

**IMPLEMENTASI METODE RESIDUAL NETWORK UNTUK DETEKSI
PENGUNAAN SABUK PENGAMAN PADA PENGEMUDI MOBIL**

SKRIPSI

Disusun sebagai salah satu syarat untuk memperoleh Gelar Sarjana Teknik pada
Program Studi Teknik Informatika di Universitas Sangga Buana YPKP



Disusun Oleh:

Firdan Abdul Kholiq

2113191037

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INFORMATIKA
UNIVERSITAS SANGGA BUANA YPKP
BANDUNG
2024**

LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Firdan Abdul Kholiq

Npm : 2113191037

Program Studi: Teknik Informatika

Menyatakan dengan sebenar-benarnya bahwa laporan Skripsi saya yang berjudul:

IMPLEMENTASI RESIDUAL NETWORK UNTUK DETEKSI PENGUNAAN SABUK PENGAMAN PADA PENGEMUDI MOBIL

Menyatakan dengan sebenarnya bahwa penulisan Skripsi ini berdasarkan hasil karya sendiri dan tidak melakukan plagiat dari hasil karya orang lain. Jika terdapat karya orang lain, saya akan mencantumkan sumber yang jelas sesuai dengan referensi yang relevant.

Demikian pernyataan ini saya buat dengan sungguh- sungguh dan apabila dikemudian hari ternyata ada pihak lain yang mengklaim baik dari judul maupun dari isi tugas akhir ini, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku di Universitas Sangga Buana YPKP.

Bandung, 04 Maret 2024



Firdan Abdul Kholiq

LEMBAR PERSETUJUAN DAN PENGESAHAN TUGAS AKHIR

Skripsi ini di ajukan oleh:

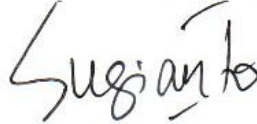
Nama : Firdan Abdul Kholiq
Npm : 2113191037
Program Studi : Teknik Informatika
Judul : IMPLEMENTASI RESIDUAL NETWORK UNTUK DETEKSI
PENGUNAAN SABUK PENGAMAN PADA PENGEMUDI
MOBIL

Untuk dipertahankan sidang Skripsi Semester Ganjil tahun 2024 di hadapan para penguji dan diterima sebagai bagian dari persyaratan yang di perlukan untuk memperoleh gelar Sarjana Teknik (ST) pada Fakultas Teknik Program Studi S1 Teknik Informatika Universitas Sangga Buana YPKP.

Bandung, 04 Maret 2024

Menyetujui,

Pembimbing



Bambang Sugiarto.ST., M.T.

NIDK. 8861060017


Penguji I



Slamet Risnanto.S.T., M.Kom.

NIDN. 0424047307

Penguji II



Gunawansyah, ST., M.Kom

NIDN. 0420027907

Mengetahui:

Ketua Program Studi S1 Teknik Informatika



Gunawan, ST., M.Kom., MOS., MTA., MCE

NIDN:040427604

KATA PENGANTAR

Puji Syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena dengan rahmat dan karunia-Nya, penulis dapat menyelesaikan skripsi berjudul “IMPLEMENTASI RESIDUAL NETWORK UNTUK DETEKSI PENGGUNAAN SABUK PADA PENGEMUDI MOBIL”. Skripsi ini dibuat dan diajukan untuk syarat untuk menyelesaikan Program Sarjana (S1) Fakultas Teknik jurusan Teknik Informatika di Universitas Sangga Buana YPKP. Saya menyadari bahwa menyelesaikan skripsi ini akan sangat sulit bagi saya jika saya tidak menerima bantuan dan bimbingan dari berbagai pihak, mulai dari masa perkuliahan hingga saat saya penulisan skripsi. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Bapak Gunawan, S.T., S. Kom., MOS., MCE. selaku Ketua Program Studi S1 Teknik Informatika.
2. Bapak Bambang Sugiarto, S.T., M.T selaku dosen pembimbing yang sudah berkenan menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini.
3. Seluruh Dosen dan Staf Fakultas Teknik Universitas Sangga Buana YPKP.
4. Kedua orang tua yang telah membantu baik secara moril dan materil.
5. Rekan-rekan seperjuangan yang memberikan dukungan dan bantuan dalam penyelesaian laporan ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas kebaikan semua pihak yang telah membantu. Saya menyadari bahwa skripsi ini masih jauh dari sempurna karena adanya keterbatasan ilmu dan pengalaman yang dimiliki. Oleh karena itu, semua kritik dan saran yang bersifat membangun akan penulis terima dengan senang hati. Penulis berharap semoga skripsi ini dapat bermanfaat bagi semua pihak yang memerlukannya.

ABSTRAK

Sabuk pengaman adalah salah satu komponen wajib yang digunakan oleh penumpang maupun pengemudi mobil dalam menjaga keamanan dan keselamatan. Salah satu komponen penting yang berkontribusi terhadap keselamatan penumpang mobil sering mengalami kecelakaan saat berkendara karena tidak menggunakan sabuk pengaman. Tujuan dari penelitian ini adalah mendeteksi penggunaan sabuk pengaman pada pengemudi mobil dengan menggunakan metode Residual Network (Resnet). Resnet-50 adalah jaringan sisa jaringan dengan 50 lapisan, dimana Resnet terkenal karena menggunakan blok sisa, untuk melatih jaringan yang lebih dalam. Hasil deteksi ini berupa label dari objek yang telah diberikan kotak pembatas, beserta nilai klasifikasi terhadap suatu kelas dari label objek yang dideteksi. Berdasarkan proses pengujian kinerja sistem ini menunjukkan nilai precision sebesar 90%, nilai recall sebesar 90%, nilai f1 score sebesar 89% dan nilai akurasi sebesar 87%.

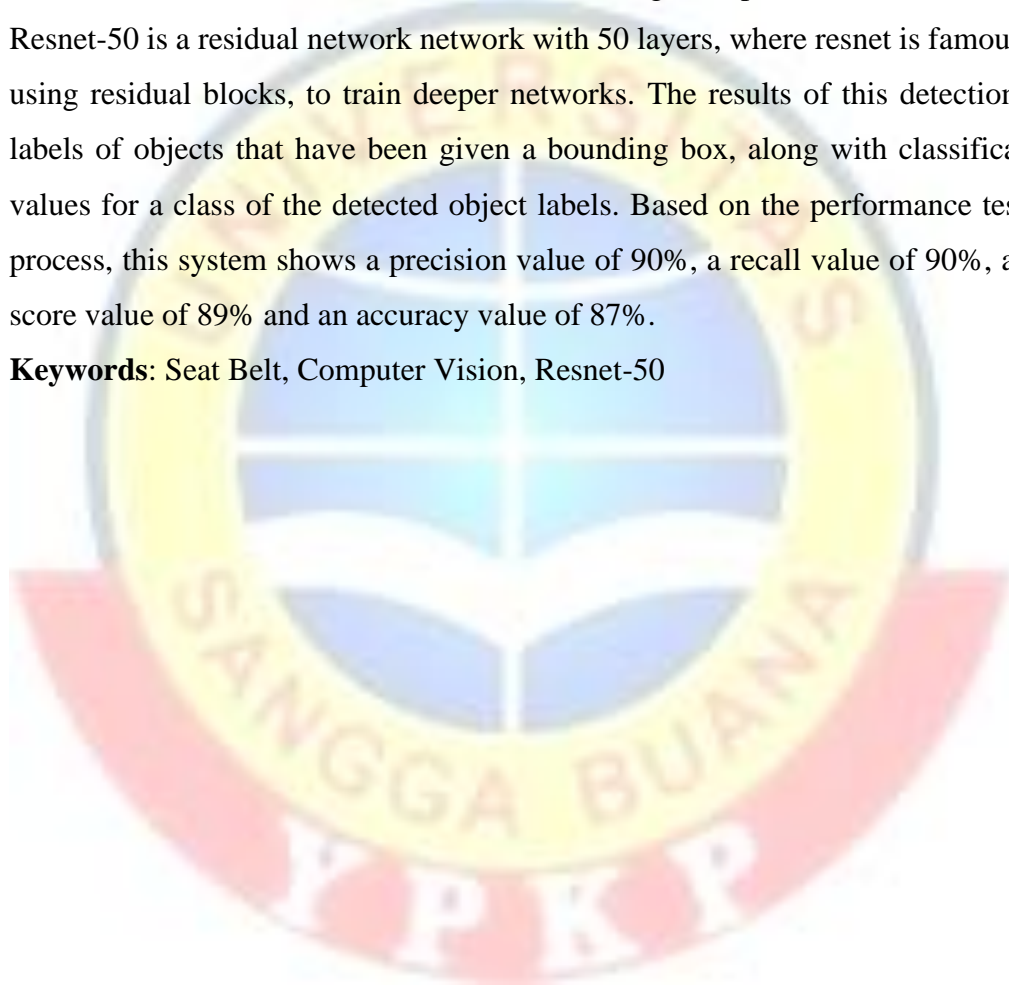
Kata Kunci: Sabuk Pengaman, Visi Komputer, Resnet-50



ABSTRACT

Seat belts are a mandatory component used by passengers and car drivers to maintain safety and security. One important component that contributes to the safety of car passengers in Indonesia is that they often experience accidents while driving because they are not wearing seat belts. In this research, the detection of seat belt use in car drivers was carried out using Computer Vision technology. Resnet-50 is a residual network network with 50 layers, where resnet is famous for using residual blocks, to train deeper networks. The results of this detection are labels of objects that have been given a bounding box, along with classification values for a class of the detected object labels. Based on the performance testing process, this system shows a precision value of 90%, a recall value of 90%, an f1 score value of 89% and an accuracy value of 87%.

Keywords: Seat Belt, Computer Vision, Resnet-50



DAFTAR ISI

LEMBAR PERNYATAAN KEASLIAN SKRIPSI	ii
LEMBAR PERSETUJUAN DAN PENGESAHAN TUGAS AKHIR	Error! Bookmark not defined.
KATA PENGANTAR	iii
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
BAB 1	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	3
1.6.1 Pengumpulan Data	3
1.6.2 Model Pengembangan Perangkat Lunak	4
1.6.3 Pengembangan sistem	4
1.7 Sistematika Penulisan	5
BAB II	7
LANDASAN TEORI	7

2.1	Tinjauan Pustaka	7
2.2	Sabuk pengaman	9
2.3	Pengelola Citra Digital.....	10
2.4	Image Processing	11
2.5	RGB	11
2.6	Computer Vision.....	12
2.7	Deep Learning.....	13
2.8	Convolutional Neural Network	13
2.3.1	Convolution Layer	14
2.3.2	Pooling Layer.....	15
2.3.3	Fully Connected Layer	16
2.3.4	Dropout	16
2.3.5	Zero Padding	18
2.3.6	ReLU Activation	18
2.4	Anchor Box	19
2.5	Classification Subnet.....	19
2.6	Box Regression Subnet.....	20
2.7	Focal Loss	20
2.8	Training Data.....	20
2.9	Preprocessing Data.....	21
2.10	Deteksi Objek.....	21
2.11	Jaringan Residual Network (<i>ResNet-50</i>)	22
2.12	Pengujian Kinerja Sistem	24
2.10	Python.....	26
2.10.1	Tensorflow	27
2.11	Website	28

2.13	UML (Unified Modeling Language)	28
BAB III	31
METODE PENELITIAN	31
3.1	Analisis dan Perancangan.....	32
3.1.1	Studi Literatur	32
3.1.2	Objek Penelitian	32
3.1.3	Pengumpulan Data	32
3.1.4	Analisis Kebutuhan.....	32
3.2	Tempat dan Waktu Penelitian	33
3.3	Dataset	33
3.4	Data Preprocessing.....	36
3.5	Skenario Penelitian.....	37
3.6	Dataset Training dan Testing	38
3.7	Permodelan Sistem	38
3.8	Convolutional Neural Network	40
3.9	Quick Plan & <i>Modelling Quick Desain</i>	42
3.4.1	Use Case	42
3.4.2	Activity Diagram	49
2.4.3	Sequence Diagram.....	51
3.4.4	Blok Diagram.....	52
3.4.5	Flowchart	54
3.4.6	Proses Residual Network.....	55
3.3	Pembuatan Prototype (Construction of Prototype)	56
3.4	Deployment Delivery & feedback	57
BAB IV	58
HASIL DAN PEMBAHASAN	58
4.1.	Lingkungan Pengembangan.....	58

4.1.1. Kebutuhan Pembangunan <i>Hardware</i>	58
4.1.2. Kebutuhan Perangkat Lunak	58
4.2 Implementasi Sistem	59
4.3.1 Antarmuka Sistem	59
4.3.2 Pemilihan Data Uji	59
4.3.3 Hasil Deteksi	61
4.4 Pengujian Kinerja Sistem	62
BAB V	65
PENUTUP	65
5.1 Kesimpulan	65
5.2 Saran	65
DAFTAR PUSTAKA	66

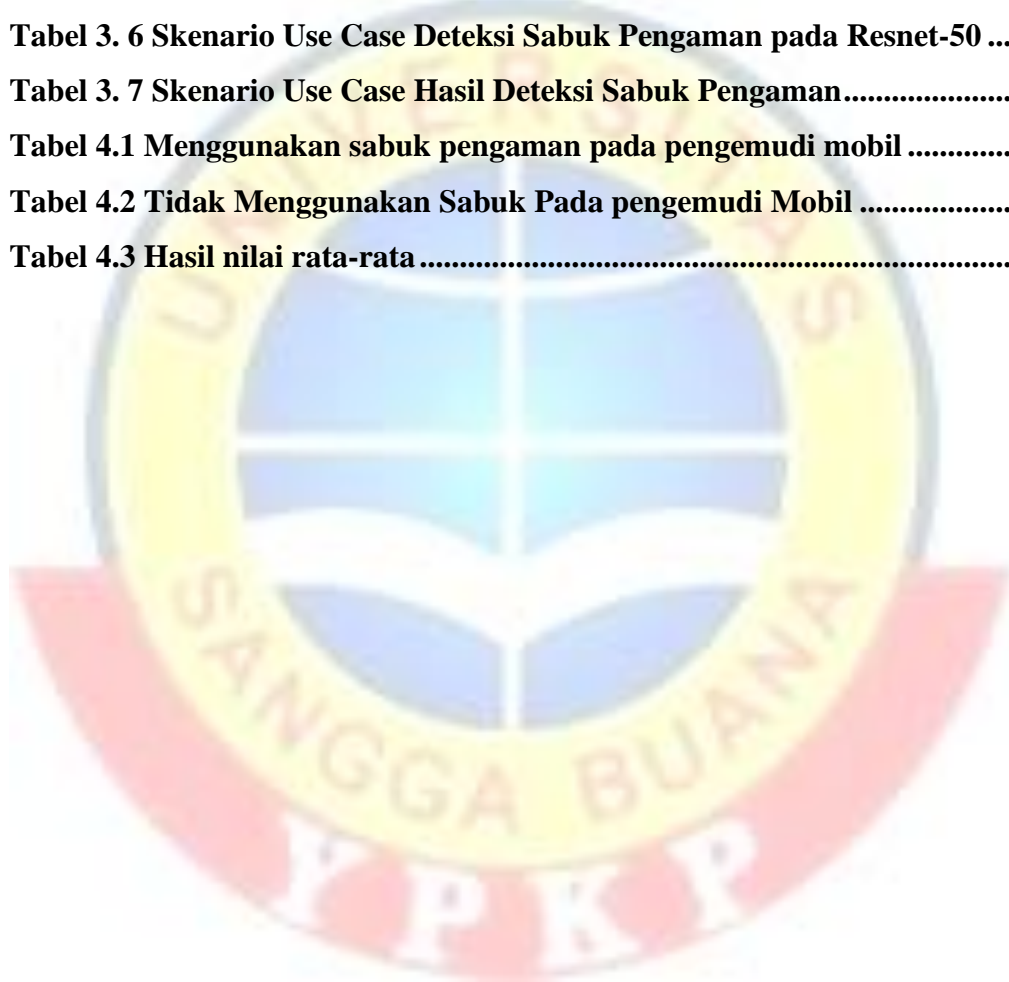


DAFTAR GAMBAR

Gambar 2.1 Sabuk Pengaman	10
Gambar 2.2 RGB.....	12
Gambar 2.3 Proses Convolutional Neural Network	14
Gambar 2.4 Proses Konvolusi.....	15
Gambar 2.5 Operasi Max Pooling	15
Gambar 2. 6 Fully Connected Layer	16
Gambar 2.7 Dropout.....	17
Gambar 2.8 Proses Zero Padding.....	18
Gambar 2.9 Anchor Box.....	19
Gambar 2.10 Arsitektur dan Jumlah Lapisan Resnet.....	23
Gambar 2.11 Confusion Matrix.....	25
Gambar 2. 12 Python	27
Gambar 3.1 Model Prototype.....	31
Gambar 3.4 Tampilan LabelImg.....	35
Gambar 3.6 Data Hasil pembuatan LabelImg	36
Gambar 3.7 Permodelan Sistem	39
Gambar 3.8 proses Convolutional Neural Network.....	41
Gambar 3.8 Use Case	42
Gambar 3.9 Activity Memilih Data Uji.....	50
Gambar 3.10 Activity Proses Deteksi Penggunaan Sabuk Pengaman	51
Gambar 3.11 Activity Proses Deteksi Penggunaan Sabuk Pengaman	52
Gambar 3.12 Blok Diagram	53
Gambar 3.13 Flowchart.....	55
Gambar 3.14 Antarmuka Sistem	56
Gambar 4.1 Antarmuka Sistem	59
Gambar 4.2 Pilih Data Uji.....	60
Gambar 4.4 Hasil Deteksi menggunakan Sabuk Pengaman.....	61
Gambar 4.3 Hasil Deteksi pengemudi menggunakan sabuk pengaman.....	61
Gambar 4.6 Hasil Deteksi tidak menggunakan Sabuk Pengaman.....	61

DAFTAR TABEL

Tabel 3. 1 Traning dan Testing	38
Tabel 3. 2 Skenario Use Case memilih dataset	43
Tabel 3. 3 Skenario Use Case Dataset Training	44
Tabel 3. 4 Skenario Use Case Dataset Testing.....	45
Tabel 3. 5 Skenario Use Case Data Processing.....	46
Tabel 3. 6 Skenario Use Case Deteksi Sabuk Pengaman pada Resnet-50	47
Tabel 3. 7 Skenario Use Case Hasil Deteksi Sabuk Pengaman.....	48
Tabel 4.1 Menggunakan sabuk pengaman pada pengemudi mobil	62
Tabel 4.2 Tidak Menggunakan Sabuk Pada pengemudi Mobil	63
Tabel 4.3 Hasil nilai rata-rata	63





BAB 1

PENDAHULUAN

1.1 Latar Belakang

Belakangan ini, terdapat minat yang signifikan dalam ilmu dan teknologi visi komputer di berbagai bidang penelitian, termasuk penegakan lalu lintas otomatis, sistem bantuan pengemudi, dan kendaraan otonom. Visi komputer berfokus pada kemampuan komputer untuk memperoleh pemahaman tingkat tinggi tentang gambar, memberikan manfaat seperti deteksi objek yang ditingkatkan dalam situasi di mana penglihatan manusia mungkin terbatas. Salah satu aplikasi penting dari visi komputer adalah dalam deteksi penggunaan sabuk pengaman.

Sabuk pengaman adalah salah satu komponen wajib yang digunakan oleh penumpang maupun pengemudi mobil dalam menjaga keamanan dan keselamatan (Wang, 2022). Keselamatan dalam berkendara merupakan aspek utama yang perlu diperhatikan untuk mengurangi risiko cedera yang dapat terjadi akibat kecelakaan lalu lintas. Salah satu komponen penting yang berkontribusi terhadap keselamatan penumpang adalah penggunaan sabuk pengaman. Sabuk pengaman telah terbukti secara efektif mengurangi risiko cedera dalam kecelakaan, terutama pada situasi dengan benturan yang kuat.

Setiap penumpang dan pengemudi mobil wajib memakai sabuk pengaman (UU LLAJ Pasal 106 ayat 6) (Revydo Bima Anshori, 2022). Ketika penumpang maupun pengemudi mengalami kecelakaan saat berkendara. Pengemudi dengan menggunakan sabuk pengaman, dapat mengurangi risiko cedera fatal sebesar 45% dan risiko cedera kritis sebesar 50% (DS Bhupal Naik, 2021). Menurut statistik, ada 1.616 pelanggaran pengemudi mobil yang tidak menggunakan sabuk pengaman pada tahun 2019 dan angka ini meningkat menjadi 1.885 pelanggaran yang tidak menggunakan sabuk pengaman pada tahun 2020 .

Saat ini, pemeriksaan sabuk pengaman umumnya dilakukan secara manual oleh otoritas, yang diketahui tidak efisien, mahal, dan tidak efektif. Pada tahun 2016, Administrasi Keselamatan Lalu Lintas Jalan Raya Nasional (NHTSA)

melaporkan bahwa 10.428 nyawa hilang di Amerika Serikat akibat tidak menggunakan sabuk pengaman. Menurut Organisasi Kesehatan Dunia (WHO), penggunaan sabuk pengaman dapat mengurangi tingkat kematian hingga 50% untuk penumpang kursi depan.

Kegagalan menggunakan sabuk pengaman adalah faktor kontribusi utama terjadinya kecelakaan, dengan banyak penumpang yang mengabaikannya. Pada saat pengereman tiba-tiba pada kecepatan tinggi, penumpang yang tidak menggunakan sabuk pengaman dapat terpental ke depan, berpotensi mengakibatkan luka serius akibat benturan dengan dashboard atau kaca depan. Hal ini terjadi karena penumpang tidak mengalami perlambatan yang sama dengan kendaraan, sehingga tubuh mereka terus bergerak ke depan sementara kendaraan melambat.

Berdasarkan uraian latar belakang masalah di atas, dibutuhkan sistem untuk mengidentifikasi pengemudi mobil yang memakai sabuk pengaman dan yang tidak memakai sabuk pengaman. Penelitian ini bertujuan untuk menemukan penggunaan sabuk pengaman pada pengemudi mobil menggunakan pendekatan Residual Network (ResNet).

1.2 Rumusan Masalah

Berdasarkan latar belakang yang ada, maka dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana pendekatan *Residual Network (ResNet)* dalam mengembangkan sistem deteksi penggunaan sabuk pengaman otomatis pada pengemudi mobil?
2. Bagaimana tingkat akurasi sistem dalam mendeteksi penggunaan sabuk pengaman pada pengemudi mobil?

1.3 Tujuan

Berdasarkan rumusan masalah yang telah disampaikan di atas, penelitian ini bertujuan sebagai berikut:

1. Mengembangkan sebuah sistem yang dapat mendeteksi penggunaan sabuk pengaman pada pengemudi mobil dengan pendekatan Residual Network (ResNet).
2. Mengukur seberapa akurat sistem untuk deteksi penggunaan sabuk pengaman pengemudi mobil.

1.4 Batasan Masalah

Dalam penelitian yang akan dilakukan, dibatasi ruang lingkup yang akan dibahas, yaitu sebagai berikut:

1. Pengambilan data uji dilakukan dengan kamera menghadap ke arah depan di persimpangan jalan raya Soekarno Hatta dan Gasibu.
2. Pengambilan data uji dilakukan pada pagi dan sore hari.
3. Kendaraan dengan kaca film memiliki batasan kegelapan maksimum sebesar 20%.
4. Sabuk pengaman pada pengemudi mobil yang menggunakan kerudung tidak terdeteksi.

1.5 Manfaat Penelitian

Penelitian ini dilakukan untuk dapat meningkatkan kesadaran pengemudi untuk mematuhi aturan keselamatan lalu lintas dengan menggunakan sabuk pengaman. Sistem dapat mendeteksi pengemudi sabuk pengaman sehingga polisi dapat memonitor kepatuhan pengemudi terhadap aturan keselamatan lalu lintas tanpa perlu melakukan penindakan langsung di jalan raya.

1.6 Metodologi Penelitian

Untuk menyelesaikan tugas akhir ini, penulis melakukan beberapa tahapan metode penelitian sebagai berikut ini:

1.6.1 Pengumpulan Data

1. Studi Kepustakaan

Merupakan tahapan awal dalam metode pengumpulan data dan tahapan untuk memahami konsep dari implementasi metode *resnet-50* untuk deteksi penggunaan sabuk pengaman pada pengemudi mobil. Pengumpulan data dan pemahaman

diperoleh dari artikel, jurnal, buku dan sumber informasi dari internet yang mendukung proses penulisan.

1.6.2 Model Pengembangan Perangkat Lunak

Model *Prototype* menjelaskan mengenai perancangan sistem. Dimana *Prototype* ini dipilih dan digunakan karena dapat memecahkan permasalahan-permasalahan dalam waktu relatif yang singkat. Model *prototype* terdiri dari beberapa tahap yang harus dilakukan dan setiap tahap memiliki aktivitas-aktivitas tersendiri seperti:

1. *Communication*, Pada bagian analisis kebutuhan dijelaskan mengenai kebutuhan yang dibutuhkan dalam pembangunan sistem pada penelitian ini. Kebutuhan tersebut terbagi menjadi dua yaitu kebutuhan pembangunan hardware dan kebutuhan pembangunan software
2. *Quick Plan & Modelling Quick Desain*, proses desain awal untuk sistem yang dibangun. Salah satu tahapannya adalah perancangan alur kerja sistem dan Blok diagram, dan Flowchart untuk proses yang berinteraksi dengan sistem menggunakan Unified Modelling Language (UML).
3. Pembangunan *Prototype*, Pada tahap ini dibuat perancangan antarmuka sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil.
4. *Deployment Delivery & feedback*, dilakukan untuk menguji apakah prototype yang sudah dibuat memenuhi tujuan dan rancangan penelitian yang sudah ditentukan.

1.6.3 Pengembangan sistem

Pengembangan sistem ini menggunakan metode *ResNet-50*. *ResNet* adalah jaringan Residual yang memiliki jaringan berjumlah 50 kedalaman konvolusi. Dalam arsitektur resnet memiliki 5-layer, layer pertama yaitu 18 kedalaman konvolusi, layer kedua 34 kedalaman konvolusi, layer ketiga 50 kedalaman konvolusi, layer keempat 101 kedalaman konvolusi, dan layer kelima 152 kedalaman konvolusi. Arsitektur ResNet50 dibagi menjadi empat bagian utama: lapisan konvolusional, blok identitas, blok konvolusional, dan lapisan terhubung penuh. Lapisan konvolusional bertanggung jawab untuk mengekstraksi fitur dari

gambar masukan, sedangkan blok identitas dan blok konvolusional bertanggung jawab untuk memproses dan mengubah fitur tersebut. Terakhir, lapisan yang terhubung sepenuhnya digunakan untuk membuat klasifikasi akhir. Lapisan konvolusional di *ResNet-50* terdiri dari beberapa lapisan konvolusional yang diikuti dengan normalisasi batch dan aktivasi ReLU. Lapisan ini bertanggung jawab untuk mengekstraksi fitur-fitur dari gambar masukan, seperti tepi, tekstur, dan bentuk. Lapisan konvolusional diikuti oleh lapisan penggabungan maksimal, yang mengurangi dimensi spasial peta fitur sambil mempertahankan fitur yang paling penting. Blok identitas dan blok konvolusional adalah blok penyusun utama *ResNet-50*. Blok identitas adalah blok sederhana yang meneruskan masukan melalui serangkaian lapisan konvolusional dan menambahkan masukan kembali ke keluaran

1.7 Sistematika Penulisan

Berikut ini adalah kerangka sistematika penulisan yang digunakan dalam penulisan laporan skripsi ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini membahas isi dari penelitian yang dilakukan. Bagian ini mencakupi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat penelitian, dan metodologi penelitian, serta sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menguraikan pengembangan atau penjelasan tambahan dari tinjauan pustaka dan teori-teori yang mendukung penelitian ini.

BAB III METODE PENELITIAN

Bab ini berisi perancangan sistem penelitian. Dalam bagian ini disajikan use case, blok diagram, activity diagram, sequence diagram, pemodelan sistem dan *flowchart* untuk menggambarkan perancangan yang dilakukan.

BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas sistem dari rancangan penelitian, dan subbab Pengujian hasil sistem untuk mencapai hasil penelitian.

BAB V PENUTUP

Bab ini berisi kesimpulan dan saran yang disajikan secara terpisah. Sub-bab Kesimpulan memberikan ringkasan tentang hasil pengujian dari implementasi sistem yang telah dibuat, Sedangkan sub-bab saran memberikan rencana untuk pengembangan sistem lebih lanjut.



BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam kegiatan penelitian ini akan dilakukan beberapa pustaka yang terkait dengan topik penelitian, antara lain:

Penelitian (Wang, 2022) berjudul *Intellegent Detection of Vehicle Driving Safety Based on Deep Learning*. Penelitian ini bertujuan untuk mengidentifikasi pengemudi yang menggunakan sabuk pengaman dengan menggunakan pendekatan *Squeeze YOLO* dan *Deconv SSD*. Saat resolusi gambar konsisten, akurasi *Deconv SSD* di bandingkan dengan algoritme *SSD* dalam data *PASCALVOC* dari 77,2% menjadi 79,6%. Pada dataset mendeteksi sabuk pengaman, *Squeeze YOLO* dapat mencapai akurasi 99,96% pada kecepatan 305 FPS, dan validitas dari percobaan diverifikasi. Kontribusi dari penelitian ini adalah untuk memperoleh pemahaman yang lebih baik tentang identifikasi pengemudi yang mematuhi penggunaan sabuk pengaman dan yang tidak, menggunakan metode *Backpropagation*.

Penelitian (Revydo Bima Anshori, 2022) berjudul *Klasifikasi Citra Kanker Serviks Menggunakan Deep Residual*. Penelitian ini berfokus pada klasifikasi gambar kanker serviks berdasarkan hasil pemeriksaan *IVA*. Hasil pemeriksaan *IVA*, yang terdiri dari gambar hasil deteksi tepi *canny* dan *CLAHE*, akan digunakan. Untuk klasifikasi, kami akan menggunakan pembelajaran mendalam dengan arsitektur *ResNet-50* dan *ResNet-101*, serta pengujian terhadap hyperparameter, termasuk *optimizer*, *learning rate*, *batch size*, dan ukuran input. Hasil terbaik dari penelitian ini akan diperoleh dengan menggunakan citra hasil deteksi tepi *Canny* dengan hyperparameter meter menggunakan *optimizer SGD*, tingkat pembelajaran 0.1, ukuran batch 32, dan ukuran input 224 x 224. Akurasi 98,26% dicapai dari hyperparameter tersebut.

Penelitian (DS Bhupal Naik, 2021) berjudul *Driver Seat Belt Detection using CNN*. Penelitian ini difokuskan pada identifikasi objek sabuk pengaman dalam gambar. Tujuan dari penelitian ini adalah untuk mengevaluasi tingkat keakuratan

dalam mendeteksi objek sabuk pengaman pada gambar menggunakan teknik *Convolution Neural Network* dan *Support Vector Machine*.

Penelitian (Irma Amelia Dewi, 2019) berjudul *Deep Learning RetinaNet based Car Detection for Smart Transportation Network*. Pada deteksi objek membantu mengkategorikan banyak objek dalam satu gambar. Ini tidak hanya menunjukkan kelas dari masing-masing objek yang terkait dengan pelabelan pada bounding box, tetapi juga menunjukkan jenis objek untuk menunjukkan keberadaannya. Tujuan dari penelitian ini adalah untuk mengevaluasi keakurasian teknik CNN yang menggunakan model RetinaNet untuk mendeteksi objek mobil. Selain itu, penelitian ini melakukan proses pengujian pada lima puluh gambar uji untuk mengukur kinerja sistem. Hasilnya menunjukkan ketepatan mencapai 86%, recall mencapai 85%, dan skor f1 mencapai 84%.

Penelitian (Nur Fadlia, 2019) berjudul *Klasifikasi Jenis Kendaraan Menggunakan Metode Convolutional Neural Network (CNN)*. Penelitian ini dilakukan dengan tingkat ketepatan sebesar 73,33%. *CNN* memakai empat lapisan konvolusi dengan dimensi filter 3×3, menggunakan fungsi aktivasi reLu, dan dua lapisan pengelompokan dengan dimensi 2×2. Hasil percobaan dan evaluasi model terhadap ketiga jenis kendaraan menunjukkan tingkat ketepatan sebesar 94,4% selama tahap pelatihan, dan 73,3% selama tahap pengujian.

Penelitian (Alexey Kashevnik, 2020) berjudul *Seat Belt Fastness Detection Based on Image Analysis from Vehicle In-abin Camera*. Penelitian ini menggunakan arsitektur YOLO untuk pengembangan algoritma berbasis ekstraksi fitur multi skala untuk mendeteksi model *convolution neural networks* (CNN) dan meningkatkan akurasi dengan menggunakan skor deteksi untuk model *Support Vector Machine* (SVM) untuk deteksi kekencangan sabuk pengaman pada kendaraan. Tujuan dari penelitian ini adalah mengklasifikasikan kekencangan sabuk pengaman pada kendaraan dengan menggunakan CNN dan SVM untuk mendeteksi objek.

Penelitian (Suprihanto, 2022) berjudul *Analisis Kinerja Resnet-50 dalam klasifikasi penyakit pada Daun Kopi Rebusta*. Indonesia merupakan negara agraris yang banyak ditanami tumbuhan salah satunya yaitu tanaman kopi. Dalam

budidaya kopi terdapat halangan seperti hama dan cuaca ekstrim yang bisa membuat tanaman layu atau terkena penyakit. Dengan teknologi yang berkembang pesat pada masa kini banyak sistem yang membantu para petani untuk membantu mengidentifikasi penyakit pada daun. Sistem ini menggunakan teknologi Convolutional Neural Network untuk mengidentifikasi dan mengklasifikasi penyakit pada daun kopi. Dalam pembangunan Arsitektur CNN diperlukan proses data training, data validation, dan data testing. Banyak metode dan arsitektur CNN yang memperoleh kinerja yang tinggi, salah satunya adalah ResNet-50. Performa dari model ini dihitung dengan menggunakan Confusion Matrix yang di dalamnya menghitung accuracy, precision, recall, specificity, dan F1-Score pada arsitektur ResNet-50 dimana berfokus pada nilai accuracy dan F1-Score pada model tersebut, penelitian dilakukan dengan dua kasus yaitu binary class classification dan multiclass classification dimana binary class untuk mengklasifikasi gambar daun kopi robusta sehat dengan yang tidak sehat dan multiclass untuk mengklasifikasikan gambar daun kopi robusta sehat dan daun sakit yang telah terbagi pada setiap jenis kategori yang ada. Hasil dari penelitian menunjukkan pada kasus binary class mencapai accuracy 92,68% dan F1-Score mencapai 92,88%, sedangkan pada kasus multiclass accuracy hanya mencapai 88,98% dan F1-Score mencapai 88,44%. Kedua kasus tersebut diukur menggunakan data testing dengan model ResNet-50 yang telah dilatih.

2.2 Sabuk pengaman

Sabuk pengaman sangat efektif dalam mengurangi kemungkinan cedera parah atau fatal pada penumpang jika terjadi tabrakan kendaraan. Kadang-kadang, penumpang mungkin lupa atau lalai mengencangkan sabuk pengaman mereka, sehingga mengurangi efektivitas perlindungan yang diberikan sabuk pengaman. Dalam situasi seperti ini, alat pendeteksi sabuk pengaman dapat menjadi solusi efektif untuk meningkatkan kesadaran pengemudi dan penumpang mengenai pentingnya penggunaan sabuk pengaman.

Alat pendeteksi sabuk pengaman ini dirancang untuk mengetahui apakah sabuk pengaman telah dipasang oleh penumpang. Hal ini dapat dicapai melalui berbagai teknologi, seperti sensor tekanan pada kursi, sensor geser, atau sensor

pengenalan penumpang. Ketika perangkat mendeteksi bahwa sabuk pengaman tidak dipasang, peringatan visual atau pendengaran dapat diaktifkan untuk mengingatkan pengemudi atau penumpang agar mengencangkan sabuk pengaman.

Penerapan alat pendeteksi sabuk pengaman dapat membantu meningkatkan kepatuhan penggunaan sabuk pengaman dan secara signifikan mengurangi risiko cedera serius atau fatal dalam kecelakaan kendaraan. Selain itu, hal ini dapat membantu kepatuhan terhadap undang-undang dan peraturan keselamatan lalu lintas yang mewajibkan penggunaan sabuk pengaman.

Oleh karena itu, implementasi mendeteksi penggunaan sabuk pengaman dapat menjadi langkah efektif dalam meningkatkan keselamatan pengemudi dan penumpang di jalan.



Gambar 2.1 Sabuk Pengaman

2.3 Pengolah Citra Digital

Pengolahan citra digital adalah proses pengolahan gambar secara digital melalui penggunaan komputer. Dengan kata lain, pengolahan citra digital menggunakan algoritma komputer untuk memperoleh gambar yang disempurnakan atau mengekstrak berbagai informasi berguna. Citra digital terdiri dari beberapa elemen, dimana setiap elemen mempunyai nilai pada titik tertentu. Elemen-elemen ini disebut sebagai elemen gambar dan piksel. Piksel paling banyak digunakan untuk mewakili elemen gambar digital. Pengolahan citra digital merupakan suatu

metode pengoperasian citra, dimana terdapat pengolahan sinyal. Proses-proses tersebut merupakan masukan citra dan keluaran citra atau karakteristik dan fitur yang terkait dengan citra tersebut. Saat ini pengolahan citra digital merupakan teknologi yang berkembang pesat dan telah menjadi salah satu disiplin ilmu komputer. Teknik pengolahan citra digital meliputi beberapa tahapan yaitu mengimpor citra melalui alat akuisisi citra, menganalisis dan memanipulasi citra serta keluarannya yang hasilnya dapat diubah menjadi citra atau laporan berdasarkan analisis citra. Teknik yang digunakan untuk pengolahan citra digital adalah pengolahan citra analog dan digital. Analisis gambar menggunakan berbagai dasar interpretasi ketika menggunakan teknik visual ini. Teknik pengolahan citra digital membantu dalam memanipulasi gambar digital dengan menggunakan komputer. Tiga fase umum yang harus dilalui semua jenis data saat menggunakan teknik digital adalah pra-pemrosesan, peningkatan, dan tampilan, ekstraksi informasi.

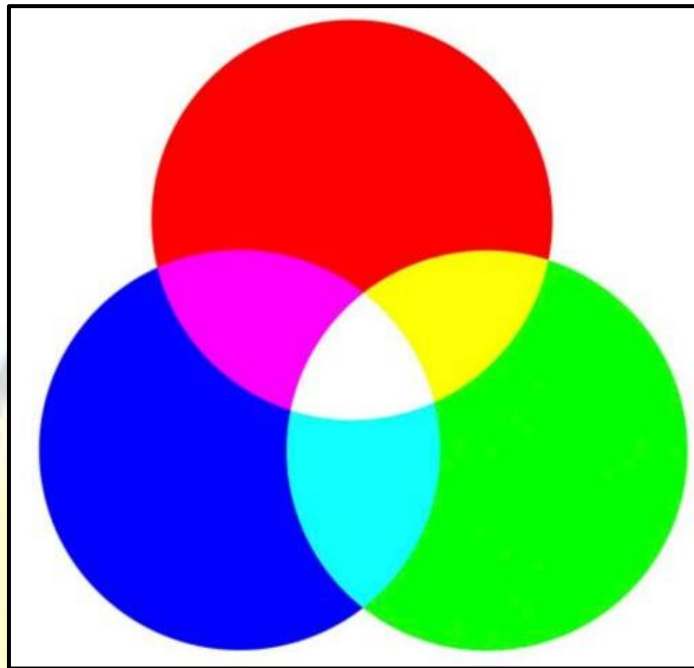
2.4 Image Processing

Pengolahan Citra atau Image Processing adalah suatu sistem yang didalamnya dilakukan proses dengan masukan berupa gambar dan hasilnya (output). Pada awalnya pengolahan citra ini dilakukan untuk meningkatkan kualitas citra, namun dengan ditandai berkembangnya dunia komputasi dengan semakin meningkatnya kapasitas dan kecepatan pemrosesan komputer, serta munculnya ilmu komputer yang memungkinkan manusia untuk mengambil informasi dari suatu gambar. image merupakan suatu pengolahan gambar yang tidak dapat dipisahkan dari bidang computer vision.

2.5 RGB

Untuk citra berwarna digunakan model RGB (Red-Green-Blue), satu citra berwarna dinyatakan dalam 3 matriks skala abu-abu berupa matriks untuk Merah (R-layer), satu matriks Hijau (G-layer) dan matriks untuk Biru (B-layer).R-layer adalah matriks yang menyatakan derajat kecerahan warna merah (misalnya untuk skala abu-abu 0-255, nilai 0 mewakili gelap (hitam) dan 255 menyatakan merah. G-layer adalah matriks yang menyatakan derajat kecerahan untuk warna hijau, dan

B-layer merupakan matriks yang mewakili derajat kecerahan untuk warna biru. Dari definisi tersebut, untuk menyajikan suatu warna tertentu dapat dilakukan dengan mudah, yaitu dengan mencampurkan ketiga warna dasar RGB (Hendy Mulyawan, 2011).



Gambar 2.2 RGB

(Hendy Mulyawan, 2011)

2.6 Computer Vision

Sistem visi komputer memanfaatkan teknologi kecerdasan buatan (AI) untuk meniru kemampuan otak manusia dalam mengenali dan mengklasifikasikan objek. Ilmuwan komputer melatih komputer untuk mengenali data visual dengan memasukkan informasi dalam jumlah besar. Algoritme pembelajaran mesin (ML) mengidentifikasi pola umum dalam gambar atau video ini dan menerapkan pengetahuan ini untuk mengidentifikasi gambar pengemudi memakai sabuk pengaman secara akurat. Jika komputer memproses gambar pengemudi mobil, komputer akan mulai membuat pola identifikasi yang dapat mendeteksi penggunaan sabuk pengaman pada gambar mobil secara akurat. Klasifikasi citra memungkinkan komputer untuk melihat citra dan secara akurat mengklasifikasikan citra dalam kelas. Penglihatan komputer memahami kelas dan membuat label pengemudi sabuk pengaman, kamera dapat mengenali sabuk pengaman dalam

sebuah foto tersebut. Tugas visi komputer yang bertujuan mengidentifikasi dan melokalisasi objek dalam gambar. Ini menggunakan klasifikasi untuk mengidentifikasi, mengurutkan, dan mengatur gambar. Deteksi objek digunakan dalam berbagai proses untuk mengontrol kamera di jalan raya untuk deteksi objek untuk memproses streaming video langsung dari kamera guna mendeteksi orang dan objek secara real-time dan memberikan notifikasi yang dapat ditindak lanjuti kepada pengguna akhir.

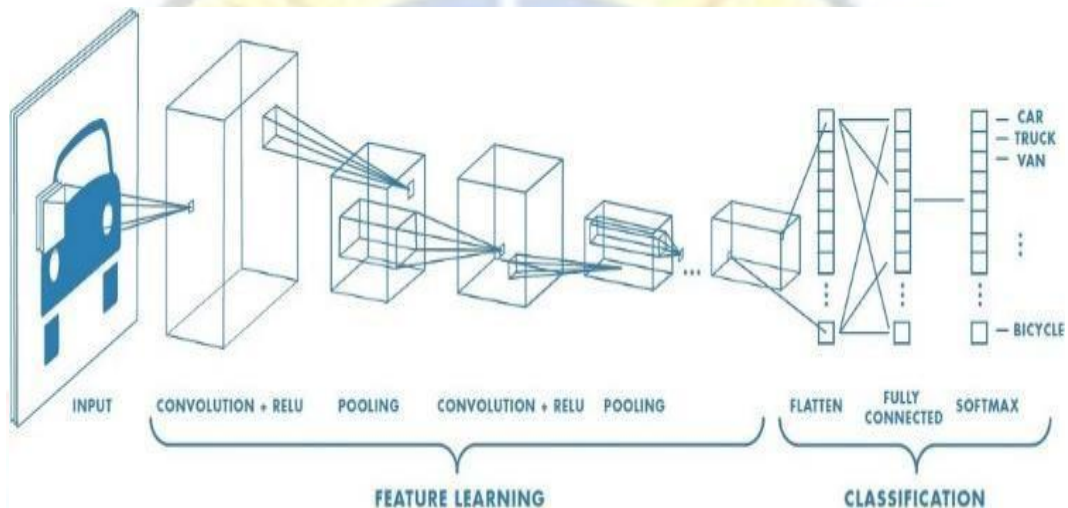
2.7 Deep Learning

Deep learning merupakan suatu bidang dari *machine learning*, pengembangan Multilayer Neural Network untuk memungkinkan deteksi objek, pengenalan suara, dan terjemahan bahasa. Dikembangkan pada tahun 1950, kemudian dapat di aplikasikan dengan sukses pada tahun 1990 (Eko Cahyono Putro, 2020). Metode *deep learning* terdiri dari berbagai lapisan atau dalam mempelajari karakteristik data dengan berbagai tingkat abstraksi (Jean Baptista Jimmy Robert Openg, 2022). *deep learning* memberikan arsitektur yang sangat baik untuk supervised learning. Oleh karena itu, *deep learning* dianggap sebagai bagian dari *machine learning*, *deep learning* terdiri dari banyak tingkatan proses informasi non-linear dan linear untuk memungkinkan *supervised learning*, *unsupervised learning* dan *semi supervised learning* (Aditya Santoso, 2018) Salah satu jenis algoritma jaringan saraf tiruan yang menggunakan data adalah *deep learning*. Algoritma ini menerima data sebagai masukan dan kemudian memprosesnya dengan menggunakan sejumlah *hidden layer* untuk melakukan transformasi non linier pada data masukan, kemudian algoritma ini untuk menghasilkan nilai keluaran (N.Nufus, 2021).

2.8 Convolutional Neural Network

Convolutional Neural Network adalah pengembangan dari *Multilayer Perceptron* (MLP) untuk mengelola data dalam bentuk citra. CNN digunakan untuk menganalisis gambar visual, mendeteksi dan mengenali objek image (Pulung Adi Nugroho, 2020). Dalam perkembangan *Convolution Neural Network* (CNN), terdapat beragam arsitektur yang sering digunakan seperti AlexNet, VGGNet,

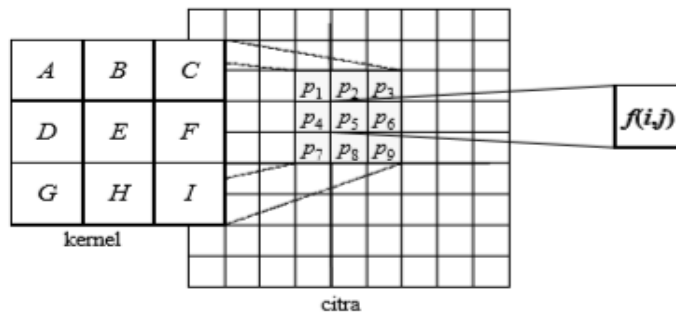
GoogLeNet, dan ResNet. Secara keseluruhan, CNN terdiri dari tiga lapisan utama, yaitu lapisan konvolusi, lapisan pengelompokan (pooling), dan lapisan terhubung sepenuhnya (fully connected layers). Pada lapisan konvolusi, operasi konvolusi diterapkan pada gambar input menggunakan kernel atau filter untuk mengekstrak fitur-fitur gambar. Selanjutnya, fungsi dari lapisan pengelompokan adalah untuk mengurangi dimensi dari fitur yang dihasilkan oleh lapisan konvolusi. Sedangkan pada lapisan terhubung sepenuhnya, gambar yang telah dikecilkan oleh lapisan pengelompokan dikonversi menjadi satu dimensi dan diklasifikasikan berdasarkan data latih yang tersedia dalam basis data. Gambar 2.1 di bawah ini memberikan gambaran umum tentang arsitektur CNN.



Gambar 2.3 Proses Convolutional Neural Network

2.3.1 Convolution Layer

Konvolusi adalah operasi untuk mengalikan nilai masukan citra dengan *filter* dalam pengolahan citra. *Konvolusi* dilakukan dengan menggeser *filter* citra dengan setiap pixel, kemudian menyimpan di dalam matriks baru (Munir, 2004). *Filter* bergerak secara iteratif dari posisi kiri atas hingga ke kanan bawah pada gambar. Apabila kernel bergerak satu kolom dari kiri kanan, itu disebut dengan 1 *strides*, dan jika filter bergerak satu kolom, itu disebut dengan 2 *strides*. Kemudian di filter dengan matrix 3x3. Pada Gambar 2.2 menjelaskan tentang proses konvolusi pada citra image.



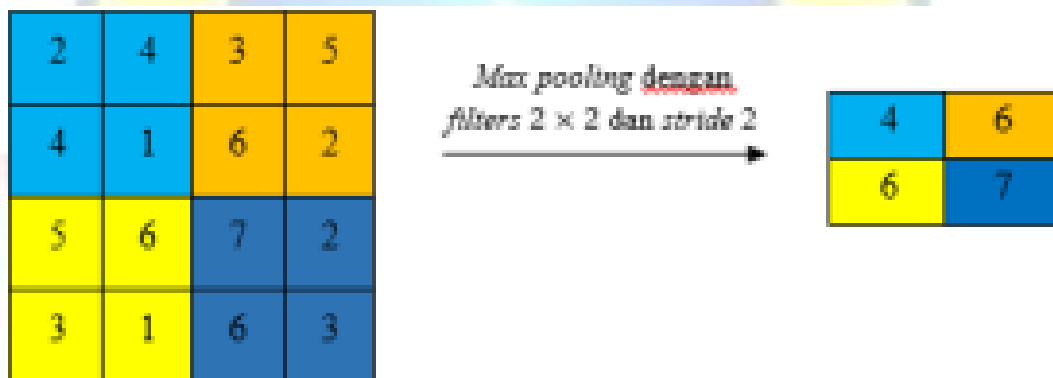
$$f(i,j) = A p_1 + B p_2 + C p_3 + D p_4 + E p_5 + F p_6 + G p_7 + H p_8 + I p_9$$

Gambar 2.4 Proses Konvolusi

(Munir, 2004)

2.3.2 Pooling Layer

Pooling Layer merupakan tahap setelah *Convolutional Layer*. Pooling layer berfungsi untuk mengurangi volume output pada feature map. Ini sehingga mengurangi jumlah perhitungan dan parameter di jaringan untuk mencegah overfitting. Operasi max pooling ditampilkan pada Gambar 2.3.



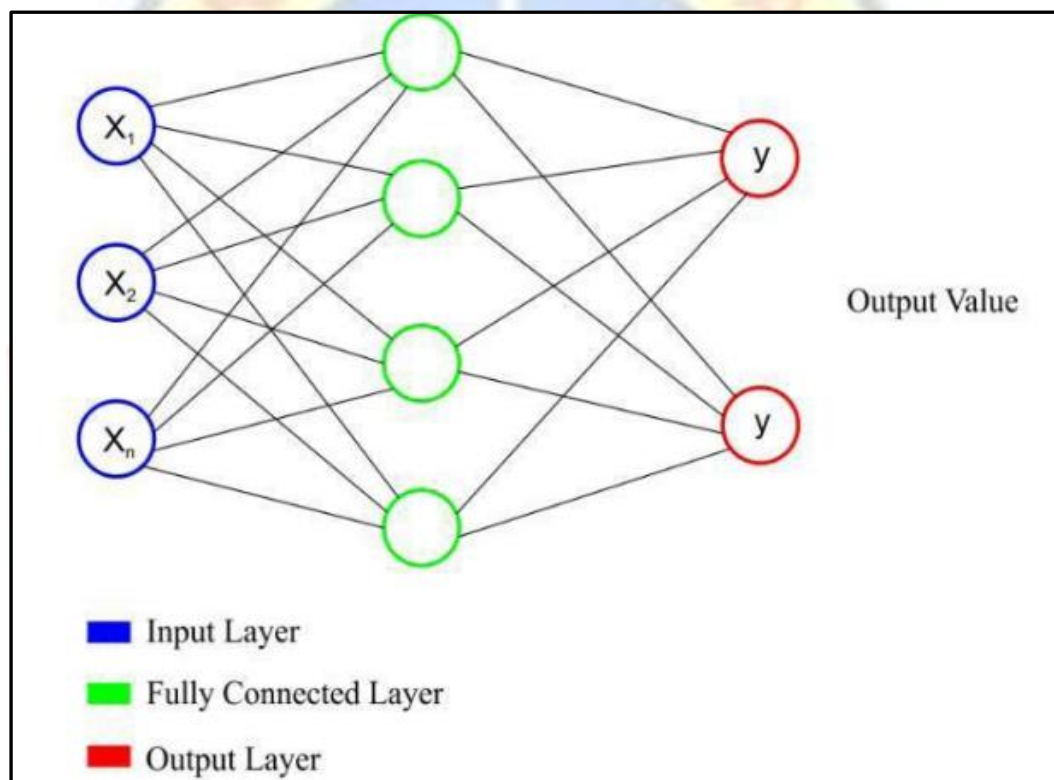
Gambar 2.5 Operasi Max Pooling

(Novelita Dwi Miranda, 2020)

Terdapat dua jenis fungsi pooling layer, yaitu *max pooling* dan *average pooling*. *Max pooling* merupakan proses yang mengambil nilai tertinggi dari setiap layer, seperti yang ditunjukkan pada Gambar 2.3, di mana terdapat matriks citra berukuran 4x4 digunakan untuk melakukan proses *max pooling* dengan *matriks filter 2x2* dengan pergeseran pixel 2 *strides* (Novelita Dwi Miranda, 2020).

2.3.3 Fully Connected Layer

Feature map atau activation map yang dihasilkan oleh lapisan sebelumnya direpresentasikan sebagai array multidimensi. Untuk melanjutkan ke tahap lapisan yang terhubung sepenuhnya, peta fitur ini perlu diratakan atau dibentuk ulang. Proses perataan ini mengubah array multidimensi menjadi format vektor, yang berfungsi sebagai masukan ke lapisan yang terhubung sepenuhnya. Lapisan yang terhubung sepenuhnya ini terdiri dari berbagai hidden layer, action function, output layer, dan loss function. Mereka bertanggung jawab untuk membangun hubungan antara setiap neuron yang dihasilkan oleh lapisan sebelumnya dan neuron di lapisan berikutnya (Ari Peryanto, 2019). Anda dapat mengamati struktur lapisan yang terhubung penuh pada Gambar 2.6.



Gambar 2. 6 Fully Connected Layer

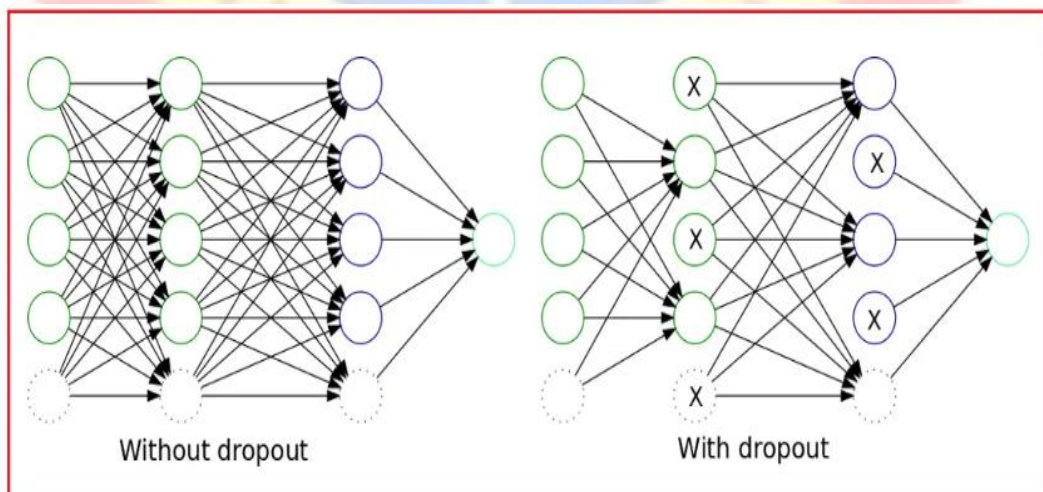
(Ari Peryanto, 2019)

2.3.4 Dropout

Dropout adalah teknik regularisasi yang digunakan dalam jaringan saraf, dimana neuron yang dipilih secara acak diabaikan selama pelatihan. Neuron yang diabaikan ini pada dasarnya "keluar" atau dikeluarkan dari jaringan untuk

sementara. Proses ini melibatkan pencegahan kontribusi neuron-neuron ini selama forward pass, dan bobotnya yang sesuai tidak diperbarui selama backpropagation.

Dropout memiliki dua tujuan utama, pertama membantu mencegah overfitting dengan menimbulkan noise dan redundansi, dan juga dapat mempercepat proses pembelajaran dengan mengurangi saling ketergantungan antar neuron. Neuron dapat dikeluarkan dari lapisan tersembunyi atau lapisan terlihat dalam jaringan. Pemilihan neuron yang akan dikeluarkan dilakukan secara acak, dan setiap neuron diberi nilai probabilitas antara nol dan satu. Probabilitas ini menentukan kemungkinan neuron putus selama setiap iterasi pelatihan. Secara acak dan tidak dipakai selama pelatihan. Neuron-neuron ini dapat dikatakan dibuang secara acak. Hal ini berarti bahwa kontribusi neuron yang dibuang akan dihentikan sementara jaringan dan bobot baru juga tidak diterapkan pada neuron pada saat melakukan backpropagation. Dropout merupakan proses mencegah terjadinya overfitting dan juga mempercepat proses belajar. Keluar mengacu pada menghilangkan neuron yang berupa lapisan maupun tersembunyi yang terlihat di dalam jaringan. Dengan menghilangkan suatu neuron, berarti menghilangkannya sementara dari jaringan yang ada. Neuron yang akan dihilangkan akan dipilih secara acak. Setiap neuron akan diberikan probabilitas yang bernilai antara nol dan satu (Ari Peryanto, 2019).



Gambar 2.7 Dropout

(Ari Peryanto, 2019)

2.3.5 Zero Padding

Zero padding adalah Teknik yang melibatkan tambahan nilai 0 sehingga dimensi *matriks citra* menjadi lebih besar (E Sentosa, H Armanto, P Pickerling, 2022). Gambar 2.4 mengilustrasikan proses *zero padding* yang mengubah dimensi matriks citra dari 6x6 menjadi 8x8.

0	0	0	0	0	0	0	0
0	3	3	4	4	7	0	0
0	9	7	6	5	8	2	0
0	6	5	5	6	9	2	0
0	7	1	3	2	7	8	0
0	0	3	7	1	8	3	0
0	4	0	4	3	2	2	0
0	0	0	0	0	0	0	0

Gambar 2.8 Proses Zero Padding

Zero padding adalah proses tambahan 1 piksel pada gambar dengan jumlah piksel 0. Tujuan dari proses ini adalah untuk meningkatkan dimensi gambar pada setiap channel warna, sehingga meningkatkan dimensi matriks citra (Veeramachaneni, 2023).

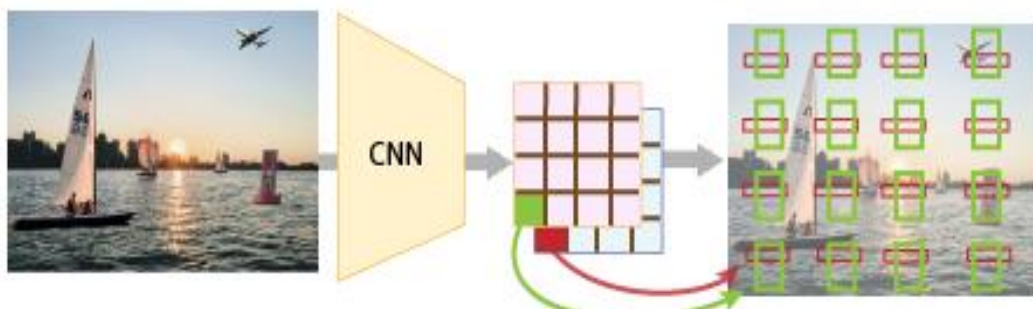
2.3.6 ReLU Activation

ReLU (Rectified Linear Unit) adalah jenis fungsi linear yang mengubah nilai x menjadi 0 ketika x bernilai negatif, sementara nilai x dipertahankan jika x bernilai positif (Agarap, 2018). Selain itu, Persamaan 2.1 dirumuskan dari *ReLU Activation*.

$$f(x) = \max(0, x) \quad (2.1)$$

2.4 Anchor Box

Anchor box merupakan sejumlah pengotakan pembatas yang ditetapkan bersama skala dan rasio tertentu. *Anchor box* digunakan untuk memprediksi skala dan rasio objek yang akan dideteksi, sering kali dipilih berdasarkan karakteristik ukuran objek (Ariel_一只猫の旅行, 2020). Pada Gambar 2.5 *Anchor Box* digunakan untuk menetapkan skala dan rasio pada objek yang akan dideteksi.



Gambar 2.9 Anchor Box

(Ariel_一只猫の旅行, 2020)

Translation-invariant, seperti pada varian RPN pada jaringan FPN digunakan pada bagian anchor. Area dari anchor tersebut berukuran mulai dari 32^2 hingga 512^2 . Anchor box membantu mengidentifikasi objek dengan memperhitungkan variasi skala dan bentuk dalam gambar.

2.5 Classification Subnet

Classification subnet merupakan proses memprediksi probabilitas keberadaan objek untuk setiap *anchor* A dan *class* objek K pada posisi spasial. Setiap level Feature Pyramid Network memiliki jaringan subnet yang sangat berhubungan. Parameter pada subnet ini dibagi pada semua tingkatan piramida. Pengambilan input *feature map* bersamaan dengan *channel citra* dari level piramida yang diberikan, subnet ini menerapkan 4 konvolusi 3×3 , masing-masing dengan *filter* C dan diikuti oleh fungsi *aktivasi ReLU*, kemudian di diikuti oleh kedalam *konvolusi* 3×3 . Langkah terakhirnya adalah penerapan *aktivasi sigmoid* pada output hasil prediksi biner dari K A per lokasi *spasial*. Pada mayoritas penelitian nilai *citra* yang digunakan sebesar 256 dan A sebesar 9 (Tsung-Yi Lin, 2017).

2.6 Box Regression Subnet

Box Regression Subnet (BRS) adalah set kotak terhadap objek yang telah di deteksi. Box Regression Subnet juga mock-up pergeseran *offset* dari setiap *anchor* ke objek *ground-truth* terdekat (MathWorks, 2020). Jaringan subnet ini mirip dengan *classification subnet*. Namun, ada sedikit perbedaan pada tahap akhir, diakhiri oleh output linear 4A pada lokasi spasial. Hal tersebut dapat dilihat pada 2.3, untuk setiap *anchor* A per lokasi *spasial*, keempat output ini memprediksi *relative offset* diantara *anchor* dan *ground-truth box*. Struktur pada *subnet classification* dan *box regression* menggunakan struktur yang sama tetapi parameter yang digunakan tetap terpisah.

2.7 Focal Loss

Focal Loss digunakan untuk mengatasi ketidakseimbangan ekstrem antara sample objek (foreground) dan sample bukan objek (background) selama pelatihan. Focal loss dimulai dari kehilangan cross entropy (CE) pada klasifikasi biner digunakan untuk mengatasi masalah. (Tsung-Yi Lin, 2017).

Dikarenakan terdapat ketidak seimbangan yang disebabkan class dalam jumlah yang besar selama proses training dari *dense detector* membanjiri *cross entropy loss*. Hasil klasifikasi terdiri dari nilai *loss* yang menjadi mayoritas dan mendominasi *gradient*. Oleh karena itu ditambahkan sebuah *modulasi* $(1-pt)^\gamma$ pada fungsi *cross entropy loss* sehingga disebut *focal loss*, yang di tujuan pada Persamaan 2.2 sebagai berikut:

$$FL(pt) = -\alpha \cdot (1 - pt)^\gamma \cdot \log(pt) \quad (2.2)$$

Di mana nilai γ yang paling optimal adalah $\gamma = 2$ dan $\alpha = 0,25$.

2.8 Training Data

Pelatihan data dapat dikategorikan data berlabel dan data tidak berlabel. Data berlabel mengacu pada sekumpulan titik data yang ditandai dengan satu atau lebih label atau anotasi yang bermakna. Misalnya, dalam konteks gambar yang menggambarkan pengemudi mengenakan sabuk pengaman, labelnya dapat berupa "Mengenakan Sabuk Pengaman (MSP)" dan "Tidak Mengenakan Sabuk Pengaman (TMSP)". Di sisi lain, data tak berlabel mengacu pada titik data yang tidak memiliki

label atau anotasi terkait. Dengan menggunakan contoh yang sama, data yang tidak berlabel berarti gambar penggunaan sabuk pengaman tanpa penjelasan khusus apa pun.

Training set adalah bagian dari kumpulan data yang digunakan untuk melatih algoritme pembelajaran mesin guna membuat deteksi atau melakukan tugas tertentu sesuai dengan tujuannya. Kami menginstruksikan algoritme agar mesin yang kami latih dapat menemukan korelasi dalam data secara mandiri.

2.9 Preprocessing Data

Pemrosesan awal data adalah proses penting di mana data mentah diubah menjadi format yang sesuai untuk pelatihan dan pengujian model ResNet-50. Proses ini melibatkan beberapa tahap:

1. Tahap pertama, Pelabelan Data, melibatkan pengorganisasian gambar dalam kumpulan data ke dalam folder terpisah berdasarkan klasifikasi atau kategori seperti yang ditentukan yang menyertai kumpulan data tersebut.
2. Tahap kedua, Segmentasi Data, berfokus pada isolasi gambar daun dari lingkungan sekitarnya untuk memastikan model hanya berkonsentrasi pada daun itu sendiri. Segmentasi ini memungkinkan model mengklasifikasikan setiap gambar daun secara akurat.
3. Tahap ketiga, Augmentasi Data, melibatkan duplikasi data dengan menerapkan transformasi seperti rotasi, refleksi, dan pembesaran pada gambar. Tujuan utama augmentasi data adalah untuk menyeimbangkan kumpulan data, khususnya dalam kasus kelas jamak, sehingga meningkatkan performa model.
4. Tahap keempat dan terakhir, Pengubahan Ukuran Data, melibatkan pengubahan ukuran semua data agar sesuai dengan ukuran input yang diperlukan oleh model ResNet-50. Langkah ini memastikan keseragaman dimensi data, memfasilitasi pemrosesan model yang efisien.

2.10 Deteksi Objek

Deteksi objek bertujuan untuk memisahkan objek atau latar depan dari latar belakang pada suatu gambar. Biasanya, untuk melakukan deteksi objek, model latar

belakang dibuat untuk gambar bergerak berdasarkan nilai piksel dari waktu ke waktu. Untuk mendeteksi suatu objek secara efektif dan efisien, perlu disediakan informasi seperti untuk mendeteksi gambar dan gambar digital. Ketika melihat suatu gambar, otak manusia dapat langsung mengenali objek dalam gambar tersebut, lokasinya, dan kondisi interaksinya. Sistem visual dan akurat, memungkinkan kita melakukan tugas-tugas kompleks. Algoritme deteksi objek yang cepat dan akurat akan memungkinkan komputer melakukan tugas serupa dan berpotensi menyelesaikan tugas secara umum.

Deteksi objek pada pengolahan citra digital merupakan suatu proses yang digunakan untuk mengetahui keberadaan suatu objek tertentu pada suatu gambar. Proses pendeteksian ini dapat dilakukan dengan menggunakan berbagai metode yang biasanya membaca fitur dari seluruh objek pada gambar masukan. Ciri-ciri objek pada gambar masukan kemudian dibandingkan dengan ciri-ciri model atau template yang digunakan. Hasil perbandingan tersebut dapat digunakan untuk mengetahui apakah suatu objek terdeteksi sebagai template yang dimaksud atau tidak. Sistem deteksi objek perlu melatih dan menguji kumpulan data dengan kotak pembatas dan diberi label untuk setiap kelas objek untuk proses pengenalan.

2.11 Jaringan Residual Network (*ResNet-50*)

Resnet-50 adalah jaringan *residual network* dengan 50 lapisan. Resnet adalah arsitektur jaringan saraf konvolusional (CNN) yang dikembangkan oleh Microsoft Research. Ini terkenal karena menggunakan blok residual, untuk melatih jaringan yang lebih dalam tanpa mengurangi kinerja. Koneksi sisa ini memungkinkan jaringan untuk mempelajari arsitektur yang jauh lebih dalam daripada yang mungkin dilakukan sebelumnya, tanpa mengalami masalah hilangnya gradien.

Arsitektur ResNet50 dibagi menjadi empat bagian utama: lapisan konvolusional, blok identitas, blok konvolusional, dan lapisan terhubung penuh. Lapisan konvolusional bertanggung jawab untuk mengekstraksi fitur dari gambar masukan, sedangkan blok identitas dan blok konvolusional bertanggung jawab untuk memproses dan mengubah fitur tersebut. Terakhir, lapisan yang terhubung sepenuhnya digunakan untuk membuat klasifikasi akhir. Lapisan konvolusional di

ResNet50 terdiri dari beberapa lapisan konvolusional yang diikuti dengan normalisasi batch dan aktivasi ReLU. Lapisan ini bertanggung jawab untuk mengekstraksi fitur-fitur dari gambar masukan, seperti tepi, tekstur, dan bentuk. Lapisan konvolusional diikuti oleh lapisan penggabungan maksimal, yang mengurangi dimensi spasial peta fitur sambil mempertahankan fitur yang paling penting. Blok identitas dan blok konvolusional adalah blok penyusun utama ResNet50. Blok identitas adalah blok sederhana yang meneruskan masukan melalui serangkaian lapisan konvolusional dan menambahkan masukan kembali ke keluaran (Kundu, 2023).

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Gambar 2.10 Arsitektur dan Jumlah Lapisan Resnet

(Atheadeth, n.d.)

Pada Gambar 2.6 ada beberapa arsitektur *Resnet*, layer pertama yaitu 18 kedalaman konvolusi, layer kedua 34 kedalaman konvolusi, layer ketiga 50 kedalaman konvolusi, layer keempat 101 kedalaman konvolusi, dan layer kelima 152 kedalaman konvolusi. Proses yang dilakukan terdiri dari operasi konvolusi 7x7, max pooling 3x3, konvolusi 1x1, aktivasi ReLU, operasi konvolusi 3x3, aktivasi ReLU, operasi konvolusi 1x1, Jumlah filter yang digunakan pada setiap operasi konvolusi pada modul residual disesuaikan dengan tingkatan layer residual network yang digunakan. Arsitektur dan jumlah lapisan *residual network* dapat dilihat pada Gambar 2.6.





Pada tahun 2015, Arsitektur Residual Network (ResNet) dengan jumlah layer sebesar 152 berhasil memenangkan juara pertama pada kompetisi *ImageNet* dalam hal deteksi, klasifikasi dan *segmentasi* gambar (He, 2015).

Model ResNet-50 yang digunakan untuk eksperimen ini terdiri dari 48 lapisan konvolusional, serta lapisan MaxPool dan Average Pool (48+1+1=50 lapisan). Dengan struktur jaringan yang lebih dalam, tingkat deteksi yang lebih baik dapat dicapai dibandingkan dengan struktur jaringan datar yang digunakan sebelumnya.

Versi model ResNet yang telah dilatih sebelumnya dengan kumpulan data ImageNet dapat diunduh dari perpustakaan PyTorch. Namun, kami menggunakan model ResNet50 yang tidak terlatih karena kami ingin menyelidiki pengoptimalan pelatihan dengan ImageNet.

2.12 Pengujian Kinerja Sistem

Confusion matrix juga dapat digunakan untuk mengukur kinerja metode klasifikasi dengan membandingkan hasil klasifikasi sistem untuk mengetahui seberapa akuratnya. Itu juga dapat digunakan untuk melakukan visualisasi terhadap hasil pembelajaran sistem, yang menampilkan dua kategori atau lebih. Pengujian kinerja untuk menguji deteksi objek, digunakan untuk menghitung *precision*, *recall*, *f1 score* dan akurasi (Karlita, et al., 2019). Recall adalah tingkat keberhasilan sistem dalam menemukan informasi, Sedangkan precision adalah tingkat ketepatan antara informasi dan jawaban yang diberikan oleh sistem. Akurasi adalah tingkat kedekatan antara nilai aktual dan prediksi. skor F1 digunakan sebagai metrik kinerja, untuk tugas apa pun yang kebutuhan skor True Positive yang tinggi. Gambar 2.7 menjelaskan tentang confusion matrix yang akan digunakan dalam pengujian sistem.

		PREDICTED VALUES	
		Positive (CAT)	Negative (DOG)
ACTUAL VALUES	Positive (CAT)	 <p>TRUE POSITIVE 6 YOU ARE A CAT</p>	 <p>FALSE NEGATIVE 1 TYPE II ERROR YOU ARE A DOG</p>
	Negative (DOG)	 <p>FALSE POSITIVE 2 YOU ARE A CAT TYPE I ERROR</p>	 <p>TRUE NEGATIVE 11 YOU ARE NOT A CAT</p>

Gambar 2.11 Confusion Matrix

(Rina, 2023)

Berdasarkan Gambar 2.7 Confusion Matrix di atas adalah:

1. **True Positive (TP)** merupakan 6 yang berarti bahwa 6 kali model kami berhasil menebak dengan benar bahwa gambar itu menunjukkan kucing.
2. **True Negative (TN)** merupakan 11 yang berarti bahwa 11 kali model kami berhasil menebak dengan benar bahwa gambar tersebut tidak menunjukkan kucing atau anjing, dan bahwa gambar tersebut sebenarnya menunjukkan anjing.
3. **False Positif (FP)** merupakan 2 yang berarti ada 2 kali model kami salah mengira gambar itu menunjukkan anjing meskipun sebenarnya itu menunjukkan kucing.
4. **False Negatif (FN)** merupakan 1 yang berarti bahwa ada satu kali model kita salah menduga bahwa gambar tersebut tidak menampilkan kucing atau anjing, meskipun sebenarnya menampilkan kucing. Perhitungan yang dilakukan oleh confusion matrix berdasarkan Gambar 2.7 dapat menggunakan rumusan sebagai berikut:

$$Precision = \frac{TP}{TP + FP} * 100\%$$

$$Recall = \frac{TP}{TP + FN} * 100\%$$

$$F1\ Score = 2 * \frac{Precision * recall}{Precision + recall}$$

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

2.10 Python

Python adalah bahasa pemrograman tujuan umum yang biasa digunakan untuk membangun situs web, perangkat lunak/aplikasi, mengotomatiskan tugas, dan melakukan analisis data. Ia dikenal karena keserbagunaan dan kemudahan penggunaannya, menjadikannya salah satu bahasa pemrograman yang paling banyak digunakan. Python sangat populer di kalangan pemula karena sintaksisnya yang ramah bagi pemula dan dukungan komunitas yang luas. Sebagai bahasa umum, Python dapat digunakan untuk membuat berbagai macam program, tidak terbatas pada domain atau aplikasi tertentu. Penggunaannya yang luas berasal dari kemampuannya untuk mengatasi beragam kebutuhan pemrograman secara efisien. Dalam hal popularitas, Python menempati peringkat keempat bahasa pemrograman terpopuler, dengan hampir 50% responden menyatakan bahwa mereka menggunakannya hampir separuh waktu kerja mereka. Nama "Python" sendiri diambil dari Monty Python, grup komedi asal Inggris. Guido van Rossum, pencipta Python, terinspirasi oleh grup tersebut saat mengembangkan bahasanya dan menganggap nama tersebut pendek dan sedikit misterius. Oleh karena itu, ia memilih menggunakan nama tersebut untuk bahasa pemrograman yang ia buat.

Python biasa dipakai dalam pengembangan perangkat lunak, membuat analisis data, visualisasi data dan otomatisasi tugas. Karena sifatnya yang relatif mudah dipelajari, bahasa pemrograman ini digunakan secara luas oleh non-programmer seperti ilmuwan dan akuntan untuk melakukan tugas harian mereka. Misalnya, dalam mengatur keuangan.



Gambar 2. 12 Python

2.10.1 Tensorflow

TensorFlow adalah kerangka kerja sumber terbuka yang dikembangkan oleh Google. Ini digunakan untuk mengembangkan dan melatih berbagai model pembelajaran mesin dan pembelajaran mendalam, serta tugas lain yang terkait dengan analisis statistik. TensorFlow adalah salah satu perpustakaan paling populer dan banyak digunakan di bidang ini. Ini digunakan untuk mengembangkan dan mengimplementasikan algoritma pembelajaran mesin dan algoritma lain yang melibatkan banyak operasi.

Tanpa sepengetahuan banyak orang, TensorFlow sebenarnya dimanfaatkan dalam kehidupan sehari-hari. Misalnya, aplikasi seperti Google Foto atau Google Voice secara tidak langsung menggunakan model TensorFlow. Model ini beroperasi pada infrastruktur perangkat keras Google yang luas dan sangat efisien dalam tugas-tugas persepsi.

Tujuan utama TensorFlow adalah untuk menyederhanakan proses pengembangan analisis tingkat lanjut bagi banyak pihak. Berikut beberapa contoh penerapan TensorFlow di berbagai bidang, seperti pengenalan gambar, suara, dan video, termasuk sistem yang diterapkan pada penggunaan sabuk pengaman pada mobil.

2.11 Website

Situs web berfungsi sebagai tempat penyimpanan pusat untuk halaman web, yang masing-masing berisi konten atau informasi yang relevan dengan tujuan situs. Beranda bertindak sebagai gerbang utama, menghubungkan pengguna ke berbagai bagian dan konten dalam situs. URL (Uniform Resource Locator) berfungsi sebagai alamat unik sebuah situs web.

Situs web biasanya menggunakan format file tunggal yang dikenal sebagai hypertext, khususnya HTML (Hypertext Markup Language). File HTML terdiri dari karakter ASCII dan menggabungkan perintah dengan konten dokumen. Di dalam file HTML, di samping perintah penataan gaya, terdapat instruksi untuk menautkan satu dokumen HTML ke dokumen lain menggunakan kata kunci sebagai penunjuk atau tautan. Tautan ini dapat mengarah ke lokasi file lokal, file lain dalam situs web yang sama, atau file yang dihosting di server eksternal. Menerapkan tautan dalam file HTML memfasilitasi akses lebih cepat ke informasi terkait sesuai dengan kata kunci tertentu.

2.13 UML (Unified Modeling Language)

Unified Modeling Language (UML) adalah bahasa pemodelan yang terutama digunakan untuk pengembangan sistem atau perangkat lunak dengan pendekatan berorientasi objek. UML mengabstraksi konsep dasar ke dalam klasifikasi struktural, perilaku dinamis, dan model manajemen. Konsep utama dalam UML berfungsi sebagai istilah yang sering ditemui saat membuat diagram, sedangkan tampilan mengategorikan diagram.

UML mencakup berbagai jenis diagram, masing-masing memiliki tujuan berbeda:

- 1 Use Case Diagram: Diagram ini mewakili tampilan eksternal dari sistem yang dimodelkan. Ini menggambarkan interaksi antara aktor dan sistem, dengan fokus pada menghasilkan nilai yang terukur. Meskipun diagram kasus penggunaan menggambarkan kasus penggunaan, penting untuk dicatat bahwa diagram hanyalah representasi visual dan bukan keseluruhan model itu sendiri.
- 2 Diagram Kelas: Kelas dalam UML mengacu pada kumpulan objek dengan atribut dan perilaku serupa. Sebuah kelas mencakup namanya, atribut (karakteristik yang terkait dengan kelas), dan operasi (tindakan yang dapat dilakukan oleh kelas). Diagram kelas secara visual mewakili hubungan dan struktur dalam suatu sistem.
- 3 Diagram Aktivitas: Diagram aktivitas menggambarkan alur kerja atau perilaku suatu sistem melalui serangkaian tindakan dan keputusan. Mereka menggambarkan bagaimana tindakan dimulai, keputusan yang mungkin terjadi, dan hasil akhirnya. Diagram aktivitas juga dapat menggambarkan aktivitas paralel dan melibatkan elemen seperti aktivitas, objek, keadaan, transisi keadaan, dan peristiwa.
- 4 Sequence diagram
Secara mudahnya sequence diagram adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logistik yang seharusnya dilakukan untuk menghasilkan sesuatu yang sesuai dengan use case diagram

Secara keseluruhan, UML menyediakan bahasa grafis untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan sistem perangkat lunak berorientasi objek. Ini menetapkan standar untuk membuat cetak biru sistem, yang mencakup berbagai aspek seperti proses bisnis, definisi kelas dalam bahasa pemrograman tertentu, skema database, dan komponen perangkat lunak. Diagram UML berfungsi sebagai alat yang efektif untuk komunikasi dan

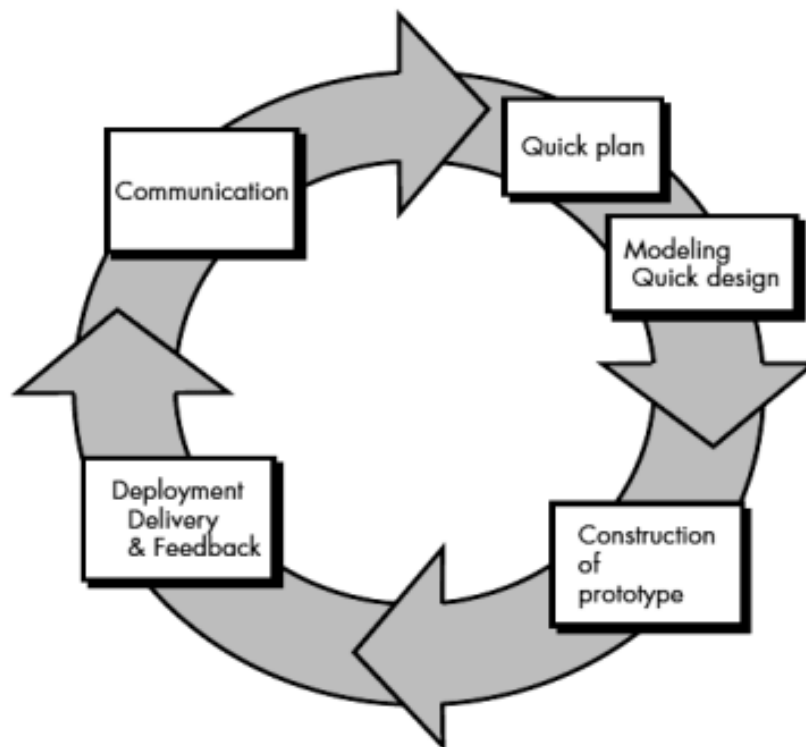
pemahaman antar pemangku kepentingan yang terlibat dalam proses pengembangan perangkat lunak.



BAB III

METODE PENELITIAN

Pada bab ini menjelaskan mengenai perancangan sistem dengan menerapkan metode pembangunan model sistem *prototype*. *Prototype* ini bertujuan untuk mengidentifikasi jalannya suatu sistem dan digunakan karena dapat memecahkan permasalahan-permasalahan dalam waktu relatif yang singkat. Model *prototype* terdiri dari beberapa tahap yang harus dilakukan dan setiap tahap memiliki aktivitas-aktivitas tersendiri seperti *Communication*, *Quick plan*, *Modelling Quick Desain*, pembentukan *Prototype* dari *Deployment Delivery & feedback*. Gambar 3.1 merupakan gambaran diagram dari model *prototype*:



Gambar 3.1 Model Prototype

(Pressman, 2010)

3.1 Analisis dan Perancangan

Pada tahap analisis dilakukan untuk menguraikan secara mendalam terhadap perancangan yang akan digunakan. Pada perancangan aplikasi deteksi penggunaan pengemudi yang memakai sabuk pengaman pada mobil juga dilakukan analisis.

3.1.1 Studi Literatur

Merupakan tahapan awal dalam metode pengumpulan data dan tahapan untuk memahami konsep dari implementasi metode resnet-50 untuk deteksi penggunaan sabuk pengaman pada pengemudi mobil. Pengumpulan data dan pemahaman diperoleh dari artikel, jurnal, buku dan sumber informasi dari internet yang mendukung proses penulisan.

3.1.2 Objek Penelitian

Data yang digunakan dari penelitian tugas akhir ini adalah data yang langsung berkaitan dengan objek penelitian, dalam hal ini data diperoleh melalui pengambilan data langsung di lapangan. Dalam penelitian ini yang menjadi objek penelitian adalah deteksi sabuk pengaman di kota Bandung.

3.1.3 Pengumpulan Data

Penelitian ini memanfaatkan rekaman video yang diambil dari kamera ponsel secara manual. Video-video ini kemudian diproses untuk membuat kumpulan data, yang nantinya akan berfungsi sebagai data pelatihan dan pengujian. Setiap objek dalam video diberi label untuk memfasilitasi proses pengujian.

3.1.4 Analisis Kebutuhan

Mengikuti metodologi pengembangan perangkat lunak yang digunakan dalam tugas akhir ini, tahap awal melibatkan analisis kebutuhan yang diperlukan. Analisis kebutuhan berfungsi sebagai langkah dasar untuk menentukan aplikasi yang akan dikembangkan. Untuk aplikasi deteksi penggunaan sabuk pengaman pada mobil ini, banyak alat untuk perancangan membuat sistem deteksi sabuk pengaman. Dalam perancangan aplikasi mendeteksi sabuk pengaman ini adalah perangkat lunak yang sesuai untuk pengembangan aplikasi. Selain kebutuhan perangkat lunak, perangkat keras juga penting untuk perancangan aplikasi deteksi

wajah ini. Berikut ini merupakan spesifikasi minimum *hardware* berdasarkan buku “*PengantarMachine Learning*” (Yunus, 2020), yang digunakan dalam membangun sistem:

1. Perangkat Keras:

- A. *Processor Intel Core i5-3217U 3.0 GHz*
- B. RAM 4 GB DDR4
- C. Penyimpanan 1 TB HDD
- D. VGA NVIDIA GPU Card

2. Perangkat Lunak

- A. Windows 10 Professional 64-bit dibutuhkan sebagai sistem operasi pada PC atau laptop yang digunakan untuk menuliskan dokumentasi laporan penelitian.
- B. JetBrains Pycharm Community Edition merupakan sebuah platform Bahasa pemrograman python yang digunakan untuk membuat kode proses back-end pada sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil.

3.2 Tempat dan Waktu Penelitian

Penelitian ini bertujuan untuk mendapat gambaran dan informasi yang lebih jelas, lengkap, serta memungkinkan dan mudah bagi peneliti untuk melakukan penelitian. Oleh karena itu, maka penulis menetapkan lokasi penelitian adalah tempat di mana penelitian akan dilakukan. Dalam hal ini, lokasi penelitian terletak di jalan Soekarno Hatta dan Gasibu Bandung.

3.3 Dataset

Dataset yang digunakan berisi 5.300 citra objek pengemudi mobil yang memakai sabuk pengaman dan 5.300 pengemudi mobil yang tidak memakai sabuk. Pengambilan dataset pengemudi diambil pada pemberhentian lampu lalu lintas di jalan Gasibu dan di jalan Soekarno Hatta Bandung. Proses pengumpulan data dilakukan pada pagi hari dan sore hari, dengan pengambilan citra objek pengemudi pada jarak 1 sampai 2 meter. Pada Gambar 3.2 merupakan contoh dataset objek pengemudi yang menggunakan sabuk pengaman dan pada Gambar

3.3 merupakan contoh dataset objek pengemudi yang tidak menggunakan sabuk pengaman.



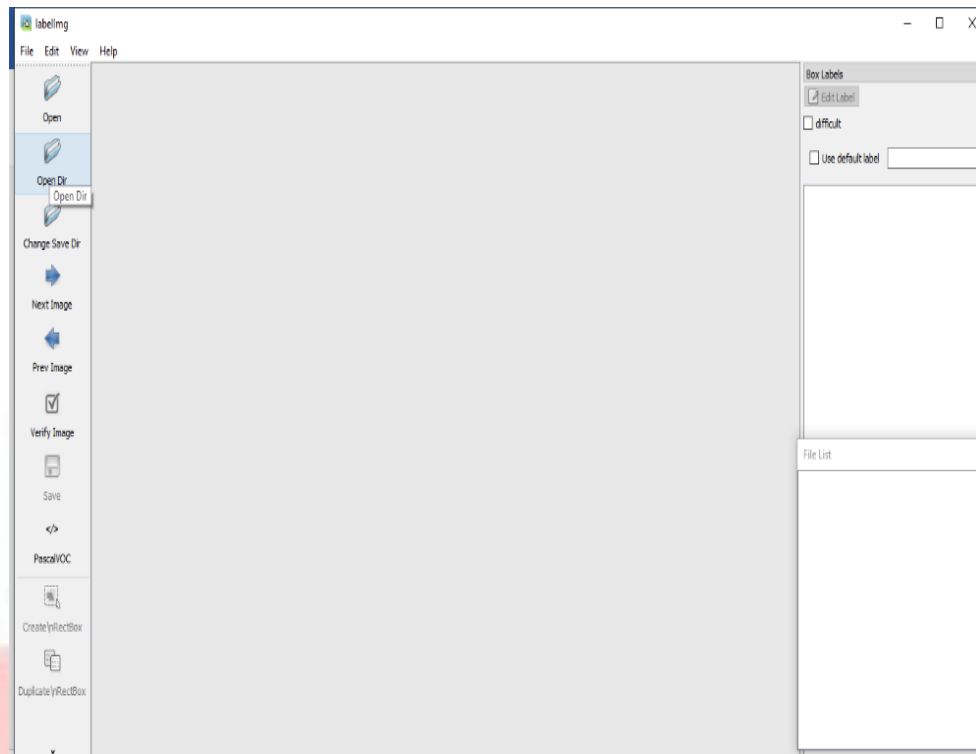
Gambar 3.2 Pengemudi yang tidak menggunakan Sabuk Pengaman



Gambar 3.3 Pengemudi yang tidak menggunakan Sabuk Pengaman

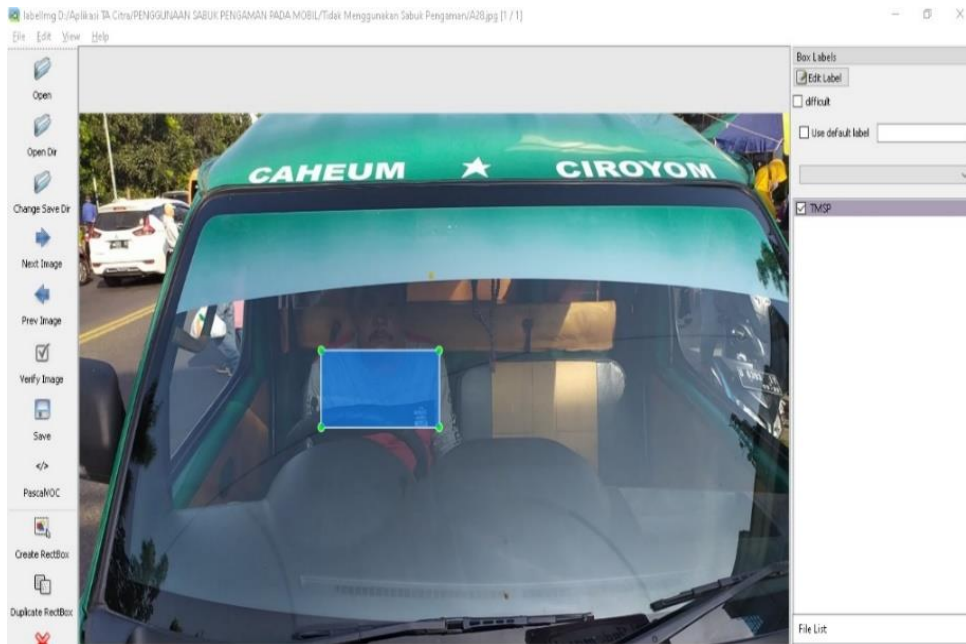
Pada dataset yang belum diberi label dilakukan pemberian label dengan menggunakan LabelImg dengan format Pascal VOC yang labelnya disimpan dalam file berekstensi .xml. berikut adalah tahapan- tahapan dalam melakukan proses labeling.

1. Membuka Anaconda Prompt.
2. Memberikan LabelImg pada pengemudi yang menggunakan sabuk pengaman dan tidak menggunakan sabuk pengaman.
3. Mengaktifkan tombol open untuk membuka direktori tempat penyimpanan file dataset. Pada Gambar 3.4 merupakan tampilan aplikasi labeling.

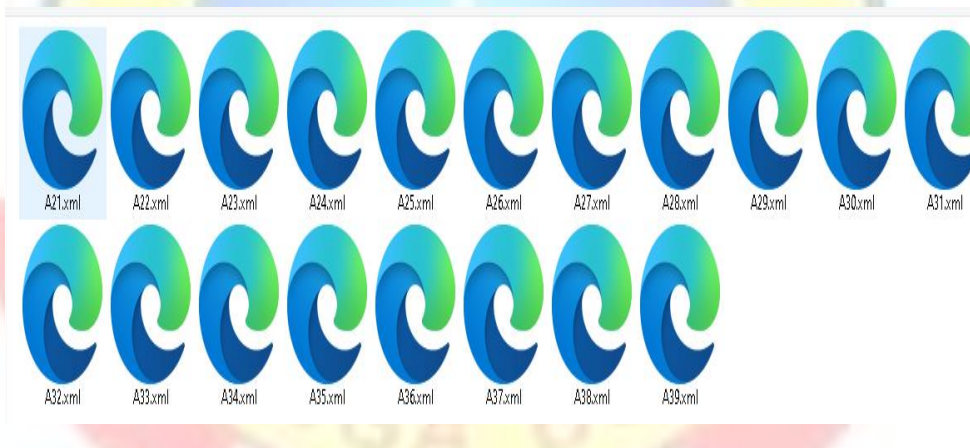


Gambar 3.2 Tampilan LabelImg

4. Memilih folder tempat penyimpanan pada dataset. Mengaktifkan tombol Create \nrectBox untuk membuat bounding box dan label pada objek. Pada Gambar 3.5 hasil pembentukan bounding box dan memberikan label pada pengemudi yang tidak menggunakan sabuk pengaman.
5. Kemudian Memilih verify image untuk menyimpan data hasil pembuatan label pada dataset. Pada Gambar 3.6 data hasil pembuatan label berekstensi *.xml



Gambar 3.5 Hasil Dataset pengemudi tidak menggunakan Sabuk



Gambar 3.3 Data Hasil pembuatan LabelImg

3.4 Data Preprocessing

Data dilakukan untuk memastikan data dimasukkan ke dalam model. Ini melibatkan normalisasi data dengan menyusutkan nilai piksel gambar ke dalam rentang [0, 1], resizing gambar, dan penerapan augmentasi data seperti rotasi, pergeseran, dan pembalikan horizontal. Modul TensorFlow dan Keras diimpor, dan jika model sebelumnya sudah ada, file tersebut dihapus. Selanjutnya, folder root

yang berisi pengemudi sabuk pengaman ditentukan, dan dilakukan pra-pemrosesan data.

1. ImageDataGenerator

Penyesuaian seperti `rescale`, `shear_range`, `zoom_range`, dan `horizontal_flip`. Ukuran batch dan dimensi gambar juga diatur, kemudian generator digunakan untuk memuat data gambar dari direktori berdasarkan label folder, dengan mode warna diubah menjadi skala abu-abu.

2. Grayscale

Setelah data siap, model CNN dibuat dengan menentukan arsitektur yang sesuai. Jumlah lapisan konvolusi, lapisan pooling, dan lapisan fully connected ditentukan, serta fungsi aktivasi untuk masing-masing lapisan. Model diinisialisasi menggunakan framework deep learning seperti TensorFlow atau Keras.

3. RGB

Untuk citra berwarna digunakan model RGB (Red-Green-Blue), satu citra berwarna dinyatakan dalam 3 matriks skala abu-abu berupa matriks untuk Merah (R-layer), satu matriks Hijau (G-layer) dan matriks untuk Biru (B-layer). R-layer adalah matriks yang menyatakan derajat kecerahan warna merah dengan skala abu-abu 0-255, nilai 0 mewakili gelap (hitam) dan 255 menyatakan merah. G-layer adalah matriks yang menyatakan derajat kecerahan untuk warna hijau, dan B-layer merupakan matriks yang mewakili derajat kecerahan untuk warna biru. Dari definisi tersebut, untuk menyajikan suatu warna tertentu dapat dilakukan dengan mudah, yaitu dengan mencampurkan ketiga warna dasar RGB.

3.5 Skenario Penelitian

Skenario penelitian ini melibatkan pengambilan video menggunakan kamera ponsel secara manual. Video-video tersebut kemudian akan diproses untuk membuat kumpulan data, yang akan digunakan sebagai data latih dan uji. Setiap objek dalam video akan diberi label untuk memfasilitasi proses pelatihan dan evaluasi model. Tujuan dari penelitian ini mungkin adalah untuk mengembangkan

sistem pengenalan objek dalam video menggunakan pendekatan deep learning atau teknik pengolahan citra komputer. Metodologi yang dapat digunakan termasuk pelatihan model jaringan saraf konvolusional seperti *Resnet-50* untuk mengidentifikasi objek pada deteksi sabuk pengaman, serta penggunaan teknik seperti transfer learning untuk meningkatkan kinerja model dengan dataset yang terbatas. Evaluasi kemudian akan dilakukan menggunakan metrik kinerja seperti akurasi pengenalan objek.

3.6 Dataset Training dan Testing

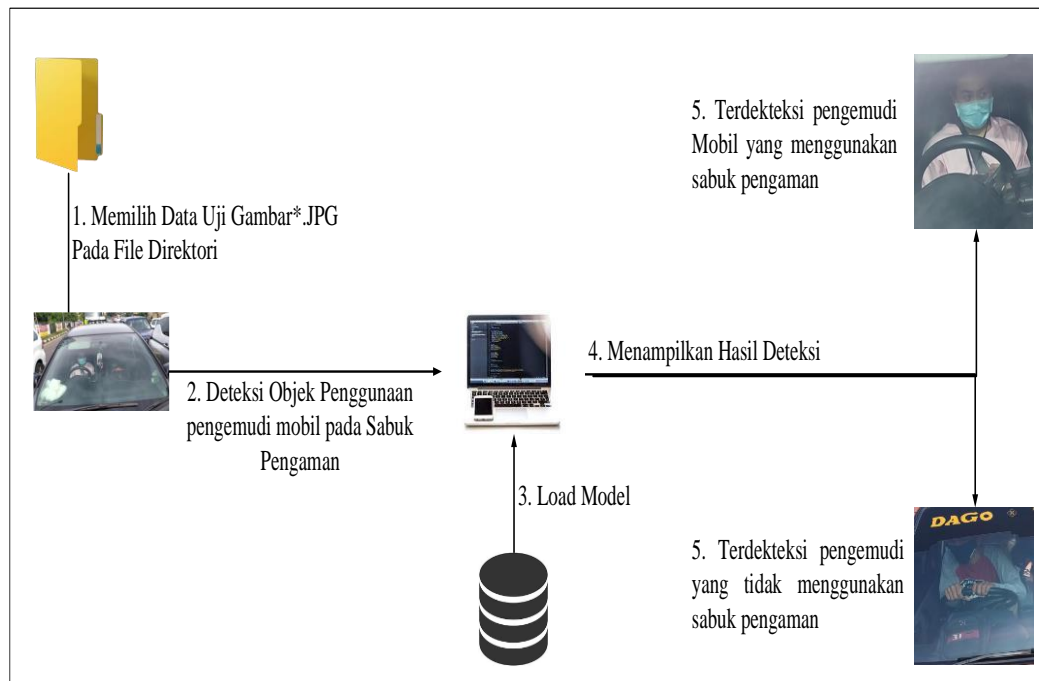
Dataset yang digunakan berisi 5.300 citra objek pengemudi mobil yang memakai sabuk pengaman dan 5.300 pengemudi mobil yang tidak memakai sabuk. Pengambilan dataset pengemudi diambil pada pemberhentian lampu lalu lintas di jalan Gasibu dan di jalan Soekarno Hatta Bandung.

Tabel 3. 1 Training dan Testing

Dataset	Training	Testing
Pengemudi memakai Sabuk Pengaman	5000 dataset	300 dataset
Pengemudi tidak memakai Sabuk Pengaman	5000 dataset	300 dataset
Total	10.000 Dataset	600 dataset

3.7 Permodelan Sistem

Pada tahap pemodelan sistem diilustrasikan alur kerja sistem dari mulai user memilih data uji dari direktori, sehingga sistem dapat melakukan deteksi terhadap penggunaan sabuk pengaman pada pengemudi mobil. Pada Gambar 3.7 diilustrasikan gambaran kerja sistem secara keseluruhan.



Gambar 3.4 Permodelan Sistem

Pada Gambar 3.7 permodelan sistem terdapat 5 langkah proses yang dilakukan sistem yang dibangun untuk mendeteksi penggunaan sabuk pengaman pada pengemudi mobil. Berikut ini merupakan proses tersebut:

1. Langkah pertama yang dilakukan adalah memilih file data uji berupa gambar pengemudi mobil pada direktori.
2. Langkah kedua adalah untuk deteksi objek penggunaan sabuk pengaman pada pengemudi mobil dengan menggunakan metode *ResNet-50*.
3. Langkah ketiga adalah load model hasil training yang berisi bobot dan bias dataset, dan dilakukan proses pencocokan bobot dan bias dari data uji dengan bobot dan bias dari dataset sehingga menghasilkan kelas objek yang dikenali berdasarkan dataset.
4. Langkah keempat adalah sistem menampilkan hasil deteksi objek berupa bounding box, dan label pada data uji.
5. Langkah kelima keluaran sistem menampilkan hasil deteksi pengemudi yang memakai sabuk pengaman atau pengemudi yang tidak memakai sabuk pengaman.

3.8 Convolutional Neural Network

Dalam proses Convolutional Neural Network (CNN), ada beberapa parameter dan teknik utama yang digunakan untuk memproses *feature map* dan filter masukan secara efisien. Salah satu parameter tersebut adalah zero-padding, yang melibatkan penambahan angka 0 di sekitar batas *feature maps* sebelum konvolusi. Hal ini membantu mencegah hilangnya informasi dan mempertahankan dimensi spasial *feature maps*.

Selain itu, parameter langkah menentukan jumlah pergeseran piksel saat menggeser filter di atas *feature maps* masukan selama konvolusi. Ini mengontrol seberapa banyak filter bergerak melintasi peta fitur pada setiap langkah, sehingga memengaruhi ukuran *feature maps*. Kemudian, fungsi aktivasi Rectified Linear Unit (ReLU) biasanya digunakan di lapisan konvolusional. Fungsi aktivasi ini, memastikan bahwa hanya nilai positif yang diteruskan, sehingga secara efektif nilai negatif menjadi angka 0. Ini membantu dalam menangkap pola dalam data masukan.

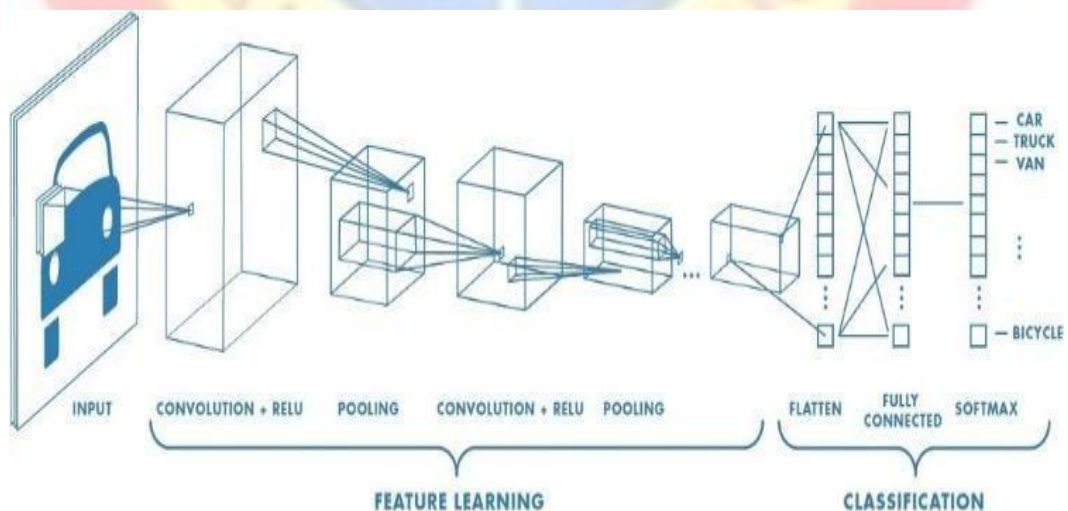
Pooling Layer adalah komponen kunci dalam *Convolutional Neural Network* (CNN) yang memainkan peran penting dalam mempercepat proses deteksi dengan mengurangi dimensi peta fitur yang dihasilkan oleh lapisan konvolusional. Hal ini dicapai dengan melakukan penghitungan pada matriks *feature maps*. Pada dasarnya ada dua jenis operasi pengumpulan yang dilakukan di lapisan ini: *max pooling* dan *average pooling*. *Max pooling* melibatkan pemilihan nilai maksimum dari setiap *feature maps*, yang secara efektif mempertahankan fitur paling menonjol yang terdeteksi oleh lapisan konvolusional. Di sisi lain, *average pooling* menghitung nilai rata-rata setiap *feature maps*.

Fully Connected Layer berfungsi sebagai tahap akhir dalam *Convolutional Neural Network* (CNN), semua proses sebelumnya diintegrasikan untuk melakukan komputasi setelah melintasi beberapa lapisan untuk pemetaan data. Lapisan ini bertanggung jawab untuk mengubah data dari *feature maps* yang diperoleh dari tahap sebelumnya menjadi format satu dimensi, yang sangat berguna untuk tugas klasifikasi.

Jadi, Fully Connected Layer menghubungkan setiap neuron di satu lapisan ke setiap neuron di lapisan berikutnya, memungkinkan pemrosesan data dan pengenalan pola yang komprehensif. Dengan mengubah *feature maps* menjadi satu dimensi, jaringan dipersiapkan untuk melakukan tugas klasifikasi dengan mengaitkan fitur tertentu dengan kelas yang berbeda. Lapisan ini bertindak sebagai jembatan antara tahapan ekstraksi fitur dan proses pengambilan keputusan klasifikasi akhir.

Dropout adalah untuk mencegah overfitting pada jaringan saraf dengan menonaktifkan sebagian neuron secara acak selama pelatihan. Hal ini membantu mengurangi ketergantungan pada neuron tertentu dan mendorong jaringan untuk mempelajari fitur yang lebih kuat. Fungsi Flatten diterapkan untuk mengubah *feature maps*. Transformasi ini sangat berguna untuk memasukkan *feature maps* ke dalam lapisan berikutnya yang terhubung sepenuhnya, memfasilitasi tugas classification di mana setiap kelas diwakili oleh neuron keluaran.

Softmax adalah untuk menghitung probabilitas setiap kelas mendapatkan klasifikasi yang benar berdasarkan keluaran. Ini menormalkan nilai keluaran ke kisaran antara 0 dan 1, memastikan bahwa nilai tersebut mewakili probabilitas yang valid. Hal ini mempermudah interpretasi keluaran jaringan dan menentukan prediksi kelas yang paling mungkin.



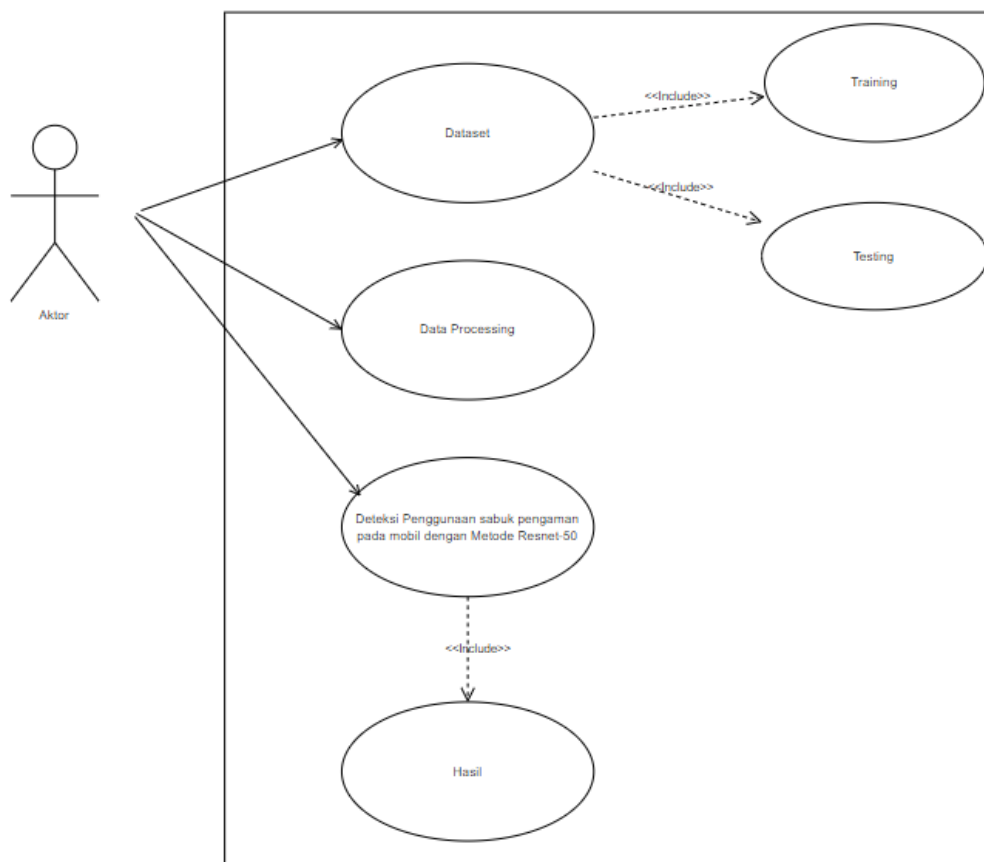
Gambar 3.5 proses Convolutional Neural Network

3.9 Quick Plan & Modelling Quick Desain

Pada perancangan sistem ini mengenai alur kerja sistem yang akan dibuat dan juga rancangan aktor –aktor, serta proses – proses yang akan berinteraksi pada aplikasi tersebut dengan menggunakan Unified Modeling Language (UML), blok diagram, Flowchart dan Residual Network.

3.4.1 Use Case

Pada Gambar 3.8 use case menjelaskan hubungan user dengan sistem. Pada sistem yang telah dibangun, user dapat memilih Dataset berupa gambar. Kemudian saat melakukan proses training dan testing akan memilih dataset. user dapat melakukan processing pada gambar penggunaan yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman dengan metode *ResNet-50*. Keluaran sistem menampilkan hasil deteksi pengemudi yang memakai sabuk pengaman atau pengemudi yang tidak memakai sabuk pengaman.



Gambar 3.6 Use Case

3.4.1.1 Skenario Use Case

Skenario use case adalah rincian tentang subbab dari proses use case yang dihasilkan. Skenario use case terdiri dari nama use case, tujuan, aktor, kondisi awal, aksi aktor, realisasi sistem. Skenario sistem penggunaan yang dihasilkan terlihat pada Tabel 3.1 memilih data uji dan Tabel 3.2 deteksi penggunaan sabuk pengaman.

Tabel 3. 2 Skenario Use Case memilih dataset

Identifikasi	
Use Case	Memilih Dataset
Tujuan	Aktor dapat memilih dataset berupa gambar objek penggunaan sabuk pengaman pada pengemudi mobil
Aktor	User
Skenario Sistem	
Kondisi Awal	Dilakukan pada saat aktor berada pada halaman utama
Aksi Aktor	Realisasi Sistem
1. Klik tombol “browse” untuk memilih data uji.	2. Menampilkan halaman direktori tempat menyimpan data file uji objek penggunaan sabuk pada pengemudi mobil.

3. Pilih gambar yang akan digunakan sebagai data uji dan pilih tombol “open”.	4. Menampilkan file data uji yang dipilih.
---	--

Tabel 3. 3 Skenario Use Case Dataset Training

Identifikasi	
Use Case	Memilih Dataset
Tujuan	Aktor dapat memilih dataset berupa gambar objek penggunaan sabuk pengaman pada pengemudi mobil
Aktor	User
Skenario Sistem	
Kondisi Awal	Dilakukan pada saat aktor berada pada halaman utama
Aksi Aktor	Realisasi Sistem
1. Klik tombol “browse” untuk memilih dataset.	2. Menampilkan halaman direktori tempat menyimpan file objek penggunaan sabuk pada pengemudi mobil.

3. Pilih gambar yang akan digunakan sebagai dataset	4. Menampilkan file untuk proses data latih.
---	--

Tabel 3. 4 Skenario Use Case Dataset Testing

Identifikasi	
Use Case	Memilih Dataset
Tujuan	Aktor dapat memilih dataset berupa gambar objek penggunaan sabuk pengaman pada pengemudi mobil
Aktor	User
Skenario Sistem	
Kondisi Awal	Dilakukan pada saat aktor berada pada halaman utama
Aksi Aktor	Realisasi Sistem
1. Klik tombol “browse” untuk memilih data uji.	2. Memproses feature map dan filter masukan secara efisien

3. Pilih gambar yang akan digunakan sebagai data uji dan pilih tombol “open”.	4. Menampilkan file data uji yang dipilih.
---	--

Tabel 3. 5 Skenario Use Case Data Processing

Identifikasi	
Use Case	User Case Data Processing
Tujuan	Aktor dapat memproses dataset untuk melakukan deteksi pada objek
Aktor	User
Skenario	
Kondisi Awal	Dilakukan pada saat aktor berada pada halaman utama.
Aksi Aktor	Realisasi Sistem

1. Klik tombol “proses” untuk memproses data uji.	2. Perubahan Ukuran Data, melibatkan perubahan ukuran semua data agar sesuai dengan ukuran input 3. normalisasi data dengan menyusutkan nilai piksel gambar
---	--

Tabel 3. 6 Skenario Use Case Deteksi Penggunaan Sabuk Pengaman dengan Metode Resnet-50

Identifikasi	
Use Case	User Case Deteksi Penggunaan Sabuk Pengaman dengan Metode <i>Resnet-50</i>
Tujuan	Aktor dapat memproses data uji untuk melakukan deteksi pada objek pengemudi yang menggunakan sabuk pengaman dan objek pengemudi yang tidak menggunakan sabuk pengaman dengan menggunakan Metode <i>ResNet-50</i> .
Aktor	User
Skenario	
Kondisi Awal	Dilakukan pada saat aktor berada pada halaman utama.

Aksi Aktor	Realisasi Sistem
1. Klik tombol “proses” untuk memproses data uji.	<p>2. tahap kedua dilakukan metode <i>ResNet-50</i> untuk menghasilkan nilai bobot secara konvolusional pada citra.</p> <p>3. Tahap ketiga keluaran sistem yang dihasilkan akan mendeteksi pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman.</p>

Tabel 3. 7 Skenario Use Case Hasil Deteksi Sabuk Pengaman

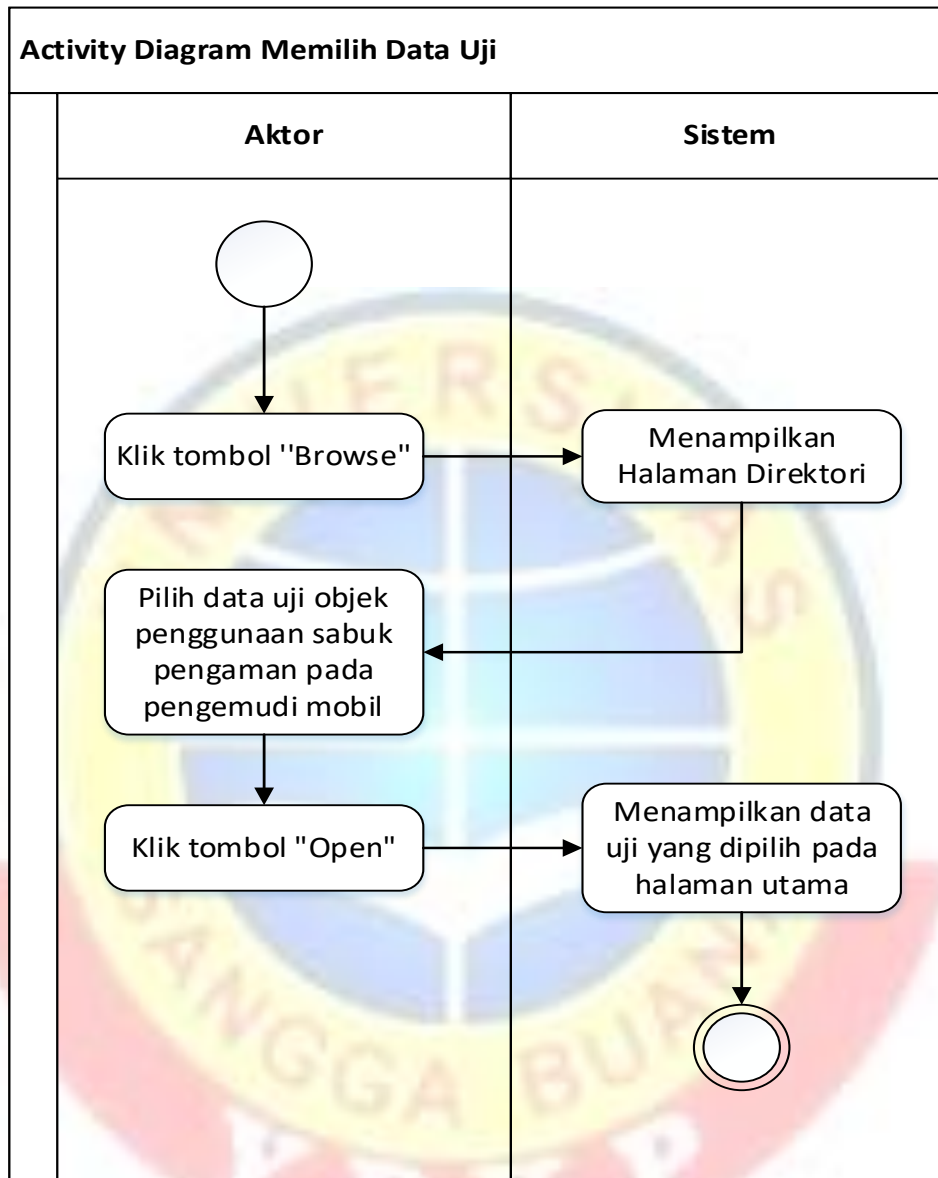
Identifikasi	
Use Case	User Case Hasil Deteksi Penggunaan Sabuk Pengaman dengan Metode <i>Resnet-50</i>
Tujuan	Aktor dapat memproses data uji untuk melakukan deteksi pada objek pengemudi yang menggunakan sabuk pengaman dan objek pengemudi yang tidak menggunakan sabuk pengaman dengan menggunakan Metode <i>ResNet-50</i> .
Aktor	User
Skenario	
Kondisi Awal	Dilakukan pada saat aktor berada pada halaman utama.

Aksi Aktor	Realisasi Sistem
1. Klik tombol “proses” untuk memproses data uji.	2. tahap kedua dilakukan metode <i>ResNet-50</i> untuk menghasilkan nilai bobot secara konvolusional pada citra. 3. Tahap ketiga keluaran sistem yang dihasilkan akan mendeteksi pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman.

3.4.2 Activity Diagram

Activity diagram merupakan suatu proses yang menjelaskan alur kerja suatu sistem yang dibuat, bertujuan untuk menjelaskan lebih lanjut proses skenario use case. Berikut akan di gambarkan activity diagram untuk masing-masing use case.

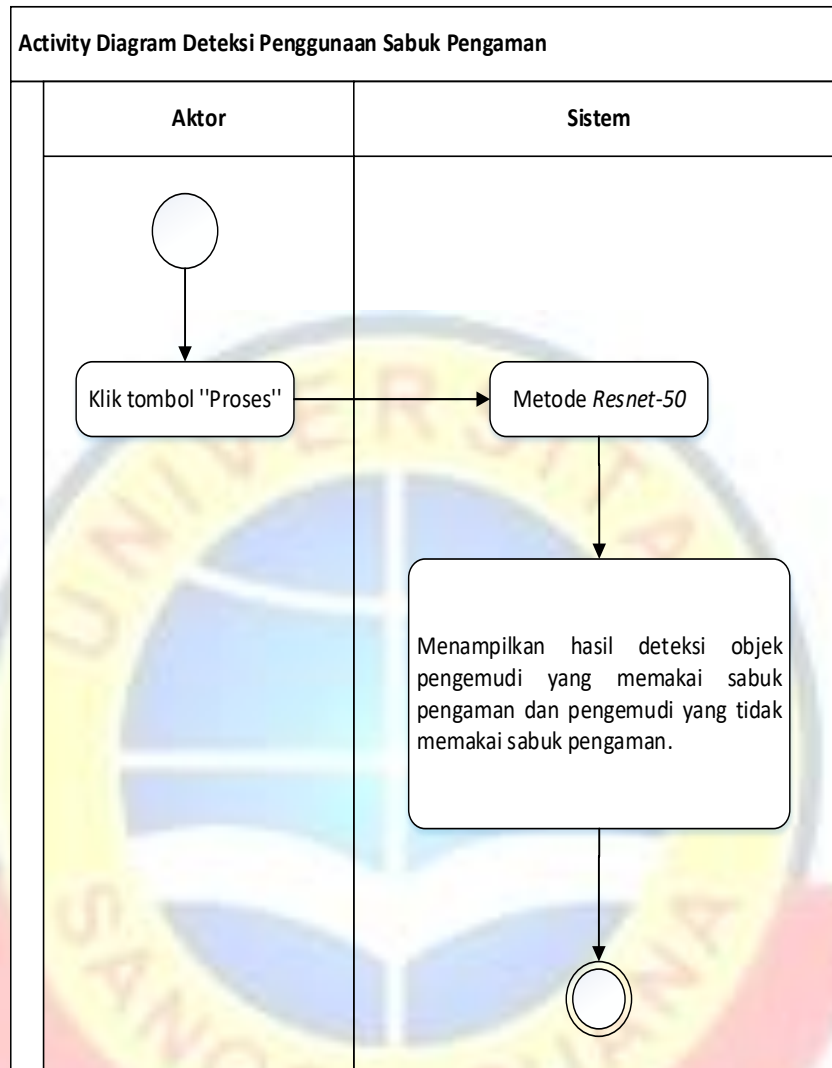
1. Activity Diagram Memilih Dataset



Gambar 3.7 Activity Memilih Data Uji

Activity diagram pada Gambar 3.9 menjelaskan bahwa ketika aktor menekan tombol “Browse” pada halaman utama, sistem akan menampilkan halaman direktori sehingga user dapat memilih data uji objek penggunaan sabuk pengaman pada halaman direktori. Ketika sudah memilih data uji objek penggunaan sabuk pengaman klik tombol “Open” untuk menampilkan data uji yang dipilih pada halaman utama.

2. Activity Diagram Deteksi Penggunaan Sabuk Pengaman



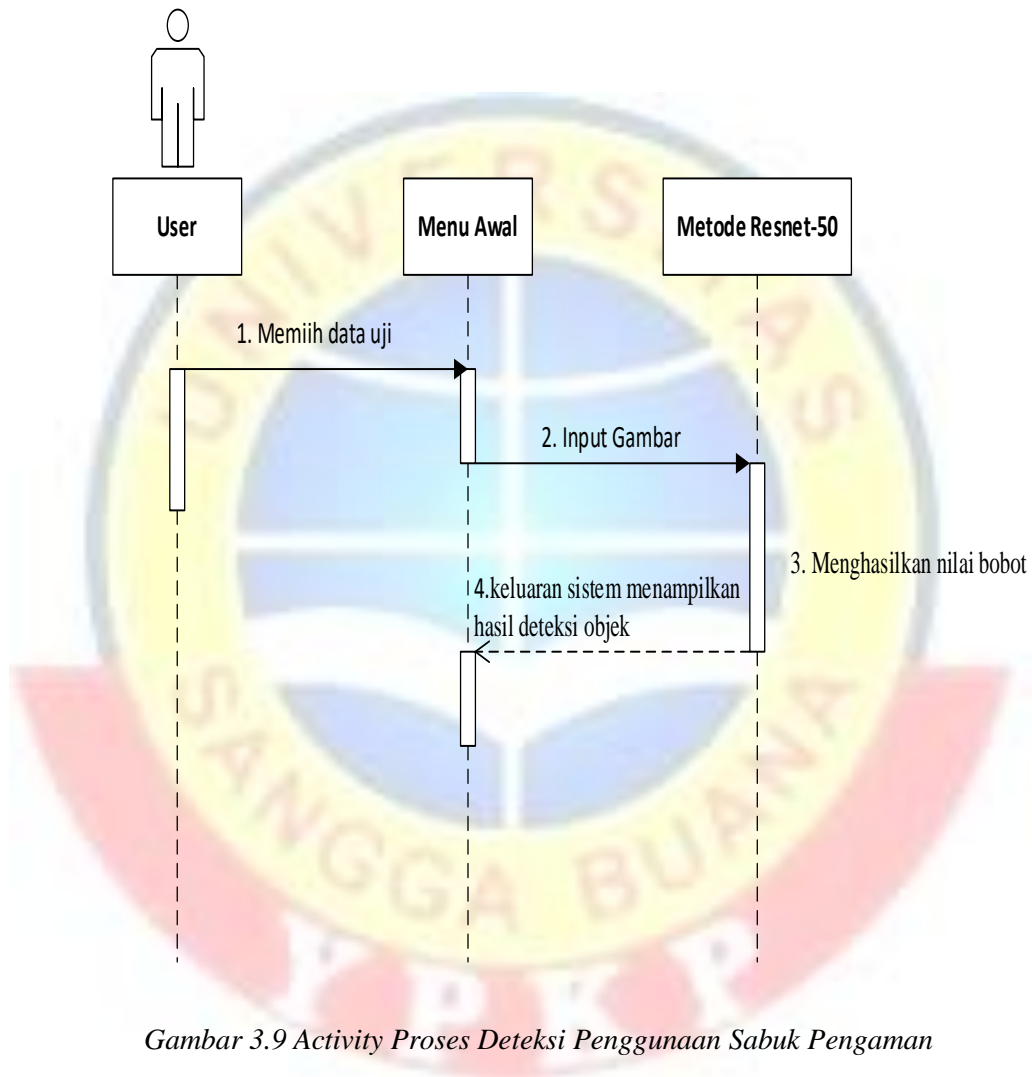
Gambar 3.8 Activity Proses Deteksi Penggunaan Sabuk Pengaman

Pada Gambar 3.10 menjelaskan mengenai interaksi yang terjadi antara user dan sistem, ketika user menekan tombol “Proses” pada halaman utama. Maka sistem akan melakukan proses Metode *ResNet-50* untuk mendeteksi objek pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman pada pengemudi mobil.

2.4.3 Sequence Diagram

Pada Gambar 3.11 merupakan sequence diagram dari sistem deteksi penggunaan pengemudi mobil pada sabuk pengaman. Pada sistem yang dibangun,

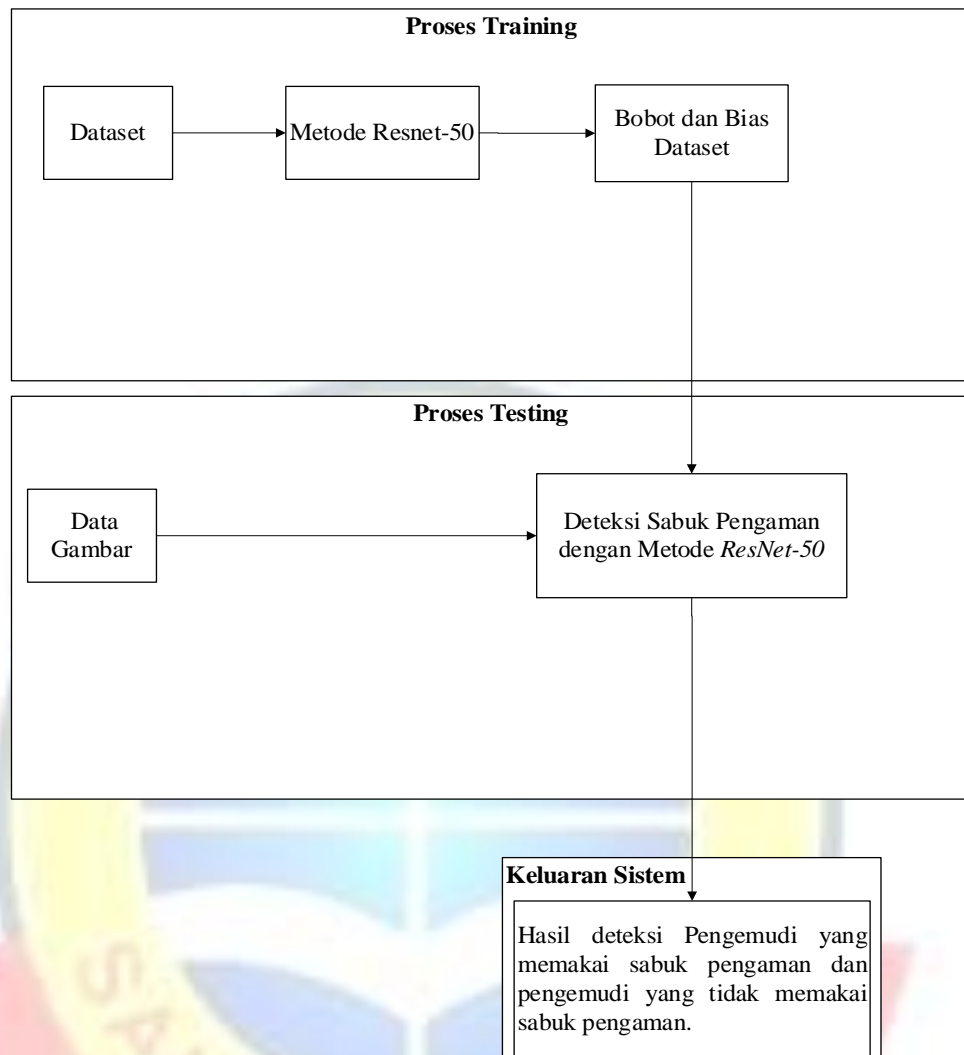
user memilih data uji berupa gambar, kemudian data uji akan dilakukan proses *ResNet-50*, Proses yang telah dilakukan menghasilkan nilai bobot untuk mendeteksi objek pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman dengan menampilkan bounding box dan kelas objek yang dideteksi.



Gambar 3.9 Activity Proses Deteksi Penggunaan Sabuk Pengaman

3.4.4 Blok Diagram

Pada tahap ini dibuat blok diagram yang merepresentasikan operasi sistem dengan menunjukkan garis besar sistem yang dibuat mulai dari proses training dan proses testing. Diagram blok dapat dilihat pada Gambar 3.12.



Gambar 3.10 Blok Diagram

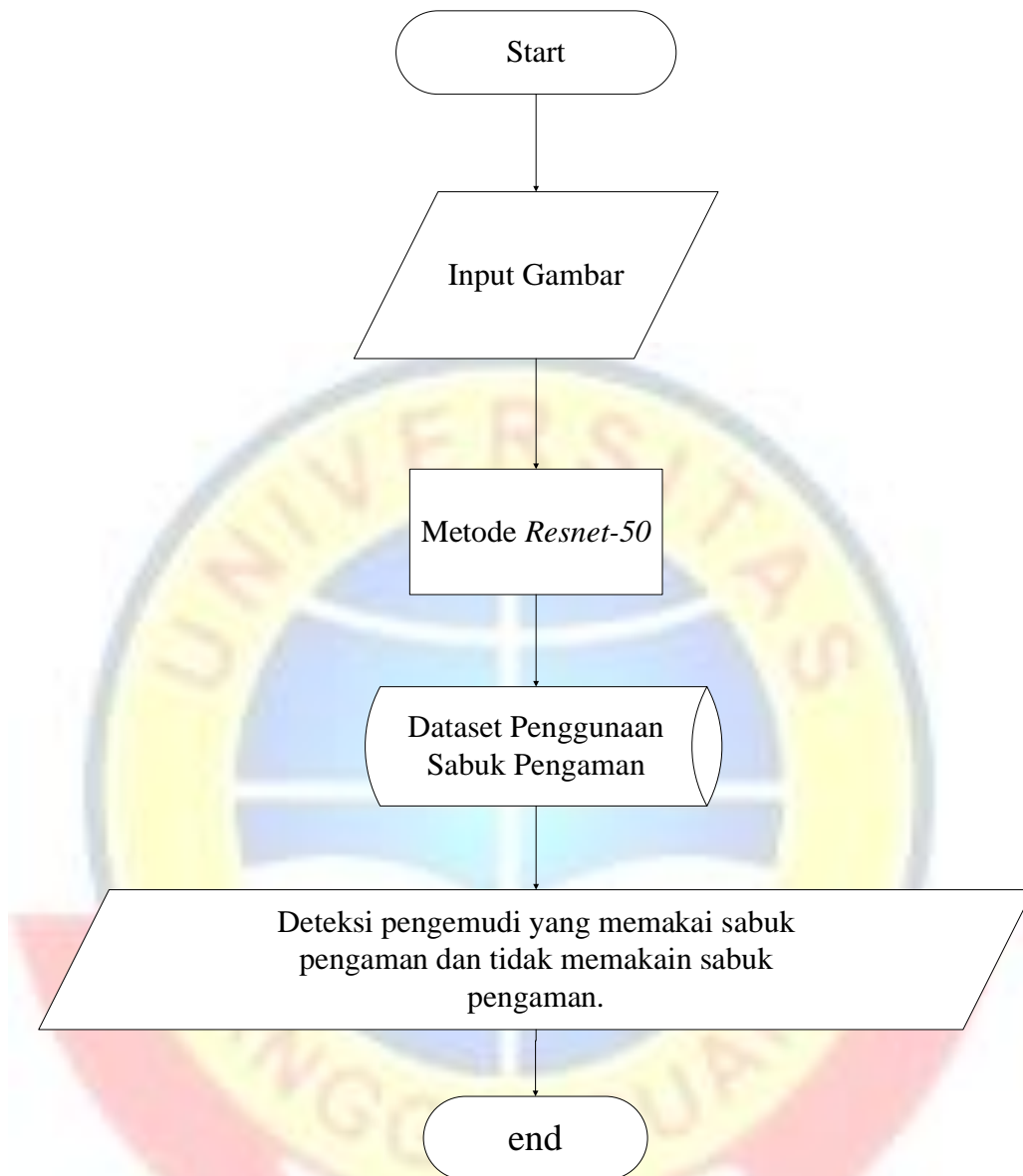
Pada Gambar 3.12 ditunjukkan blok diagram yang mengilustrasikan proses deteksi penggunaan sabuk pengaman pada pengemudi mobil. Penjelasan setiap bloknya akan dijelaskan sebagai berikut:

- a. Pada proses training, dataset yang digunakan mencakup objek pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman.
- b. Setiap masukan dataset dilakukan proses training menggunakan metode *ResNet-50*. Dataset terdiri dari *ground truth box* yang mewakili keberadaan objek yang sebenarnya pada *training image* yang ditandai dengan *bounding box* dan kelas objek.

- c. Pada tahap testing, masukan data uji berupa gambar penggunaan pengemudi mobil pada sabuk pengaman.
- d. Masukan sistem dilakukan proses metode *ResNet-50* untuk menghasilkan nilai bobot dan bias untuk mendeteksi penggunaan pengemudi mobil pada sabuk pengaman.
- e. Kemudian melakukan proses pencocokan nilai bobot dan bias data uji dengan nilai bobot dan bias dataset sehingga menghasilkan kelas objek yang dikenali berdasarkan dataset.
- f. Keluaran sistem yang dihasilkan akan mendeteksi pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman.

3.4.5 Flowchart

Dalam flowchart yang dibuat pada Gambar 3.13 menggambarkan cara kerja sistem untuk deteksi penggunaan pengemudi mobil pada sabuk pengaman. Proses sistem untuk mendeteksi penggunaan pengemudi mobil pada sabuk terdapat beberapa tahapan yaitu tahap pertama adalah masukan sistem berupa file gambar pengemudi menggunakan sabuk pengaman dan pengemudi yang tidak menggunakan sabuk pengaman. Kemudian tahap kedua dilakukan proses metode *ResNet-50* untuk menghasilkan nilai bobot secara konvolusional pada citra. Kemudian tahap ketiga dilakukan proses pencocokan nilai bobot data uji dengan nilai bobot dataset. Tahap keempat keluaran sistem yang dihasilkan dapat mendeteksi pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman.



Gambar 3.11 Flowchart

3.4.6 Proses Residual Network

Pada proses ResNet yang dilakukan menggunakan 50 layer. Proses yang dilakukan terdiri dari operasi konvolusi 7x7, max pooling 3x3, konvolusi 1x1, aktivasi ReLU, operasi konvolusi 3x3, aktivasi ReLU, operasi konvolusi 1x1, Jumlah filter yang digunakan pada setiap operasi konvolusi pada modul residual disesuaikan dengan tingkatan layer residual network yang digunakan. Proses

ResNet menghasilkan nilai bobot untuk memprediksi adanya objek pengemudi yang memakai sabuk pengaman dan pengemudi yang tidak memakai sabuk pengaman yang dikenali berdasarkan model dataset yang telah dibuat.

3.3 Pembuatan Prototype (Construction of Prototype)

Pada tahap ini dibuat perancangan antarmuka sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil.



Gambar 3.12 Antarmuka Sistem

Halaman menu pada Gambar 3.14 adalah halaman menu deteksi penggunaan sabuk pengaman pada pengemudi mobil. Masing-masing citra gambar memiliki format *.jpeg/png. Lokasi direktori file untuk memilih gambar data uji. Pada proses citra akan melakukan proses Metode *ResNet-50* untuk mendeteksi objek

penggunaan pengemudi mobil pada sabuk pengaman. Sistem menampilkan hasil deteksi pengemudi mobil yang memakai sabuk pengaman dan tidak memakai sabuk pengaman.

3.4 Deployment Delivery & feedback

Tahap deployment delivery & feedback dilakukan untuk menguji apakah prototype yang telah dibuat sesuai dengan rancangan dan tujuan penelitian yang telah ditentukan. deployment delivery & feedback pertama yang akan dilakukan adalah pengujian fungsionalitas menggunakan black-box secara alpha, baik dalam lingkungan pengembang, tahap berikutnya dilakukan kinerja algoritma dengan perhitungan precision dan recall, Rincian lengkap mengenai pengujian tersebut akan dijelaskan pada bab 4.



BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini dijelaskan implementasi dari sistem yang telah dibuat dan hasil pengujian sistem. Implementasi dilakukan dengan menggunakan bahasa pemrograman *Python*. Setelah melakukan implementasi, maka dilanjutkan pengujian terhadap data uji.

4.1. Lingkungan Pengembangan

Sub-bab ini menjelaskan implementasi dari hasil penelitian antara lain adalah perangkat yang digunakan, implementasi Arsitektur *Residual Network*, dan implementasi Sistem.

4.1.1. Kebutuhan Pembangunan *Hardware*

Berikut ini merupakan spesifikasi *hardware* yang digunakan dalam membangun sistem:

1. Komputer atau laptop dengan spesifikasi:
 - a. *Processor Intel Core i3-3217U 1.8 GHz.*
 - b. RAM 6 GB.
 - c. *Storage 500 GB HDD 5400 RPM.*
2. Kamera dengan spesifikasi:
 - a. Kamera: 48 MP (Resolusi 1280x720).

4.1.2. Kebutuhan Perangkat Lunak

Pada bagian perangkat lunak, yang dibutuhkan dalam pembangunan sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil dijelaskan sebagai berikut:

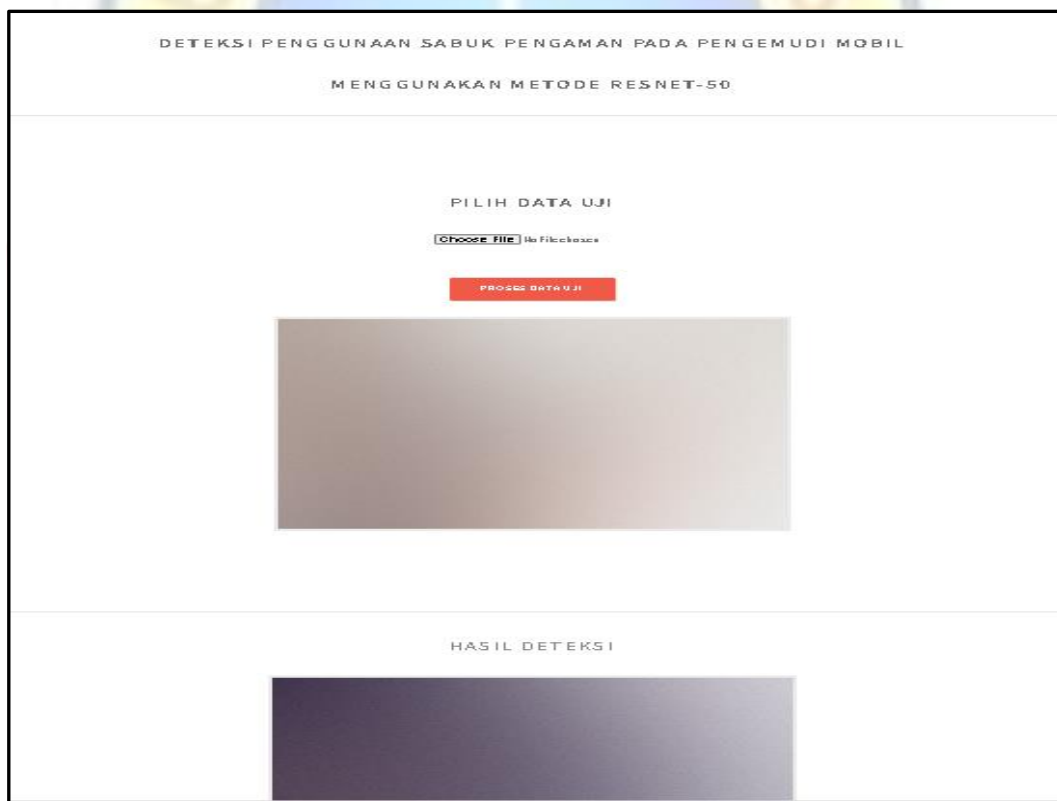
- a. Windows 10 Professional 64-bit dibutuhkan sebagai sistem operasi pada laptop yang digunakan untuk menuliskan dokumentasi laporan penelitian.
- b. JetBrains Pycharm Community Edition merupakan sebuah platform Bahasa pemrograman python yang digunakan untuk membuat kode proses back-end pada sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil.

4.2 Implementasi Sistem

Implementasi sistem merupakan tahap untuk menerapkan hasil dari perancangan yang telah dilakukan terhadap sistem yang akan dibangun, sehingga sistem tersebut dapat digunakan. Implementasi dalam penelitian ini mencakup implementasi antarmuka sistem, pilih data uji dan hasil deteksi.

4.3.1 Antarmuka Sistem

Gambar 4.1 adalah antarmuka sistem deteksi pengemudi yang memakai sabuk pengaman dan tidak memakai sabuk pengaman. Pada antarmuka sistem ini menampilkan judul penelitian, menampilkan tombol choose file, proses data uji dan hasil deteksi penggunaan pengemudi pada sabuk pengaman dengan metode *Resnet-50*. Pada Gambar 4.1 menunjukkan Antarmuka sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil.

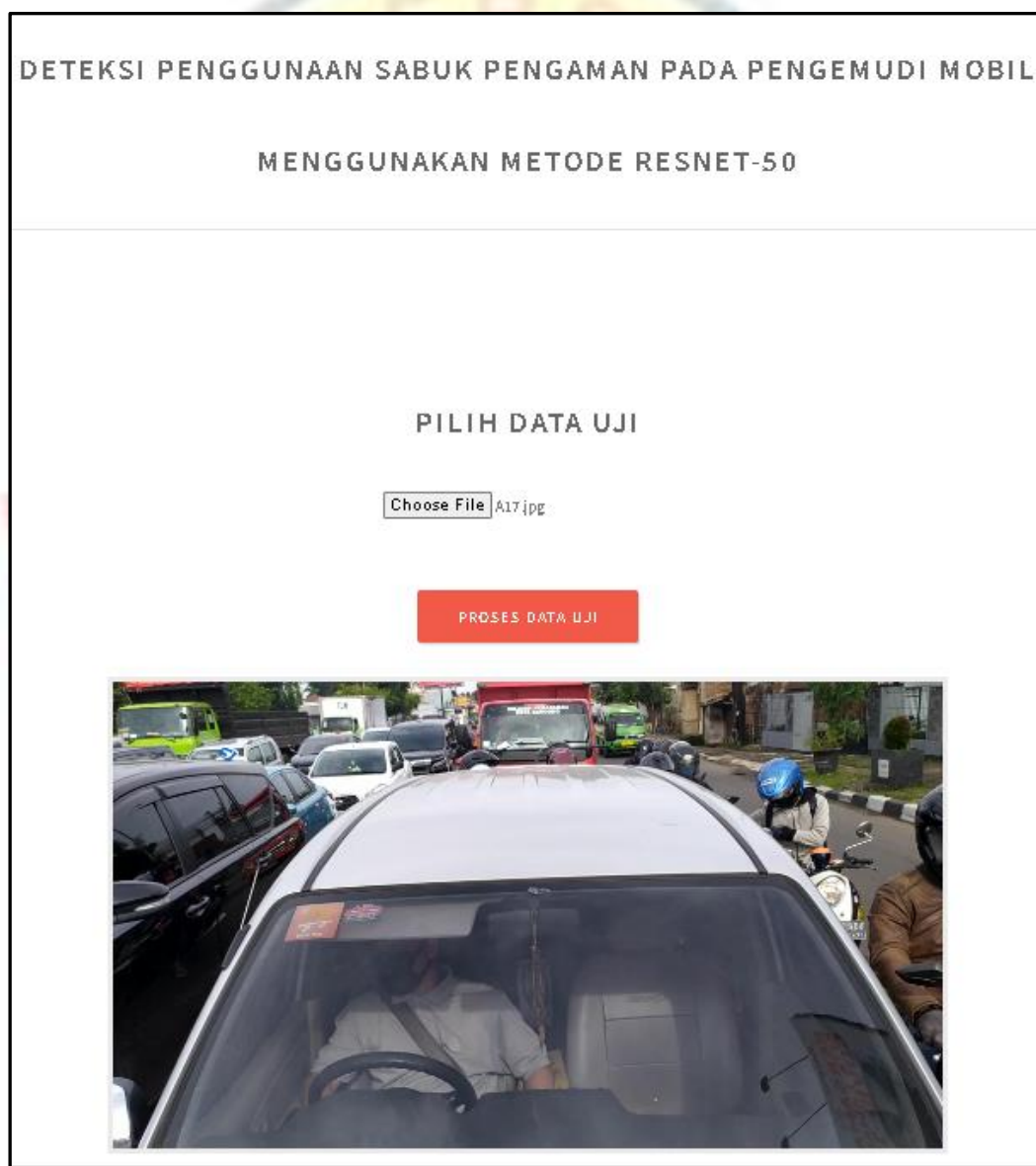


Gambar 4.1 Antarmuka Sistem

4.3.2 Pemilihan Data Uji

Pada Gambar 4.2 adalah menu pilih data uji pada sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil. Pada menu data uji, terdapat tombol

“Choose file” yang berfungsi untuk memilih data uji yang tersimpan pada file direktori dan tombol “Proses data uji” untuk melakukan tahap selanjutnya yaitu melakukan proses metode *ResNet-50* untuk menghasilkan nilai bobot secara konvolusional pada citra. Setelah itu, sistem akan melakukan pencocokan nilai bobot dengan dataset penggunaan sabuk pengaman. Selanjutnya akan menghasilkan output yang menunjukkan apakah pengemudi yang memakai sabuk pengaman dan tidak memakai sabuk pengaman.



Gambar 4.2 Pilih Data Uji

4.3.3 Hasil Deteksi

Gambar 4.3 dan 4.4 adalah hasil deteksi pengemudi yang menggunakan sabuk pengaman dan hasil deteksi tidak menggunakan sabuk pengaman. Hasil deteksi ini berupa label dari objek yang telah diberikan *bounding box*, beserta nilai *klasifikasi* terhadap suatu kelas dari label objek yang dideteksi.



Gambar 4.4 Hasil Deteksi pengemudi menggunakan sabuk pengaman



Gambar 4.4 Hasil Deteksi Pengemudi yang tidak menggunakan sabuk pengaman

Pada Gambar 4.3 adalah hasil deteksi pengemudi menggunakan sabuk pengaman (MSP) dan pada Gambar 4.4 adalah hasil deteksi pengemudi tidak menggunakan sabuk pengaman (TMSP).

4.4 Pengujian Kinerja Sistem

Pengujian kinerja sistem dilakukan untuk mengukur precision, recall, f1 score dan accuracy dalam mendeteksi penggunaan pengemudi mobil pada sabuk pengaman dengan menggunakan metode *ResNet-50*. Data untuk deteksi penggunaan sabuk pengaman diambil pada pemberhentian lampu lintas di jalan Gasibu dan jalan Soekarno Hatta. Pengambilan data dilakukan pada pagi hari dan sore hari, dengan pengambilan citra objek pengemudi pada jarak 1 sampai 2 meter. Pengujian menggunakan 30 citra objek penggunaan sabuk pengaman yang dibagi menjadi dua tahap, pengujian tahap pertama dilakukan terhadap 15 citra objek pengemudi yang menggunakan sabuk pengaman. Pengujian tahap kedua dilakukan terhadap 15 citra objek pengemudi yang tidak menggunakan sabuk pengaman. Pada Tabel 4.1 Hasil nilai rata-rata *precision*, *recall*, *f1 score* dan *accuracy*.

Tabel 4.1 Menggunakan sabuk pengaman pada pengemudi mobil

Pengujian	Nama citra (*jpg)	TP	FP	FN	TN	Terdeteksi	Precision	Recall	F1 Score	Accuracy
1	A1	1	0	1	0	2	1	0,5	0,66	0,5
2	A2	1	0	0	0	1	1	1	1	1
3	A3	2	0	0	0	2	1	1	1	1
4	A4	1	0	0	0	1	1	1	1	1
5	A5	2	1	0	0	3	0,66	1	0,79	0,66
6	A6	1	0	0	0	1	1	1	1	1
7	A7	1	0	0	0	1	1	1	1	1
8	A8	1	0	0	0	1	1	1	1	1
9	A9	0	0	0	0	0	0	0	0	0
10	A10	1	0	0	0	1	1	1	1	1
11	A11	1	0	0	0	1	1	1	1	1
12	A14	2	0	0	0	2	1	1	1	1
13	A15	1	0	0	0	1	1	1	1	1
14	A16	1	0	0	0	1	1	1	1	1
15	A17	1	0	0	0	1	1	1	1	1
Rata -Rata							91%	90%	89%	87%

Tabel 4.2 Tidak Menggunakan Sabuk Pada pengemudi Mobil

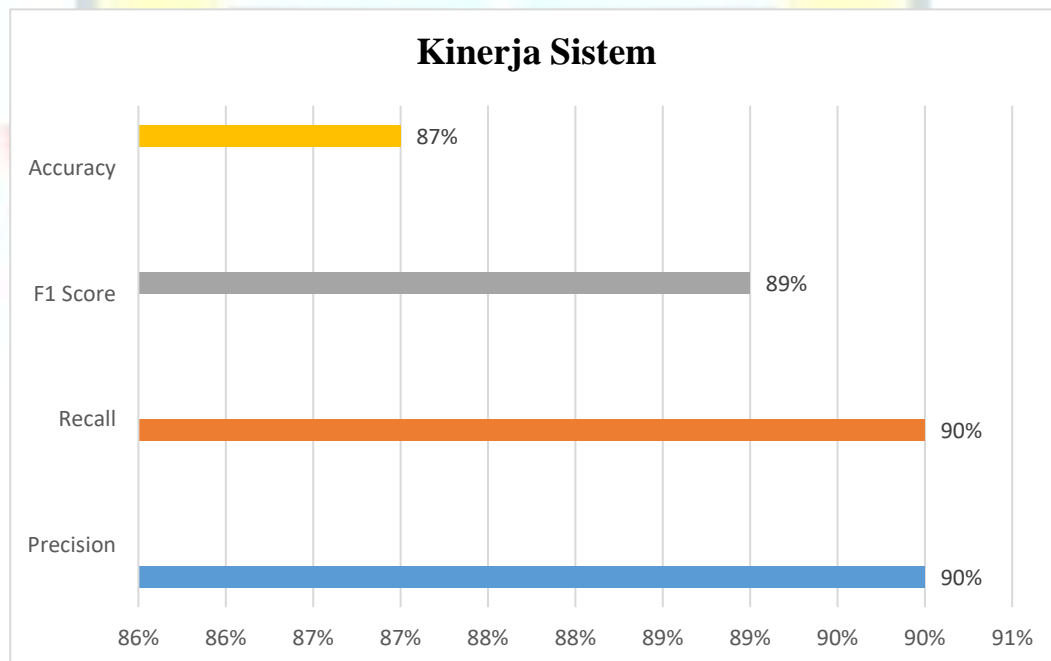
Pengujian	Nama citra (*jpg)	TP	FP	FN	TN	Terdekteksi	Precision	Recall	F1 Score	Accuracy
1	A21	1	0	0	1	2	1	1	1	1
2	A22	1	0	0	0	1	1	1	1	1
3	A23	1	0	0	0	1	1	1	1	1
4	A24	1	0	0	0	1	1	1	1	1
5	A25	1	0	0	0	1	1	1	1	1
6	A26	1	0	0	0	1	1	1	1	1
7	A27	1	0	0	0	1	1	1	1	1
8	A28	1	0	0	0	1	1	1	1	1
9	A29	2	0	0	0	2	1	1	1	1
10	A30	1	0	0	0	1	1	1	1	1
11	A31	1	0	0	0	1	1	1	1	1
12	A32	1	0	0	1	2	1	1	1	1
13	A33	1	1	0	0	2	0,5	1	0,66	0,5
14	A34	0	0	0	0	0	0	0	0	0
15	A35	2	0	1	1	4	1	0,66	0,79	0,75
Rata -Rata							90%	91%	89%	88%

Pada Tabel 4.1 dan Tabel 4.2 merupakan hasil pengujian Confusion matrix deteksi penggunaan pengemudi mobil pada sabuk pengaman dengan metode *ResNet-50*. Pada pengujian pertama dengan 15 citra objek pengemudi yang memakai sabuk pengaman menunjukkan nilai precision sebesar 91%, nilai recall sebesar 90%, nilai f1 score sebesar 89% dan nilai accuracy sebesar 87%. Pada pengujian kedua dengan 15 citra objek pengemudi yang tidak memakai sabuk pengaman menunjukkan nilai precision sebesar 90%, nilai recall sebesar 91%, nilai f1 score sebesar 89% dan nilai accuracy sebesar 88%.

Tabel 4.3 Hasil nilai rata-rata

Rata-rata					
No	Pengujian	Precision	Recall	F1 Score	Accuracy
1	15 Citra Objek pengemudi mobil yang menggunakan sabuk pengaman.	91%	90%	89%	87%
2	15 Citra Objek pengemudi mobil yang tidak menggunakan sabuk pengaman.	90%	91%	89%	88%
Rata-Rata		90%	90%	89%	87%

Gambar 4.5 Kinerja Sistem



Pada Gambar 4.5 merupakan Graphic hasil pengujian kinerja sistem deteksi penggunaan pada mobil pada sabuk pengaman dengan metode *ResNet-50*. Pengujian kinerja sistem ini menunjukkan nilai precision sebesar 90%, nilai recall sebesar 90%, nilai f1 score sebesar 89% dan nilai accuracy sebesar 87%.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan uraian pembahasan dan permasalahan pada bab sebelumnya mengenai deteksi penggunaan pengemudi mobil pada sabuk pengaman dengan metode *Resnet-50*, maka dapat diambil kesimpulan sebagai berikut:

1. Sistem deteksi penggunaan sabuk pengaman pada pengemudi mobil dapat dilakukan dengan menggunakan metode *Resnet-50* (*Residual Network 50*).
2. Berdasarkan proses pengujian kinerja sistem ini menunjukkan nilai precision sebesar 90%, nilai recall sebesar 90%, nilai f1 score sebesar 89% dan nilai accuracy sebesar 87%.

5.2 Saran

Berdasarkan penelitian, ada beberapa saran untuk pengembangan sistem yang dapat dilakukan, sebagai berikut:

1. Dalam mendeteksi sabuk pengaman pada pengemudi mobil menggunakan fitur tembus pandang pada kaca mobil yang gelap dan memantulkan cahaya, agar penggunaan sabuk pengaman dapat dideteksi.
2. Menggunakan kamera yang lebih baik untuk mendeteksi penggunaan pengemudi mobil pada sabuk pengaman dapat meningkatkan kualitas deteksi dan akurasi sistem yang akurat.

DAFTAR PUSTAKA

- Atheadeth, L., t.thn. *researchgat.* [Online]
Available at: https://www.researchgate.net/figure/ResNet-50-architecture-Example-network-architecture-for-ImageNet_fig2_371517021
- Aditya Santoso, Gunawan Ariyanto, 2018. Implementasi Deep Learning berbasis Keras untuk Pengenalan Wajah. *Jurnal teknik elektro.*
- Agarap, A., 2018. Deep Learning using Rectified Linear Units (ReLU). *Cornell University.*
- Alexey Kashevnik, Ammar Ali, Igor Lashkov, Nikolay Shilov, 2020. Seat Belt Fastness Detection Based on Image Analysis from Vehicle In-abin Camera. *researchgate.*
- Anon., t.thn.
- Ari Peryanto, A. Y. R. U., 2019. Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network. *Jurnal Format.*
- Ariel_一只猫的旅行, 2020. *MathWorks.* [Online]
Available at: <https://www.cnblogs.com/ariel-dreamland/p/13644578.html>
- Deyun Wang, 2022. Intelligent Detection of Vehicle Driving Safety Based on Deep Learning. *hindawi.*
- DS Bhupal Naik, G Sai Lakshmi, V Ramakrishna Sajja, D Venkatesulu, J Nageswara Rao, 2021. Driver's Seat Belt Detection Using CNN. *Turkish Journal of Computer and Mathematics Education.*
- E Sentosa, H Armanto, P Pickerling, 2022. Pengenalan Ekspresi Wajah dengan CNN dan Wavelet. *INSYST.*
- Eko Cahyono Putro, Rolly Maulana Awangga, Roni Andarsyah, 2020. *Tutorial Object Detection People With Faster region-Based Convolutional Neural Network(Faster R-CNN).* Bandung: Kreatif Industri Nusantara.
- He, K., Zhang, X., Ren, S., & Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *In Proceedings of the IEEE international conference on computer vision .*

- Hendy Mulyawan, M Zen Hadi Samsono, Setiawardhana, 2011. IDENTIFIKASI DAN TRACKING OBJEK BERBASIS IMAGE. *CORE*.
- Irma Amelia Dewi, Lisa Kristiana , Arsyad Ramadhan Darlis, 2019. Deep Learning RetinaNet based Car Detection for Smart Transportation Network. *ELKOMIKA*.
- Jean Baptista Jimmy Robert Openg, Marselina Endah Hiswati, Hamzah Hamzah, 2022. Klasifikasi Unggas Ordo Anseriformes Berdasarkan Citra Menggunakan Metode Deep Learning Dengan Algoritma Convolutional Neural Network (CNN). *Artificial Neural Networks*.
- Karlita, T. et al., 2019. Deteksi Region of Interest Tulang pada Citra B-mode secara Otomatis Menggunakan Region Proposal Networks. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*.
- Kundu, N., 2023. *Menjelajahi ResNet50: Pandangan Mendalam pada Arsitektur Model dan Implementasi Kode*. [Online] Available at: <https://medium.com/@nitishkundu1993/exploring-resnet50-an-in-depth-look-at-the-model-architecture-and-code-implementation-d8d8fa67e46f>
- MathWorks, 2020. *MathWorks*. [Online] Available at: <https://www.mathworks.com/help/vision/ref/nnet.cnn.layer.rcnnboxregressionlayer.html>
- N.Nufus, D.M Arifin, A.S Satyawan, R.A.S Nugraha, M.I Asyasyakuur, N.N.A.M Santi, C.H Parangin, Ema, 2021. Sistem Pendeteksi Pejalan Kaki Di Lingkungan terbatas berbasis SSD MobileNet V2 Dengan Menggunakan Gambar 360 Ternormalisasi. *Prosiding Seminar Nasional Sains Teknologi dan Inovasi Indonesia*.
- Novelita Dwi Miranda, Ledy Novamizanti, Syamsul Rizal, 2020. CONVOLUTIONAL NEURAL NETWORK PADA KLASIFIKASI SIDIK JARI MENGGUNAKAN RESNET-50. *Jurnal Teknik Informatika (JUTIF)*.

- Nur Fadlia, Rifki Kosasih, 2019. KLASIFIKASI JENIS KENDARAAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN). *semanticscholar*.
- Pressman, 2010. *Rekayasa Perangkat Lunak*. s.l.:s.n.
- Pulung Adi Nugroho, Indah Fenriana, Rudy Arijanto, M.Kom, 2020. IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL (CNN) PADA EXPRESI MANUSIA. *ALGOR*.
- Revydo Bima Anshori, Hilman Fauzi, Thomhert Suprpto Siadari, 2022. Klasifikasi Citra Kanker Serviks Menggunakan Deep Residual Network. *e-Proceeding of Engineering*.
- Rina, 2023. *Medium*. [Online]
Available at: <https://esairina.medium.com/memahami-confusion-matrix-accuracy-precision-recall-specificity-dan-f1-score-610d4f0db7cf>
- Rinaldi Munir, 2004. *Pengolahan citra digital dengan pendekatan algoritmik*. Bandung: s.n.
- Suprihanto, I. A. M. F. M. A. Z. Z., 2022. Analisis Kinerja ResNet-50 dalam Klasifikasi Penyakit pada Daun Kopi Robusta. *JURNAL INFORMATIKA*, p. 116~122.
- Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár, 2017. Focal loss for dense object detection. *In Proceedings of the IEEE international conference on computer vision*.
- Veeramachaneni, T., 2023. *Medium*. [Online]
Available at: <https://medium.com/@trinadh.datascientist/understanding-convolutional-neural-networks-architecture-convolution-pooling-and-parameter-b915f376ab59>
- Yunus, M., 2020. *Alat dan Bahan untuk Machine Learning*. [Online]
Available at: <https://medium.com/@yunusmuhammad007/3-alat-dan-bahan-untuk-machine-learning-92c717286624>



LAMPIRAN

Proses Deteksi

```
@app.route('/uploads/<filename>')
def uploaded_file(filename):
    PATH_TO_TEST_IMAGES_DIR = app.config['UPLOAD_FOLDER']
    TEST_IMAGE_PATHS = [os.path.join(PATH_TO_TEST_IMAGES_DIR, filename.format(i))
    for i in range(1, 2)]

    for image_path in TEST_IMAGE_PATHS:
        labels_to_names = {0: 'MSP', 1: 'TMSP'}
        image = read_image_bgr(image_path)

        draw = image.copy()
        draw2 = image.copy()
        draw3 = image.copy()

        fileoriginal=app.config['UPLOAD_FOLDER']+filename
        filename.partition('.')
        filename, separator, extension = filename.partition('.')
        filepreprocess='static/preprocessing/'+filename+'preprocess.jpg'
        filenameFPN=filename+"-FPN.jpg"

        preproc=preprocess_image(image)
        image = preprocess_image(image)
        image, scale = resize_image(image)

        cv2.imwrite(filepreprocess, preproc)
```

```

boxes, scores, labels=predictdata(image)
boxes /= scale
filedrawbox=drawUntreshold(boxes, scores, labels, draw2, filenameFPN)
filedrawtrbox=drawtreshold(boxes, scores, labels, draw3, filenameFPN)
MSP=0
TMSP=0
for box, score, label in zip(boxes[0], scores[0], labels[0]):
    if score < 0.5:
        break
    color = label_color(label)
    b = box.astype(int)
    draw_box(draw, b, color=color, thickness=2)
    caption = "{} {:.3f}".format(labels_to_names[label], score)
    draw_caption(draw, b, caption)
    print(caption)
    if label == 0:
        MSP = "Pengemudi Yang Menggunakan Sabuk Pengaman"
    else:
        TMSP = "Pengemudi Yang Tidak Menggunakan Sabuk Pengaman"
file=filename+'-test.jpg'
plt.figure(figsize=(20, 20))
plt.axis('off')
cv2.imwrite('static/'+file, draw)
directoryfile="static/"+file
return jsonify(Name=directoryfile, OriginalFile=fileoriginal, Preprocessing=filepreprocess, Drawbox=filedrawbox,
                Treshold=filedrawtrbox, msp=MSP, tmsp=TMSP)

```

Metode resnet 50

```
def predictdata(ImageFile):  
    keras.backend.tensorflow_backend.set_session(get_session())  
    models_path = os.path.join('dataset', sorted(os.listdir('dataset'), reverse=True)[0])  
  
    model = models.load_model(models_path, backbone_name='resnet50')  
    model = models.convert_model(model)  
    print(model.summary())  
  
    start = time.time()  
    boxes, scores, labels = model.predict_on_batch(np.expand_dims(ImageFile, axis=0))  
    print("processing time: ", time.time() - start)  
    return boxes, scores, labels
```





KARTU BIMBINGAN SKRIPSI
S1 - TEKNIK INFORMATIKA
UNIVERSITAS SANGGA BUANA - YPKP

TAHUN AJAR	Ganjil 2023/2024
NPM	2113191037
NAMA	Firdan Abdul Kholiq
PEMBIMBING	Bambang Sugiarto, ST., M.T.
JUDUL	IMPLEMENTASI RESIDUAL NETWORK UNTUK DETEKSI PENGGUNAAN SABUK PENGAMAN PADA PENGEMUDI MOBIL



NO	TANGGAL	POKOK BAHASAN	PARAF PEMBIMBING
	26/12/2023	Bab I	[Signature]
	13/01/2024	Revisi Bab I	[Signature]
	30/01/2024	Bab III A	[Signature]
	05/02/2024	Revisi III A	[Signature]
	13/02/2024	Bab III	[Signature]
	21/02/2024	Revisi Bab III	[Signature]
	23/02/2024	Bab IV	[Signature]
	27/02/2024	Revisi Bab IV	[Signature]
	01/03/2024	Bab V	[Signature]
	01/03/2024	Revisi Bab V	[Signature]

Cat:

1. Minimal bimbingan sebanyak 8x.
2. Kartu ini dikumpulkan sebagai syarat sidang beserta berkas yang lainnya.

Bandung, 01 Maret 2024

Pembimbing

Sugiarto
(Bambang Sugiarto, ST., M.T.)



UPT PERPUSTAKAAN
UNIVERSITAS SANGGA BUANA YPKP
Jl. PHH Mustofa No. 68 Bandung Gedung E Lantai 5
Email: library@usbypkp.ac.id Website: perpustakaan.usbypkp.ac.id

Surat Keterangan Cek Plagiarisme
Nomor : 104/III/SKCP/USB-YPKP/2024

Sehubungan dengan kewajiban Cek Plagiarisme dengan *similarity check maximal 25%* sebagai salah satu kelengkapan persyaratan administrasi bagi mahasiswa tingkat akhir, dengan ini UPT Perpustakaan Universitas Sangga Buana menerangkan bahwa:

Nama : Firdan Abdul Kholiq
NPM : 2113191037
Program Studi : S1 Teknik Informatika
Judul Karya Tulis Ilmiah : "IMPLEMENTASI METODE RESIDUAL NETWORK UNTUK DETEKSI PENGGUNAAN SABUK PENGAMAN PADA PENGEMUDI MOBIL"
Tanggal Cek Turnitin : 02-Mar-24
Status : Lulus dengan **22% Similarity Check**

Adalah benar telah dilakukan *similarity check* sebagaimana data tersebut diatas, dan surat ini dibuat berdasarkan keadaan yang sebenar benarnya, untuk bisa dipergunakan sebagaimana mestinya.

Bandung, 02-Mar-24

Kepala UPT Perpustakaan



Widyapuri Prasastiningtyas, S.Sos., M.I.kom.

NIP. 432.200.173