

# **Deteksi Objek Jenis Kendaraan Menggunakan Metode YOLO**

*Vehicle Type Object Detection Using YOLO Method*

## **PROPOSAL PROYEK AKHIR**

**Diajukan sebagai syarat untuk mengambil Mata Kuliah Proyek Akhir**

oleh :

**ROHMAN PRASETYO**

**6705184116**



**D3 TEKNOLOGI TELEKOMUNIKASI**

**FAKULTAS ILMU TERAPAN**

**UNIVERSITAS TELKOM**

**2021**

## LEMBAR PENGESAHAN

Proposal Proyek Akhir dengan judul :

Deteksi Objek Jenis Kendaraan Menggunakan Metode YOLO

*Vehicle Type Object Detection Using YOLO Method*

oleh :

ROHMAN PRASETYO

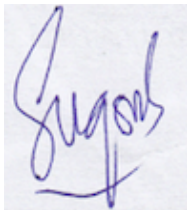
6705184116

Telah diperiksa dan disetujui untuk diajukan sebagai syarat mengambil  
Mata Kuliah Proyek Akhir  
pada Program Studi D3 Teknologi Telekomunikasi Universitas Telkom

Bandung, 8 Maret 2021

Menyetujui,

Pembimbing I



Sugondo Hadiyoso, S.T., M.T.

NIP. 13870076

Pembimbing II



Yuli Sun Hariyani, S.T., M.T.

NIP. 14880049

## ABSTRAK

Pertumbuhan jumlah kendaraan di Negara Indonesia sangat pesat dan hal tersebut menyebabkan timbulnya banyak masalah, diantaranya adalah masalah sistem keamanan parkir dan kontrol akses kendaraan di jalan raya. Dengan berkembangnya teknologi, dapat dideteksi secara otomatis oleh sistem dengan bantuan *Digital Image Processing* dan *Artificial Neural Network*. Baru baru ini penerapan metode *deep learning* banyak diusulkan peneliti untuk pemantauan arus lalu lintas dengan berbagai metode pendekatan seperti Faster R-CNN, SPP(Spatial Pyramid Pooling), SSD(Single Shoot multibox Detector ), R-FCN, You Only Look Once (YOLO), YOLOv2 dan YOLOv3.

Pada penelitian ini, metode *You Only Look Once* (YOLO) untuk mendeteksi objek jenis kendaraan dalam gambar. Dengan mengumpulkan beberapa *image* jenis kendaraan kemudian di proses menggunakan sistem *deep learning* dengan YOLO kemudian hasil outputannya akan menampilkan jenis kendaraan secara otomatis. Pada penelitian ini, sistem diharapkan dapat membantu dalam memberikan pertimbangan yang diperlukan dan berguna bagi pihak yang membutuhkannya, terutama untuk klasifikasi jenis kendaraan.

Kata kunci : *Digital Image Processing, Artificial Neural Network, deep learning, You Only Look Once.*

# DAFTAR ISI

LEMBAR PENGESAHAN .....	i
ABSTRAK .....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	iv
DAFTAR TABEL .....	v
BAB I.....	6
PENDAHULUAN .....	6
1.1 Latar Belakang .....	6
1.2 Tujuan dan Manfaat.....	7
1.3 Rumusan Masalah .....	7
1.4 Batasan Masalah.....	7
1.5 Metodologi.....	7
BAB II .....	8
DASAR TEORI.....	8
2.1 Kecerdasan Buatan ( <i>Artificial Intellegence</i> ) .....	8
2.2 Deep Learning .....	8
2.3 YOLO Object Detection .....	9
2.3.1 <i>Convolution Layer</i> .....	10
2.3.2 <i>Pooling Layer</i> .....	11
2.3.3 <i>Fully Connected Layer (FC Layer)</i> .....	11
BAB III.....	12
MODEL SISTEM .....	12
3.1 Blok Diagram Sistem .....	12
3.2 Tahapan Perencanaan.....	14
3.3 Perancangan.....	15
BAB IV .....	16
BENTUK KELUARAN YANG DIHARAPKAN.....	16
4.1 Keluaran Yang Diharapkan .....	16
4.2 Jadwal Pelaksanaan.....	16
DAFTAR PUSTAKA.....	17

## **DAFTAR GAMBAR**

Gambar 2. 3 Klasifikasi YOLO Object Detection .....	9
Gambar 2. 4 Jaringan Saraf Convolutional .....	10
Gambar 2.3.2 Proses Pooling Layer dengan cara Max Pool .....	11
Gambar 3.1 Model Sistem Deteksi Objek Jenis Kendaraan menggunakan Metode YOLO .....	12
Gambar 3.3 Diagram alir sistem Deteksi Objek Jenis Kendaraan Menggunakan Metode YOLO .....	15

## DAFTAR TABEL

Tabel 4.1 Jadwal Pelaksanaan.....	16
-----------------------------------	----

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Arus lalu lintas dan transportasi pada saat ini tengah menjadi perbincangan hangat ditengah masyarakat ilmiah maupun politik. Pertumbuhan jumlah kendaraan di Negara Indonesia sangat pesat dan hal tersebut menyebabkan timbulnya banyak masalah, diantaranya adalah masalah sistem keamanan parkir dan kontrol akses kendaraan di jalan raya. Untuk pemantauan pengguna lalu lintas dengan mengekstraksi informasi dari gambar seperti kecepatan, komposisi lalu lintas, kemacetan lalu lintas, bentuk kendaraan, jenis kendaraan, nomor identifikasi kendaraan, dan terjadinya pelanggaran lalu lintas atau kecelakaan di jalan. Baru baru ini penerapan metode *deep learning* banyak diusulkan peneliti untuk pemantauan arus lalu lintas dengan berbagai metode pendekatan seperti Faster R-CNN, SPP(Spatial Pyramid Pooling), SSD(Single Shoot multibox Detector ), R-FCN, You Only Look Once (YOLO), YOLOv2 dan YOLOv3.

Metode You Only Look Once (YOLO) merupakan salah satu metode yang paling cepat dan akurat pada pendeteksian objek bahkan mampu melebihi hingga 2 kali kemampuan algoritma lain. Dalam penelitian ini, dengan meningkatkan YOLOv2 model yang disebut YOLOv3\_Vehicle diusulkan untuk mendeteksi kendaraan. Dalam penelitian ini, penulis mengajukan suatu metode, yaitu metode YOLOv3 karena dinilai lebih baik dari versi sebelumnya dimana mempunyai akurasi yang lebih tinggi. Dengan metode yang digunakan diharapkan dapat bermanfaat untuk pendeteksian kendaraan dalam pemantauan arus lalu lintas dengan dataset yang dikumpulkan.

Berdasarkan uraian diatas penerapan sistem cerdas yaitu object detection dengan menggunakan YOLO dapat menjadi solusi untuk mengklasifikasi setiap kendaraan yang berada di jalan raya setiap harinya. Data klasifikasi kendaraan yang didapat bisa digunakan untuk pertimbangan menentukan pengawasan atau penertiban secara berkala di jalan raya. Oleh karena itu penulis membuat proyek akhir dengan judul “Deteksi Objek Jenis Kendaraan Menggunakan Metode YOLO”. Dengan sistem yang dibuat diharapkan dapat membantu

dalam memberikan pertimbangan yang diperlukan dan berguna bagi pihak yang membutuhkannya, terutama untuk klasifikasi setiap kendaraan.

## **1.2 Tujuan dan Manfaat**

Adapun tujuan dari Proyek Akhir ini adalah dapat menghasilkan sebuah sistem cerdas yang dapat mempermudah pengguna dalam menindak lanjuti pelanggaran dan mengenali jenis kendaraan secara otomatis.

## **1.3 Rumusan Masalah**

Adapun rumusan masalah dari Proyek Akhir ini, sebagai berikut:

1. Bagaimana perancangan deteksi objek jenis kendaraan dengan metode YOLO?
2. Bagaimana cara kerja deteksi objek jenis kendaraan dengan metode YOLO?
3. Bagaimana hasil dari deteksi objek jenis kendaraan dengan metode YOLO?

## **1.4 Batasan Masalah**

Dalam Proyek Akhir ini, dilakukan pembatasan masalah sebagai berikut:

1. Sistem hanya bisa digunakan pada intensitas cahaya yang cukup.
2. Sistem deteksi jenis objek kendaraan menggunakan metode YOLO.

## **1.5 Metodologi**

Metodologi pada penelitian ini, sebagai berikut:

1. Melakukan pencarian dan pengumpulan informasi yang berhubungan dengan tugas akhir melalui sumber jurnal, *internet*, dan buku referensi.
2. Melakukan persiapan terhadap perancangan yang akan dilakukan.
3. Melakukan perancangan sistem deteksi objek jenis kendaraan secara otomatis berdasarkan referensi yang didapatkan dari berbagai studi literatur.
4. Melakukan simulasi terhadap sistem deteksi objek jenis kendaraan yang sudah dibuat.



## **BAB II**

### **DASAR TEORI**

#### **2.1 Kecerdasan Buatan ( *Artificial Intelligence* )**

Kecerdasan buatan merupakan suatu cabang dalam bidang komputer sains yang membahas bagaimana sebuah komputer dapat meniru cara berpikir seperti manusia. Dengan kecerdasan buatan komputer dapat mengambil kesimpulan dan memutuskan suatu permasalahan seperti layaknya manusia. Menurut Luckin dan Colen dalam jurnal milik Ahmad Sudi Pratikno (2017) menyatakan bahwa Artificial Intelligence sebagai suatu sistem komputer yang dirancang untuk berinteraksi dengan dunia melalui kemampuan-kemampuan tertentu dan perilaku intelijen yang kita sadari seperti manusia pada umumnya. Hal tersebut dikuatkan oleh Colen yang menyatakan bahwa Artificial Intelligence merupakan teori yang mendasari tentang mekanisme suatu kecerdasan serta metode empirik untuk membangun dan menguji kemungkinan-kemungkinan model dalam mendukung suatu teori.

#### **2.2 Deep Learning**

*Deep learning* adalah salah satu jenis algoritma jaringan saraf tiruan yang menggunakan metadata sebagai input dan mengolahnya menggunakan sejumlah lapisan tersembunyi (*hidden layer*) transformasi non linear dari data masukan untuk menghitung nilai output. Algoritma pada *deep learning* memiliki fitur yang unik yaitu sebuah fitur yang mampu mengekstraksi secara otomatis. Hal ini berarti algoritma yang dimilikinya secara otomatis dapat menangkap fitur yang relevan sebagai keperluan dalam pemecahan suatu masalah. Algoritma semacam ini sangat penting dalam sebuah kecerdasan buatan karena mampu mengurangi beban pemrograman dalam memilih fitur yang eksplisit. Dan, algoritma ini dapat digunakan untuk memecahkan permasalahan yang perlu pengawasan (*supervised*), tanpa pengawasan (*unsupervised*), dan semi terawasi (*semi supervised*).

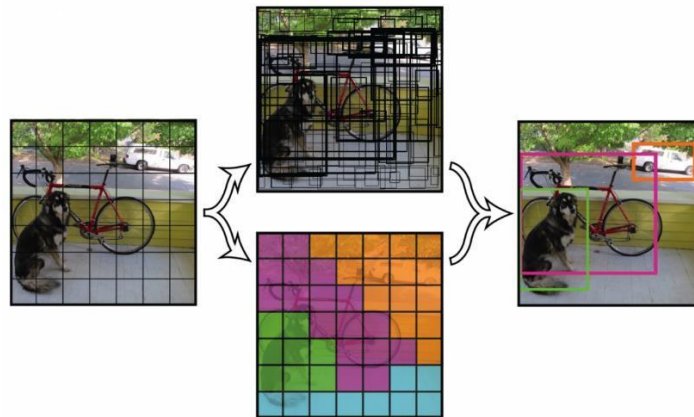
Jenis *Deep Learning* :

- *Deep Learning* untuk pembelajaran tanpa pengawasan (*Unsupervised Learning*): tipe ini digunakan pada saat tabel variabel target tidak tersedia dan korelasi nilai yang lebih tinggi harus dihitung dari unit yang diamati untuk menganalisa polanya.

- *Hybrid Deep Networks (Deep Learning Gabungan)*: pendekatan tipe ini bertujuan agar dapat dicapai hasil yang baik dengan menggunakan pembelajaran yang diawasi untuk melakukan Analisa pola atau dapat juga dengan menggunakan pembelajaran tanpa pengawasan.

## 2.3 YOLO Object Detection

YOLO (*you only look once*) merupakan algoritma *real object detection* yang baru-baru ini sangat populer untuk dikembangkan. YOLO menggunakan pendekatan yang sangat berbeda dengan algoritma sebelumnya, yakni menerapkan jaringan syaraf tunggal pada keseluruhan gambar. Pendeteksian objek dilakukan dengan membingkai objek yang akan dideteksi sebagai *regression problem* dan memisahkan special pada *bounding boxes* dan *class probabilities*. Dengan menggunakan *single neural network* untuk memprediksi *bounding boxes* dan *class probabilities* dari seluruh gambar pada satu kali evaluasi. Karena metode ini menggunakan *single neural network* untuk semua *detection pipeline*, maka performa deteksi ini bisa dioptimasi dari *end-to-end* (Redmon, Diyyala, Girshick, dan Farhadi, 2015).



Gambar 2. 3 Klasifikasi YOLO Object Detection

(Sumber: <https://machinelearning.mipa.ugm.ac.id/2018/08/05/yolo-you-only-look-once/>)

YOLO memiliki kemampuan pada base model yang bisa memproses *45 frame per second* secara *real time* pada versi yang lebih kecil bisa memproses *155 frame per second* secara *real-time*. YOLO juga memiliki arsitektur yang sederhana yaitu jaringan syaraf convolutional.

Layer	kernel	stride	output shape
Input			(416, 416, 3)
Convolution	3x3	1	(416, 416, 16)
MaxPooling	2x2	2	(208, 208, 16)
Convolution	3x3	1	(208, 208, 32)
MaxPooling	2x2	2	(104, 104, 32)
Convolution	3x3	1	(104, 104, 64)
MaxPooling	2x2	2	(52, 52, 64)
Convolution	3x3	1	(52, 52, 128)
MaxPooling	2x2	2	(26, 26, 128)
Convolution	3x3	1	(26, 26, 256)
MaxPooling	2x2	2	(13, 13, 256)
Convolution	3x3	1	(13, 13, 512)
MaxPooling	2x2	1	(13, 13, 512)
Convolution	3x3	1	(13, 13, 1024)
Convolution	3x3	1	(13, 13, 1024)
Convolution	1x1	1	(13, 13, 125)

Gambar 2. 4 Jaringan Saraf Convolutional

(Sumber: <https://machinelearning.mipa.ugm.ac.id/2018/08/05/yolo-you-only-look-once/>)

Jaringan saraf ini hanya menggunakan jenis lapisan standar; konvolusi dengan kernel 3 x 3 dan *max pooling* dengan 2 x 2 kernel. Lapisan konvolusional terkahir 1 x 1 kernel digunakan untuk mengecilkan data ke bentuk 13 x 13 x 125. 13 x 13 ini seharusnya terlibat familiar: itu adalah ukuran *grid* yang dibagi menjadigambar. 125 merupakan *channel* untuk setiap *grid*. 125 ini berisi data untuk kotak pembatas dan prediksi kelas. Setiap sel *grid* memprediksi 5 kotak sekeliling dan dijelaskan oleh 25 elemen data.

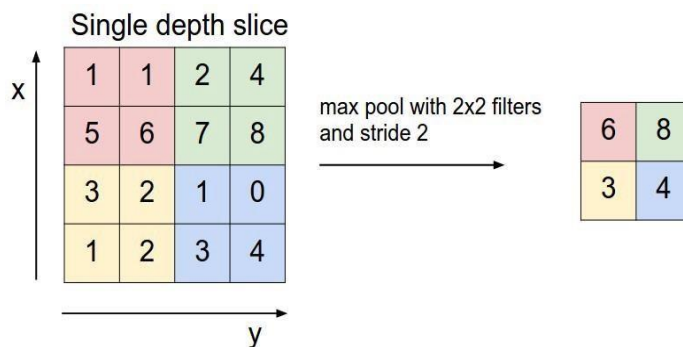
- X, Y untuk lebar dan tinggi kotak pembatas (dua elemen data)
- Skor keyakinan (satu elemem data)
- Distribusi probabilitas yang lebih dari 20 kelas (20 elemen data)

Berikut ini urutan rumus atau cara yang digunakan untuk bisa menerapkan metode YOLO *object detection*.

### 2.3.1 Convolution Layer

Pada *Convolution Layer* melakukan operasi konvolusi pada output dari layer sebelumnya. Konvolusi pada data citra bertujuan untuk mengekstraksi fitur dari citra input. Konvolusi akan menghasilkan transformasi linear dari data input sesuai informasi spasial pada data. Bobot pada layer menspesifikasikan kernel konvolusi yang digunakan, sehingga kernel konvolusi dapat dilatih berdasarkan input pada CNN.

### 2.3.2 Pooling Layer



Gambar 2.3.2 Proses Pooling Layer dengan cara Max Pool

(Sumber: <https://medium.com/@16611110/apa-itu-convolutional-neural-network-836f70b193a4>)

*Pooling layer* biasanya berbeda setelah convolutional layer. Pada prinsipnya *pooling layer* terdiri dari sebuah *filter* dengan ukuran dan *stride* tertentu yang akan bergeser pada seluruh area *feature map*. Dengan tujuan adalah mengurangi dimensi dari *feature map* (*Downsampling*), sehingga mempercepat komputasi karena parameter yang harus diperbarui semakin sedikit dan mengatasi *overfitting*.

### 2.3.3 Fully Connected Layer (FC Layer)

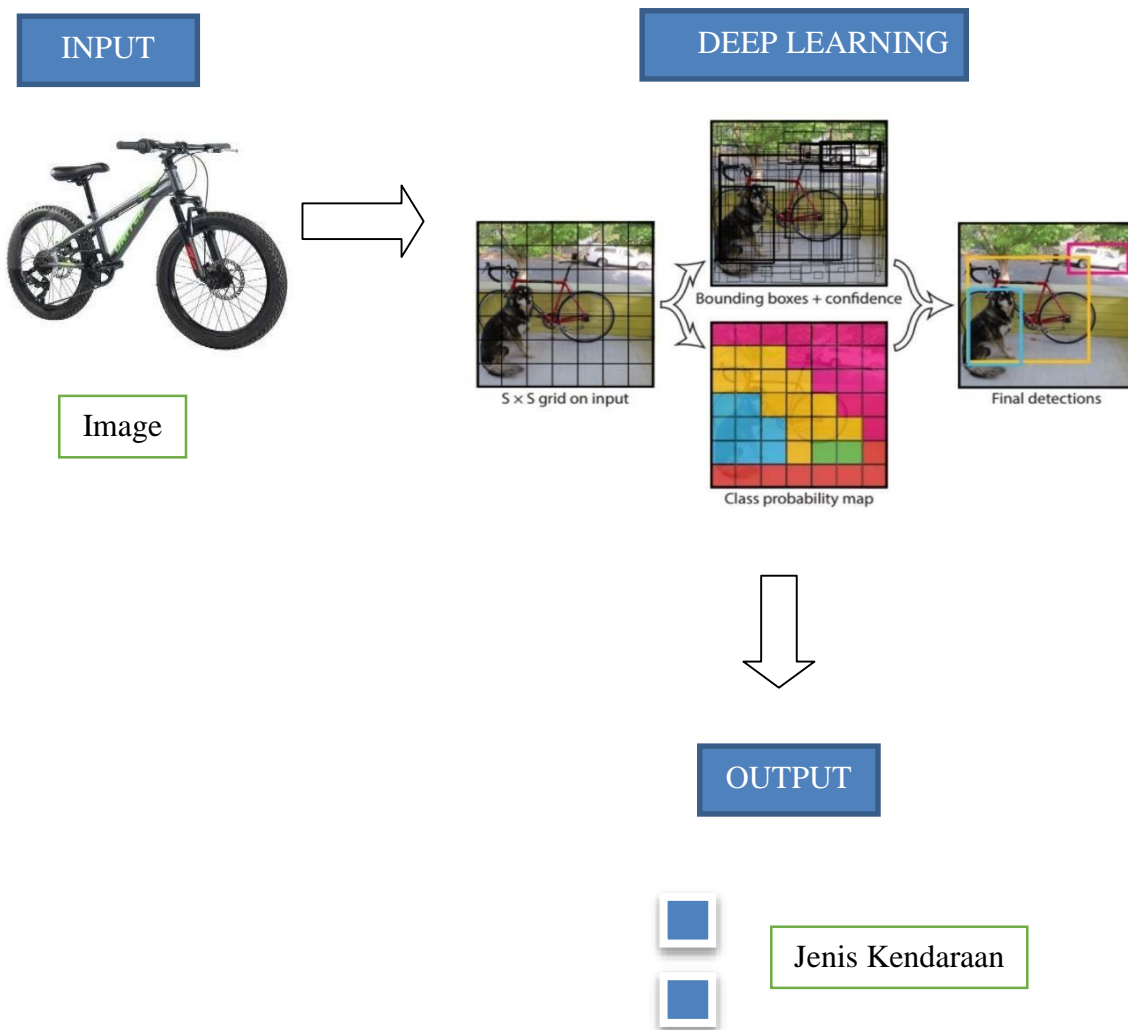
Pada *Fully connected layer* adalah syaraf tiruan yang dimana dari *input* data hingga *output* pada suatu *layer* dapat terhubung secara penuh.

## BAB III

### MODEL SISTEM

#### 3.1 Blok Diagram Sistem

Pada bab ini akan dijelaskan mengenai bagaimana cara mendeteksi objek jenis kendaraan menggunakan *python* ini dapat bekerja dengan menggunakan objek secara real time dan terdiri dari model sistem, realisasi sistem, dan pengujian. Adapun sistem kerja dari deteksi objek jenis kendaraan menggunakan *python* yang telah dibuat dapat digambarkan seperti ini:



Gambar 3.1 Model Sistem Deteksi Objek Jenis Kendaraan menggunakan Metode YOLO

Dengan membuat deteksi objek kendaraan dengan menggunakan permograman *python*. tahap awal dengan mengumpulkan gambar objek jenis kendaraan sebagai input. kemudian di proses dengan menggunakan sistem *Deep learning*. *Deep learning* merupakan bagian dari *Artificial Intelligence (AI)* dan *machine learning* yang dapat berfungsi mendeteksi objek, pengenalan suara dan sebagainya. Dalam sistem *Deep learning* terdapat *Convolutional network (CNN)*. Dalam CNN terdapat metode yang digunakan untuk mendeteksi objek, yaitu YOLO (*You Only Look Once*).

YOLO menggunakan pendekatan berbeda untuk mendeteksi objek jenis kendaraan. Dalam hal ini YOLO membagi input gambar menjadi grid berukuran  $S \times S$ , dimana nilai  $S$  adalah 7 dengan input gambar berukuran 448 x 448. Untuk selanjutnya memperoleh bounding box, dilakukan konvolusi dari input gambar. Sebuah bounding box memiliki 5 nilai yang perlu disimpan, koordinat  $x$ , koordinat  $y$ , lebar (width), tinggi (height), dan confidence score (nilai probabilitas bounding box pada sebuah objek). Untuk semua atribut pada bounding box akan dilakukan normalisasi sehingga nilainya menjadi antara 0 hingga 1. Koordinat  $x$  dan  $y$  akan dinormalisasi menyesuaikan titik kiri atas dari grid yang bersangkutan. Tinggi dan lebar akan dinormalisasi sesuai dengan ukuran objek yang dideteksi berupa jenis kendaraan. Berikut jenis kendaraan yang dapat dideteksi, antara lain :

<p>1. Mobil</p> 	<p>2. Motor</p> 
<p>3. Truk</p> 	<p>4. Sepeda</p> 

### 3.2 Tahapan Perencanaan

Proses sistem perancangan deteksi objek jenis kendaraan dengan metode YOLO dan prosesnya, tahapan pembuatannya adalah sebagai berikut :

1. Analisis *Dataset*

Langkah awal dari merancang deteksi objek jenis kendaraan menggunakan metode YOLO adalah dengan mempersiapkan dataset untuk proses training penulis menggunakan beberapa dataset dari internet dan pengambilan gambar di jalan raya yang didalamnya terdapat mobil, sepeda, truk, dan sepeda motor.

2. Mempelajari Framework

Untuk merancang deteksi objek jenis kendaraan ini diperlukan framework darknet atau darkflow dimana framework ini menggunakan library yang sama dengan tingkat akurasi mengenali objek lebih baik.

3. Model Sistem

Sebelum proses training perlu untuk mengkonfigurasi proses konvolusi dengan model sistem yang sudah dibuat. File untuk konfigurasi adalah file dengan ekstensi .cfg. YOLO sudah menyediakan konfigurasi default. Untuk membuat konfigurasi baru dapat menyalin konfigurasi yang disediakan kemudian ubah sesuai dengan kebutuhan.

4. Proses Training

Setelah semua dataset gambar diberi label langkah selanjutnya adalah melakukan training untuk menghasilkan model yang akan dipakai untuk pendeteksian objek. Semakin banyak jumlah gambar semakin lama juga proses trainingnya, tetapi akurasi yang dihasilkan akan semakin baik.

5. Testing

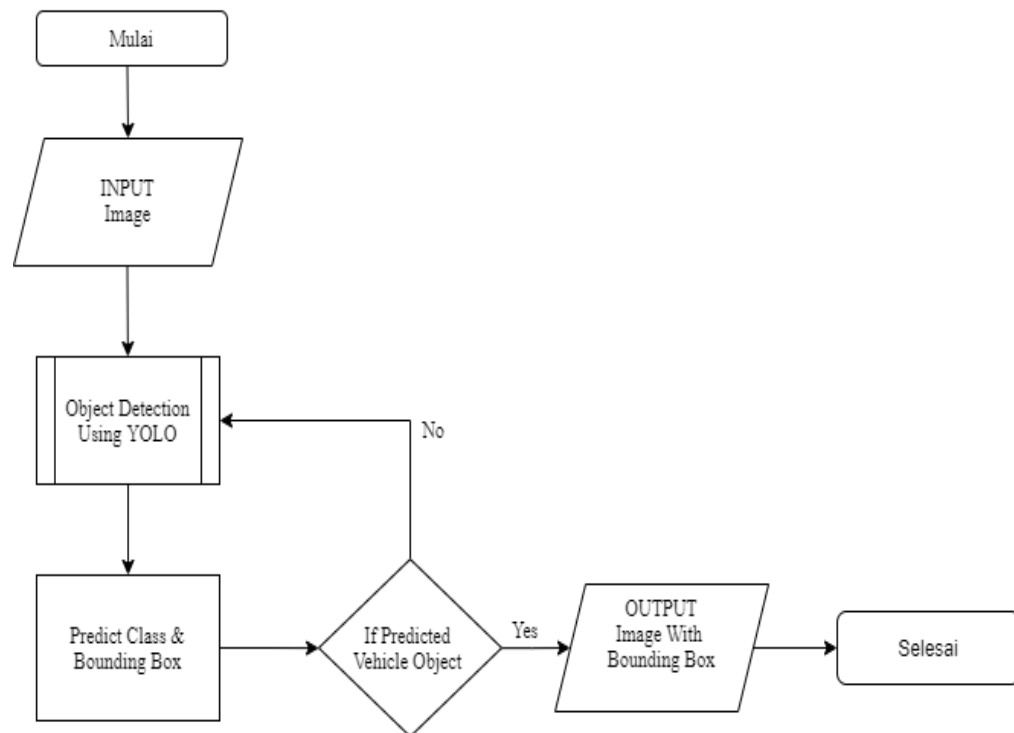
Setelah dilakukan proses training, melakukan tahapan testing untuk model sistem deteksi objek jenis kendaraan dengan menggunakan dataset yang telah dikumpulkan.

6. Optimasi

Optimasi dilakukan dari proses testing sebelumnya dengan tujuan sistem dapat bekerja dengan optimal dan memberikan hasil deteksi yang maksimal.

### 3.3 Perancangan

Pada Proyek Akhir ini akan dirancang sistem deteksi objek jenis kendaraan menggunakan metode YOLO yang diambil dari beberapa referensi yang telah digabungkan dengan menyelesaikan tahap perencanaan dimulai dari analisis dataset hingga melakukan optimasi agar sistem berfungsi dengan baik. Adapun langkah perancangan dalam penelitian ini di gambarkan dalam *flowchart* pada Gambar 3.3. yaitu :



Gambar 3.3 Diagram alir sistem Deteksi Objek Jenis Kendaraan Menggunakan Metode YOLO



## BAB IV

### BENTUK KELUARAN YANG DIHARAPKAN

#### 4.1 Keluaran Yang Diharapkan

Perancangan pada Proyek Akhir ini, akan dibuat sistem dengan spesifikasi sebagai berikut:

- Sistem dapat mendeteksi objek jenis kendaraan.
- Sistem dapat melakukan prediksi dataset yang digunakan.
- Sistem dapat mengidentifikasi objek jenis kendaraan.

#### 4.2 Jadwal Pelaksanaan

Adapun jadwal pengerjaan Proyek Akhir bisa dilihat pada tabel Tabel 4.1 sebagai berikut :

*Tabel 0.1 Jadwal Pelaksanaan*

Judul Kegiatan	Waktu					
	Feb	Mar	Apr	Mei	Juni	Juli
Studi Literatur						
Perancangan						
Dataset						
Model Sistem						
Pengujian						
Analisa						
Pembuatan Laporan						

## DAFTAR PUSTAKA

- [1] Shianto, K. A., Gunadi, K., & Setyati, E., "Deteksi Jenis Mobil Menggunakan Metode YOLO Dan Faster R-CNN". Jurnal Infra, 7(1), 157 – 163, 2019.
- [2] Limantoro, S. E., Kristian, Y., & Purwanto, D. D., "Pemanfaatan Deep Learning pada Video Dash Cam untuk Deteksi Pengendara Sepeda Motor". Jurnal Nasional Teknik Elektro Dan Teknologi Informasi, 7(2), 167–173, 2018.
- [3] Susanto, K. N., Gunadi, K., & Setyati, E., "Pengenalan Karakter pada Plat Nomor Indonesia dengan Tilt Correction dan Metode Faster R-CNN". Jurnal Infra, 7(1), 1–7, 2019.
- [4] Hardiansyah, "Pendeteksian Objek Secara Realtime Menggunakan Pendekatan Fast Convolutional Neural Network". Jurnal Ilmu Komputer JIK Vol. III, no ISSN : 2089-5305, 2020.
- [5] Afrizal Zein, "Pendeteksian Kantuk Secara Real Time Menggunakan Pustaka OPENCV dan DLIB PYTHON Real Time Sleepiness Detection Using OPENCV Library and PYTHON DLIB," vol. 28, no. 2, pp. 22–26, 2018.
- [6] Nugroho, H., "Deteksi Citra Objek Lingkaran Dengan Menggunakan Metode Ekstraksi Bentuk Circularity". INTEGER: Journal of Information Technology, 2(1), 2017.
- [7] Syafitri, N. and Saputra, A. "Prototype Pendeteksi Jumlah Orang Dalam Ruangan". IT Journal Research and Development, 1(2), pp. 36–48, 2017.
- [8] Syaikhoni, A., & Ariyadi, A. *Deteksi Objek Dengan Tensorflow Object Detection API*. Jakarta: <https://mti.binus.ac.id/2018/12/26/deteksi-objek-dengan-tensorflow-object-detection-api/>. 2018.
- [9] Hidayati, Q. *Kendali Lampu Lalu Lintas dengan Deteksi Kendaraan*. Jogjakarta: JNTETI, Vol. 6, No. 2. 2017.



# UNIVERSITAS TELKOM

## FAKULTAS ILMU TERAPAN

### KARTU KONSULTASI

## SEMINAR PROPOSAL PROYEK AKHIR

NAMA / PRODI : Rohman Prasetyo / D3Teknologi Telekomunikasi NIM : 6705184116

JUDUL PROYEK AKHIR : Deteksi Jenis Kendaraan Menggunakan Metode YOLO

CALON PEMBIMBING : I. Sugondo Hadiyoso, S.T., M.T.

II. Yuli Sun Hariyani, S.T., M.T.

NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING I
1	15-03-2021	BAB 1 (SELESAI)	
2	15-03-2021	BAB 2 (SELESAI)	
3	17-03-2021	BAB 3 (SELESAI)	
4	17-03-2021	BAB 4 (SELESAI)	
5	18-03-2021	FINALISASI PROPOSAL	
6			
7			
8			
9			
10			
NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING II
1		BAB 1 (SELESAI)	
2	15-03-2021	BAB 2 (SELESAI)	
3	17-03-2021	BAB 3 (SELESAI)	
4		BAB 4 (SELESAI)	
5	18-03-2021	FINALISASI PROPOSAL	
6			
7			
8			
9			
10			