Google Colab* dan melakukan koinfigurasi agar *opencv* dan GPU aktif ketika mengeksekusi *darknet*

Peneliti melakukan verifikasi terhadap arsitektur CUDA pada GPU yang dijalankan pada *notebook*

Peneliti melakukan konfigurasi *darknet* dengan perintah *make* untuk membuat *darknet* dapat dieksekusi untuk menjalankan *training*

Menjalankan *helper function*(fungsi pembantu)

Peneliti menjalankan *helper function*(fungsi pembantu) *python*

d. Melakukan Konfigurasi *Drive*

Peneliti melakukan konfigurasi *drive* agar *notebook* terhubung dengan *google drive* milik Peneliti untuk memudahkan proses pemindahan *dataset* ke dalam *cloud VM(Virtual Machine) google colab*.

Peneliti melakukan *mounting google drive* milik Peneliti ke dalam *cloud VM(Virtual Machine) google colab* dengan menjalankan perintah *drive.mount*

Setelah dijalankan , Peneliti mengakses url yang diberikan oleh *google colab*, dan Peneliti diarahkan menuju halaman *login google* dan diperintahkan melakukan login menggunakan akun *google* milik Peneliti untuk mengizinkan *google colab* mengakses *google drive* milik Peneliti.

Untuk mengizinkannya , Peneliti menyalin kode autentikasi yang diberikan *google login*

Peneliti Meletakkan kode tersebut pada kolom yang terdapat pada *notebook google colab* untuk menyelesaikan proses *mounting google drive* milik Peneliti.

e. Melakukan pemindahan *dataset* dari komputer local ke *Google Drive*

Peneliti melakukan *upload dataset* data gambar plat nomor sepeda motor di Indonesia beserta dengan *file-file* pendukungnya

Peneliti melakukan upload terhadap file-file menuju *google drive* dengan nama folder *yolov4* Juga menambahkan folder *Archive* dan *Backup* untuk keperluan proses *training*.

obj.zip : file dataset yang akan di training

[ABDITEK][2][2023][42-52]

ISSN : 2964-5379 (Online)

Journal homepage: <http://ojs.uninus.ac.id/index.php/Abditek>

test.zip : file dataset yang akan dijadikan sebagai validasi hasil *training*

yolov4-obj.cfg : file konfigurasi dari yolov4 untuk *training*

obj.names : file yang berisi nama class atau label dari objek

obj.data : file yang berisi konfigurasi dataset untuk proses *training*

generate_test.py : file yang digunakan untuk melakukan *generate* terhadap *dataset* untuk validasi

generate_train.py : file yang digunakan untuk melakukan *generate* terhadap *dataset* untuk *training*

f. Melakukan Pemindehan *Dataset* dari *Google Drive* menuju *cloud VM Colab notebook* berjalan pada *cloud VM* , maka dari itu Peneliti perlu untuk memindahkan *dataset* yang telah diupload menuju *google drive* menuju *cloud VM* untuk memudahkan proses *training* karena *training* dilakukan pada *colab notebook*. Peneliti menyalin dataset menuju *cloud VM* dengan menjalankan perintah *cp* pada *colab notebook*

Karena *dataset* masih dalam format *zip* , Peneliti melakukan ekstraksi *zip*

g. Konfigurasi File untuk *Training*

Proses *training* memerlukan beberapa file agar dapat berjalan, dan file tersebut perlu dikonfigurasi agar sesuai dengan *dataset* yang akan di *training*.

1. Peneliti melakukan editing terhadap file *yolov4-obj.cfg* pada baris tertentu di dalam file tersebut agar sesuai dengan dataset yang ada.

Peneliti melakukan konfigurasi resolusi(*width, height*) model untuk deteksi plat nomor menggunakan resolusi standar yang ditetapkan yaitu 416,416 pada baris *width* dan *height* di dalam file *yolov4-obj.cfg*

Peneliti melakukan konfigurasi *class* yang akan di *training* yaitu 1 *class* yang bernama *plat_nomor* dengan mengedit baris *classes* menjadi "1" di dalam file *yolov4-obj.cfg*

```
1058 classes=1
```

2. Peneliti memindahkan file *obj.names* dan *obj.data* ke dalam folder yang bernama *data* pada *cloud VM*

```
[ ] # upload the obj.names and obj.data files t
!cp /mydrive/yolov4/obj.names ./data
!cp /mydrive/yolov4/obj.data ./data
```

3. Peneliti membuat data untuk proses *training* dan *validating* menggunakan program *python* yang sudah disediakan yang akan menghasilkan sebuah file *txt* bernama *train.txt* dan *test.txt* yang didalamnya terdapat lokasi setiap data gambar dari *dataset* yang akan di *training* dan *validating* menjadi sebuah model untuk melakukan deteksi objek plat nomor sepeda motor di Indonesia.

[ABDITEK][2][2023][42-52]

ISSN : 2964-5379 (Online)

Journal homepage: <http://ojs.uninus.ac.id/index.php/Abditek>

```
[22] # upload the generate_train.py and generate_test.py script to cloud
!cp /mydrive/yolov4/generate_train.py ./
!cp /mydrive/yolov4/generate_test.py ./
```

Now simply run both scripts to do th

```
[23] !python generate_train.py
!python generate_test.py
```

```
[24] # verify that the newly generated train.txt and test.txt can be seen
!ls data/
```

```
9k.tree      giraffe.jpg      labels          person.jpg
coco9k.map   goal.txt         obj             scream.jpg
coco.names   horses.jpg       obj.data       test
dog.jpg      imagenet.labels.list  obj.names      test.txt
eagle.jpg    imagenet.shortnames.list  openimages.names  train.txt
```

h. Melakukan *training dataset*

Peneliti melakukan proses *training* menggunakan *darknet* dengan perintah *darknet detector train* . Proses *training* tersebut membutuhkan waktu sekitar 2 hingga 3 jam lamanya hingga mendapat sebuah model untuk melakukan deteksi objek plat nomor sepeda motor di Indonesia.

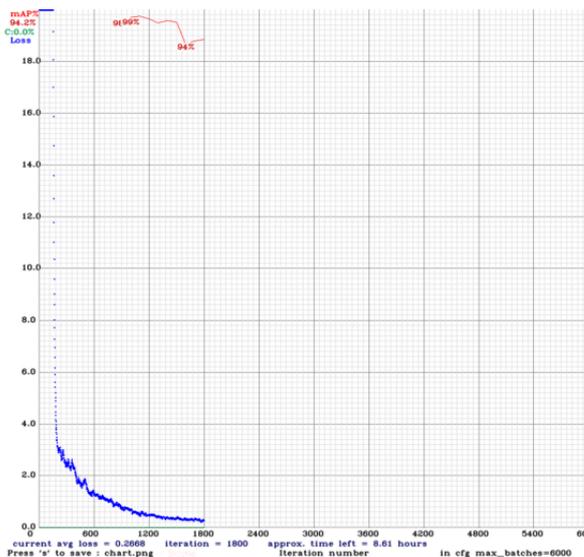
```
! # train your custom detector! (uncomment %capture below if you run into memory issues or yo
%capture
!./darknet detector train data/obj.data cfg/yolov4-obj.cfg yolov4.conv.137 -dont_show -map
```

i. Mengunduh *Model* dari *Google Drive*

Hasil dari training yang dilakukan adalah model *darknet* dengan format *.weights* yang secara otomatis tersimpan di *google drive* yang sudah diintegrasikan dengan *colab notebook*. Peneliti mengunduh model tersebut dari *google drive*

DISKUSI

Setelah proses training selesai untuk mengevaluasi hasil training.



Grafik menunjukkan 2 data yang cukup penting untuk melakukan evaluasi model, yaitu data *loss* dengan grafik berwarna biru dan *mAP* (*mean average precision*) dengan grafik berwarna merah.

Model yang di *train* mendapat nilai *loss* 0,2 dan nilai *map* 92,4 %.

loss merupakan nilai kegagalan dari data yang di *train*, ambang batas sebuah model dikatakan akurat jika nilai *loss* dibawah 2,0. Artinya model hasil *training* yang dilakukan Peneliti dapat disimpulkan model yang akurat.

Nilai *mAP* (*mean Average Precision*) menunjukkan tingkat akurasi deteksi objek dan letaknya. Nilai ini menjadi parameter utama untuk mengukur akurasi suatu model yang diaplikasikan dan seberapa bagus model tersebut untuk menangani objek tanpa ada kesalahan. Semakin tinggi nilai *mAP*, semakin tinggi pula akurasi yang didapat, namun sebagai konsekuensinya, kecepatan komputasi menjadi lebih rendah.

Dengan nilai *mAP* 92,4% , yang Peneliti peroleh dari model , dapat diartikan model tersebut mempunyai akurasi yang tinggi.

Intergrasi dengan YOLOv4

Setelah tahap instalasi selesai , Peneliti melakukan integrasi antara *tesseract* dengan YOLOv4 agar sistem OCR (*Optical Character Recognition*) dapat dijalankan dengan sistem deteksi objek dan pengenalan objek plat nomor sepeda motor di Indonesia.

Integrasi dilakukan dengan mengarahkan YOLOv4 kepada direktori aplikasi *tesseract* pada *source code* dalam file *core/utills.py*

KESIMPULAN

Berdasarkan pembahasan laporan yang penulis lakukan , penulis dapat menarik kesimpulan bahwa :

- a. Model yang dibuat dapat digunakan untuk mendeteksi dan mengenali objek plat nomor sepeda motor Indonesia serta mencatat karakternya dengan menggunakan dataset yang berisi 883 gambar , mendapat nilai *loss* 0,2 dan nilai *map* 92,4 % .
- b. Sistem deteksi dan pengenalan plat nomor sepeda motor Indonesia berhasil dijalankan yang dibuktikan pada hasil pengujian dengan rata-rata jumlah keberhasilan yaitu 11 dari 12 kali percobaan dalam setiap pengujian dengan persentasinya sebesar 87,8%.
- c. Sistem pengenalan karakter plat nomor sepeda motor Indonesia berhasil juga dijalankan yang dibuktikan pada hasil pengujian dengan rata-rata jumlah keberhasilannya yang dihitung dari keseluruhan percobaan dari setiap pengujian yaitu 31 karakter yang dapat dikenali dari 78 karakter keseluruhan dengan persentasinya sebesar 38,3%.

DAFTAR PUSTAKA

aditya.yanuar.r. (2018, June). *Pengenalan Deep Learning*. Diambil kembali dari Universitas Gadjah Mada Menara Ilmu Machine Learning: <https://machinelearning.mipa.ugm.ac.id/2018/06/10/pengenalan-deep-learning/>

[ABDITEK][2][2023][42-52]

ISSN : 2964-5379 (Online)

Journal homepage: <http://ojs.uninus.ac.id/index.php/Abditek>

Ahmad, A. (2017). Mengenal Artificial Intelligence, Machine Learning, Neural Network, dan Deep Learning.

Alfarisi, H. M. (2020, March 23). *Darknet atau Darkflow, Dua Framework Algoritma Deep Learning YOLO (You Only Look Once)*. Diambil kembali dari Medium Article: <https://haiqalmuhamadalfarisi.medium.com/darknet-atau-darkflow-dua-framework-algoritma-deep-learning-yolo-you-only-look-once-3fb3552f8963>

Alfarisi, H. M. (2020, Maret 22). *Mengenal Perbedaan Artificial Intelligence, Machine Learning, Neural Network & Deep Learning (Part III)*. Diambil kembali dari <https://haiqalmuhamadalfarisi.medium.com/mengenal-perbedaan-artificial-intelligence-machine-learning-neural-network-deep-learning-part-bef034a145bd>

Andy. (2020, May 30). *Mengenal Apa Itu GitHub dan Cara Menggunakannya*. Diambil kembali dari qwords blog: <https://qwords.com/blog/apa-itu-github/>

Badan Pusat Statistik. (2018). *Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis, 1949-2018*. Diambil kembali dari www.bps.go.id

Dewi, S. R. (2019). DEEP LEARNING OBJECT DETECTION PADA VIDEO MENGGUNAKAN TENSORFLOW DAN CONVOLUTIONAL NEURAL NETWORK. Yogyakarta.