

**LAPORAN FINAL PROJECT
MATA KULIAH DATA MINING II**

**“PENGEMBANGAN MODEL KLASIFIKASI TIPE KENDARAAN UNTUK
SISTEM PENGENALAN GAMBAR OTOMOTIF”**



ANGGOTA:

Muhammad Razzan Ramadhana	164221014
Brahmantya Fikri Setya Putra	164221034
Reinhart Ananda Siswadi	164221046
Jovita Suryo Angeline	164221094

**PROGRAM STUDI TEKNOLOGI SAINS DATA
FAKULTAS TEKNOLOGI MAJU DAN MULTIDISIPLIN
UNIVERSITAS AIRLANGGA
SURABAYA
2024**

DAFTAR ISI

DAFTAR ISI.....	2
BAB I.....	4
PENDAHULUAN.....	4
1.1 Latar Belakang.....	4
1.2 Rumusan Masalah.....	4
1.3 Tujuan Penelitian.....	5
BAB II.....	6
LANDASAN TEORI.....	6
2.1 Deep Learning.....	6
2.2 Convolutional Neural Networks.....	6
2.3 Stanford Cars Dataset.....	6
2.4 VGG19.....	7
2.5 RESNET50.....	7
2.6 Computer Vision.....	7
BAB III.....	8
METODOLOGI PENELITIAN.....	8
3.1 Sumber Data.....	8
3.2 Metodologi.....	8
3.2.1 Flowchart.....	8
3.2.2 Preprocessing Data.....	9
3.2.3 Splitting Data.....	9
3.2.4 Augmentasi Data.....	9
3.2.5 Model ResNet50.....	9
3.2.6 Model VGG19.....	10
3.2.7 Model CNN.....	11
3.2.8 Training Data.....	11
3.2.9 Testing Dataset.....	12
BAB IV.....	13
HASIL DAN PEMBAHASAN.....	13
4.2 Hasil Training.....	13
4.2.1 Hasil Training Model ResNet50.....	14
4.2.2 Hasil Training Model VGG19.....	14
4.2.3 Hasil Training Model CNN.....	15
4.3 Hasil Testing.....	16
4.3.1 Hasil Testing Model ResNet50.....	16
4.3.2 Hasil Testing Model VGG19.....	16

4.3.3 Hasil Testing Model CNN.....	16
4.4 Confusion Matrix.....	17
4.4.1 Confusion Matrix ResNet50.....	17
4.4.2 Confusion Matrix VGG19.....	17
4.4.3 Confusion Matrix CNN.....	18
4.5 Implementasi Model.....	18
BAB V.....	19
KESIMPULAN DAN SARAN.....	19
LAMPIRAN.....	21

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan teknologi kecerdasan buatan, banyak perubahan signifikan pada berbagai sektor, mulai dari transportasi, keamanan, hingga ritel otomotif. Salah satu penerapannya adalah aplikasi untuk mendeteksi dan pengklasifikasian tipe kendaraan secara otomatis. Kemampuan untuk mengenali jenis dan model kendaraan dari gambar memiliki dampak luas, seperti optimasi sistem lalu lintas, pengelolaan parkir otomatis dan juga pengenalan model kendaraan di sektor ritel.

Dalam sistem gerbang tol, sistem klasifikasi kendaraan konvensional biasanya mengandalkan sensor berat atau dimensi kendaraan untuk menentukan golongan tarif tol. Dengan penerapan teknologi pengenalan gambar berbasis kamera, sistem klasifikasi kendaraan dapat dilakukan secara otomatis melalui identifikasi langsung tipe kendaraan berdasarkan visual. Hal ini memungkinkan pengurangan biaya operasional, peningkatan kecepatan transaksi, dan akurasi klasifikasi yang lebih tinggi. Teknologi ini juga selaras dengan inisiatif digitalisasi transportasi di era modern, khususnya dalam mendukung pengelolaan tol yang efisien dan ramah pengguna (Hasan et al., 2020).

Namun, pengenalan tipe kendaraan tidak lepas dari tantangan teknis yang kompleks. Seperti variasi desain dan model kendaraan satu dengan lainnya, baik antar merk maupun satu merk menyebabkan kesulitan dalam membedakan fitur visual yang serupa. Pembaruan model dari waktu ke waktu pada kendaraan oleh produsen membuat model pada dataset yang lama tidak mengenali kendaraan keluaran terbaru. Dan juga kondisi pengambilan gambar yang beragam dari sudut pandang, kondisi pencahayaan hingga kualitas kamera yang digunakan yang mempengaruhi kualitas visual yang diekstrak.

Untuk mengatasi tantangan tersebut, pendekatan berbasis Machine Learning, khususnya Deep Learning, menjadi solusi yang efektif. Salah satu arsitektur utama dalam Deep Learning, yaitu Convolutional Neural Networks (CNN), telah terbukti unggul dalam tugas-tugas klasifikasi gambar. CNN memiliki kemampuan untuk secara otomatis mengekstrak fitur penting dari gambar, seperti bentuk bodi, logo, dan pola unik lainnya, yang sangat penting dalam pengenalan tipe kendaraan.

Penelitian ini bertujuan untuk menciptakan sistem yang dapat mengklasifikasikan kendaraan secara real-time dengan akurasi tinggi, sehingga memberikan solusi efektif untuk permasalahan yang ada (Kumar & Saini, 2021). Penelitian ini tidak hanya memberikan kontribusi pada pengembangan teknologi pengenalan gambar, tetapi juga membuka peluang untuk aplikasi lebih lanjut dalam pengembangan sistem transportasi cerdas dan solusi otomotif modern.

1.2 Rumusan Masalah

Dari latar belakang yang telah kita jabarkan kita mendapati rumusan masalah untuk penelitian kita sebagai berikut:

1. Bagaimana mengembangkan sistem yang mampu mengenali jenis dan model kendaraan secara akurat dari gambar dalam berbagai kondisi, seperti variasi desain, sudut pandang, pencahayaan, dan kualitas gambar?
2. Bagaimana meningkatkan akurasi klasifikasi tipe kendaraan dengan memanfaatkan dataset yang relevan, yaitu Stanford Car Dataset dan hasil *scraping* Google dan Pinterest, melalui penerapan arsitektur Deep Learning, khususnya Convolutional Neural Networks (CNN) dan *Pre-Trained Model* seperti ResNet50 dan VGG19?
3. Bagaimana mengatasi tantangan dalam pembaharuan model kendaraan baru yang tidak terwakili dalam dataset lama untuk memastikan sistem tetap relevan dan efektif di masa mendatang?
4. Bagaimana teknologi pengenalan kendaraan dapat diterapkan secara praktis untuk mendukung sistem penggolongan kendaraan pada gerbang tol otomatis?

1.3 Tujuan Penelitian

Tujuan penelitian pengembangan model klasifikasi tipe kendaraan untuk sistem pengenalan gambar otomotif adalah:

1. Mengembangkan sistem yang mampu mengenali dan mengklasifikasikan jenis serta model kendaraan secara akurat dari gambar, dengan mengatasi tantangan variasi desain, sudut pandang, pencahayaan, dan kualitas gambar.
2. Menerapkan arsitektur Deep Learning, khususnya Convolutional Neural Networks (CNN) dan *Pre-Trained Model* seperti ResNet50 dan VGG19, untuk meningkatkan akurasi dalam klasifikasi tipe kendaraan.
3. Memanfaatkan data gabungan dari Stanford Car Dataset dan data hasil *scraping* untuk melatih model klasifikasi kendaraan, dengan fokus pada optimalisasi fitur penting seperti bentuk bodi dan pola visual lainnya.
4. Mengadaptasi sistem agar mampu mengenali model kendaraan baru, sehingga tetap relevan dengan pembaruan yang dilakukan oleh produsen kendaraan.
5. Mengeksplorasi potensi aplikasi teknologi pengenalan kendaraan dalam mendukung sistem penggolongan kendaraan pada gerbang tol otomatis.

BAB II

LANDASAN TEORI

2.1 Deep Learning

Deep Learning merupakan salah satu cabang dari Machine Learning yang meniru cara kerja otak manusia dalam memproses informasi seperti pengenalan gambar, pemrosesan *natural language*, dan pengenalan suara. Deep Learning menggunakan neural networks untuk mempelajari representasi data yang lebih abstrak dan kompleks. Salah satu struktur utama dalam Deep Learning adalah Convolutional Neural Network (CNN), yang digunakan untuk pengklasifikasian gambar dan pengenalan objek.

2.2 Convolutional Neural Networks

Convolutional Neural Networks (CNN) adalah salah satu teknik deep learning untuk pengolahan dan analisis gambar. Struktur CNN terdiri dari lapisan konvolusi, pooling, dan fully connected berperan penting dalam mengurangi dimensi data sambil mempertahankan karakteristik penting yang diperlukan untuk klasifikasi. CNN juga sangat efektif dalam mengenali pola visual seperti bentuk dan tekstur yang ada pada gambar. Struktur ini memungkinkan CNN untuk mengidentifikasi objek dalam gambar dan melakukan klasifikasi jenis kendaraan dengan sangat presisi, yang keduanya sangat penting untuk sistem pengenalan gambar kendaraan (Krizhevsky et al., 2012).

Selain itu, CNN juga memiliki kemampuan untuk menyesuaikan diri untuk bekerja dengan resolusi dan variasi objek yang berbeda. Kemampuan adaptasi ini ditingkatkan melalui penerapan teknik seperti augmentasi data dan transfer learning, yang memperkuat model CNN untuk hasil yang lebih konsisten dan tahan terhadap variasi pada data (Szegedy et al., 2015).

2.3 Stanford Cars Dataset

Stanford Cars Dataset adalah salah satu kumpulan data populer yang dirancang untuk tugas klasifikasi gambar pada domain pengenalan merek dan model kendaraan. Dataset ini berisi 16.185 gambar mobil yang dibagi menjadi 8.144 untuk pelatihan dan 8.041 untuk pengujian, mencakup 196 kategori berdasarkan merek, model, dan tahun pembuatan kendaraan. Salah satu tantangan utama dataset ini adalah ketidakseimbangan data, di mana setiap label memiliki jumlah data pelatihan yang relatif sedikit, membuat proses pembelajaran mendalam menjadi lebih menantang. Selain itu, setiap gambar telah diberi anotasi dengan label spesifik serta area bounding box, yang memungkinkan penggunaan metode preprocessing seperti cropping untuk meningkatkan kualitas data input.

Dalam penelitian pengolahan gambar, Stanford Cars Dataset sering digunakan untuk menguji algoritma jaringan syaraf konvolusional (CNN) karena kompleksitasnya dalam menangkap detail visual seperti bentuk, pola, dan warna. Dataset ini juga mendukung studi tentang performa model deep learning populer, seperti DenseNet, ResNet, dan Inception, dalam hal akurasi klasifikasi dan efisiensi komputasi. Fokus utama pada dataset ini adalah memastikan bahwa model tidak hanya dapat mengklasifikasikan dengan akurat tetapi juga dapat diimplementasikan secara efisien dalam sistem tertanam, seperti yang ditunjukkan pada eksperimen menggunakan Jetson Nano pada penelitian ini (Wibisono, Ari, et al. 2023)

2.4 VGG19

VGG19 adalah arsitektur jaringan Convolutional Neural Network (CNN) dengan model yang terdiri dari 19 lapisan, termasuk 16 lapisan konvolusi dan 3 lapisan fully connected, penggunaan kernel berukuran 3×3 pada setiap lapisan konvolusi. Karakteristik utama dari VGG19 adalah desainnya yang mendalam dan sederhana, dengan semua lapisan konvolusi memiliki struktur yang seragam. Model ini menggunakan max pooling untuk mengurangi dimensi data secara bertahap, sambil mempertahankan informasi penting dalam fitur spasial. VGG19 sangat efektif untuk tugas-tugas klasifikasi gambar, ekstraksi fitur, dan transfer learning karena kemampuannya menghasilkan representasi fitur yang kaya. Namun, model ini memiliki kekurangan dalam efisiensi memori karena jumlah parameter yang sangat besar (Simonyan & Zisserman, 2014).

2.5 RESNET50

ResNet50 adalah salah satu varian dari arsitektur Residual Network (ResNet) yang dirancang untuk mengatasi masalah degradasi performa yang sering muncul pada jaringan yang sangat dalam. ResNet50 terdiri dari 50 lapisan, termasuk lapisan residual yang memungkinkan informasi melewati lapisan melalui shortcut connection atau skip connection. Struktur ini membantu model mempertahankan performa tinggi meskipun jaringan semakin dalam. ResNet50 memanfaatkan blok bottleneck, yang terdiri dari kombinasi lapisan konvolusi 1×1 , 3×3 , dan 1×1 , untuk mengurangi jumlah parameter tanpa mengorbankan kemampuan ekstraksi fitur. ResNet50 banyak digunakan dalam tugas-tugas klasifikasi, deteksi objek, dan transfer learning, dengan performa yang konsisten dan efisiensi tinggi (He et al., 2015).

2.6 Computer Vision

Computer Vision adalah cabang dari kecerdasan buatan yang berfokus pada pengembangan teknik untuk memungkinkan komputer "melihat" dan memahami dunia visual, seperti yang dilakukan oleh manusia. Pada dasarnya, computer vision bertujuan untuk mengekstrak informasi dari gambar dan video untuk kemudian digunakan dalam pengambilan keputusan otomatis. Salah satu bidang utama dalam computer vision adalah klasifikasi objek, di mana sistem berusaha mengenali objek yang ada dalam gambar dan mengklasifikasikannya ke dalam kategori tertentu. Untuk melakukan ini, berbagai teknik dan algoritma digunakan, terutama dalam konteks pengenalan pola dan analisis citra, yang melibatkan ekstraksi fitur dari gambar dan penerapannya dalam model pembelajaran mesin beberapa tahun terakhir, kemajuan besar dalam bidang computer vision didorong oleh perkembangan algoritma deep learning, khususnya jaringan saraf konvolusional (CNN). CNN, yang telah terbukti sangat efektif dalam pengolahan gambar, menggunakan lapisan-lapisan konvolusional untuk mendeteksi fitur visual hierarkis dalam gambar, seperti tepi, tekstur, dan bentuk. Metode ini sangat kuat dalam mengklasifikasikan gambar dan juga memungkinkan sistem untuk belajar secara otomatis dari data yang sangat besar, seperti yang ada pada Stanford Cars Dataset. Dengan menggunakan CNN, model dapat mengenali berbagai jenis objek atau kendaraan dalam gambar secara akurat, bahkan dalam kondisi yang penuh dengan variasi seperti pencahayaan, sudut pandang, atau latar belakang yang berbeda. Hal ini menjadikan computer vision dan teknik deep learning seperti CNN sangat relevan untuk aplikasi seperti pengenalan merek dan model mobil secara otomatis (Voulodimos, A. et al 2018).

BAB III

METODOLOGI PENELITIAN

3.1 Sumber Data

Penelitian ini menggunakan gabungan dataset Stanford Cars Dataset dan melakukan *scrapping* platform Pinterest dan Google, yang merupakan kumpulan data yang terdiri dari 1.961 gambar kendaraan dengan 3 kategori. Dataset ini dibagi menjadi dua bagian: 1.568 gambar untuk pelatihan dan 393 gambar untuk pengujian. Setiap gambar dalam dataset ini memiliki anotasi yang mencakup label kategori kendaraan serta area bounding box yang digunakan untuk preprocessing seperti cropping.

Berikut kategori kendaraan yang ada pada dataset:

Tabel 1. Variabel Penelitian

Tipe Kendaraan	Keterangan
Mobil Biasa	Terdiri dari mobil jenis SUV, Sedan, dan Hatchback
Truck	Kendaraan besar yang digunakan untuk mengangkut barang berat.
Bus	Kendaraan besar yang digunakan untuk membawa penumpang.

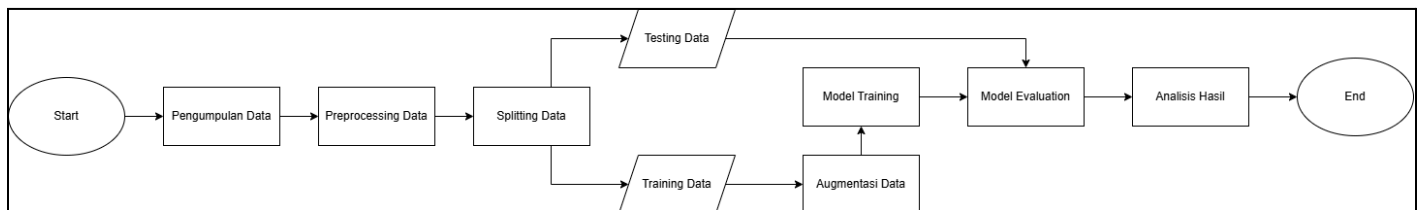
Dataset ini juga memiliki tantangan dalam hal ketidakseimbangan data antar kategori, di mana beberapa kategori kendaraan memiliki lebih sedikit gambar dibandingkan dengan kategori lainnya. Oleh karena itu, augmentasi gambar dilakukan untuk mengatasi masalah ini.

3.2 Metodologi

Metodologi penelitian ini mencakup beberapa tahap, dimulai dari preprocessing data, pemilihan dan pelatihan model, hingga evaluasi performa model. Berikut penjelasan rinci dari setiap langkah yang dilakukan:

3.2.1 Flowchart

Gambar 1. Flowchart Metodologi



Proses pengembangan model dimulai dengan pengumpulan data yang diambil dari Stanford Cars Dataset, Pinterest, dan Google. Tahap berikutnya adalah preprocessing pada data, di mana data dibersihkan dan diformat agar siap digunakan. Selanjutnya, data dibagi menjadi dua bagian: satu untuk pelatihan dan satu lagi untuk pengujian. Data pelatihan digunakan untuk melatih model dan dilakukan augmentasi untuk meningkatkan variasi dan kualitas data. Setelah model dilatih, langkah berikutnya adalah evaluasi untuk menilai kinerjanya. Terakhir, analisis akhir dilakukan untuk meninjau hasil dan performa model sebelum proses dianggap selesai.

3.2.2 Preprocessing Data

Pada tahap preprocessing, gambar-gambar dari dataset ini mengalami beberapa langkah pemrosesan agar siap digunakan oleh model. Langkah-langkah preprocessing meliputi:

- **Rescaling:** Semua nilai pixel gambar dinormalisasi dengan membaginya dengan 255 agar berada dalam rentang $[0, 1]$. Ini bertujuan untuk mempercepat proses pelatihan dan meningkatkan kestabilan numerik model.
- **Resize:** Gambar diubah ukurannya menjadi 224x224 piksel, karena ukuran ini lebih sesuai dengan arsitektur model yang digunakan.
- **Konversi ke RGB :** Gambar dikonversi ke RGB untuk memastikan konsistensi format warna, terutama jika gambar asli mungkin memiliki mode warna yang berbeda.
- **Konversi ke array NumPy :** Mengonversi objek PIL Image menjadi array NumPy untuk memudahkan manipulasi data numerik.

3.2.3 Splitting Data

Pada tahap ini, dataset dibagi menjadi dua subset: data pelatihan dan data pengujian. Data pelatihan terdiri dari 1.568 gambar untuk pelatihan dan 393 gambar untuk pengujian. Sebelum digunakan, data dibagi menggunakan parameter *test_split=0.2*, yang membagi 80% data untuk pelatihan dan 20% untuk pengujian. Teknik ini memastikan bahwa model dapat memantau kinerjanya dengan baik selama pelatihan.

3.2.4 Augmentasi Data

Untuk meningkatkan jumlah data pelatihan dan mengatasi masalah ketidakseimbangan data, augmentasi gambar dilakukan. Augmentasi ini bertujuan untuk menambah keragaman gambar yang dapat membantu model belajar lebih baik. Teknik augmentasi yang diterapkan adalah:

- **rotation_range=20 :** Gambar akan diputar secara acak hingga 20 derajat.
- **width_shift_range=0.2 :** Gambar akan digeser secara horizontal secara acak hingga 20% dari lebar total.
- **height_shift_range=0.2 :** Gambar akan digeser secara vertikal secara acak hingga 20% dari tinggi total.
- **zoom_range=0.2 :** Gambar akan diperbesar atau diperkecil secara acak hingga 20%.
- **horizontal_flip=True :** Gambar akan dibalik secara horizontal secara acak.

3.2.5 Model ResNet50

ResNet50 adalah salah satu arsitektur deep neural network yang populer untuk tugas pengenalan gambar dan berbagai tugas lain dalam computer vision. ResNet50 diperkenalkan dalam makalah "Deep Residual Learning for Image Recognition" oleh He et al. pada tahun 2015.

Model yang menggunakan ResNet50 pre-trained memanfaatkan bobot yang telah dilatih pada dataset ImageNet. Bagian akhir model asli dihapus (*include_top=False*) untuk memungkinkan penambahan lapisan klasifikasi khusus. Input gambar berformat RGB dengan resolusi 224x224 piksel, dan semua lapisan dalam base model dibekukan (*trainable=False*) untuk mempertahankan parameter yang telah dipelajari sebelumnya. Pendekatan ini membantu mempercepat pelatihan dan mencegah overfitting, terutama untuk dataset kecil, karena fitur yang dipelajari dari ImageNet cukup umum untuk digunakan.

Lapisan klasifikasi baru ditambahkan di atas base model. Pertama, digunakan GlobalAveragePooling2D untuk meratakan output spasial ResNet50 menjadi vektor fitur, diikuti dengan Dense berisi 256 unit dan aktivasi ReLU untuk mendeteksi pola kompleks. Lapisan Dropout dengan tingkat penghapusan 50% digunakan untuk mencegah overfitting, dan lapisan terakhir berupa Dense dengan 3 unit serta aktivasi softmax digunakan untuk klasifikasi 3 kelas mobil, menghasilkan probabilitas untuk setiap kelas.

Model dikompilasi menggunakan optimizer Adam yang stabil dan cocok untuk transfer learning. Fungsi loss yang digunakan adalah Sparse Categorical Cross Entropy untuk klasifikasi multi-kelas dengan label integer, dan evaluasi performa dilakukan menggunakan metrik accuracy. Pendekatan ini memanfaatkan ResNet50 sebagai extractor fitur untuk mengenali pola umum, sementara lapisan tambahan difokuskan pada fitur spesifik dari dataset. Dengan demikian, model ini efisien untuk meningkatkan akurasi klasifikasi pada dataset kecil.

3.2.6 Model VGG19

VGG19 digunakan sebagai model pre-trained untuk tugas klasifikasi 3 kelas. Model VGG19 dipilih dengan bobot yang telah dilatih sebelumnya pada dataset ImageNet (weights='imagenet'). Lapisan klasifikasi bawaan model dihapus (include_top=False) agar dapat ditambahkan lapisan baru yang sesuai dengan dataset Anda. Input gambar diatur berformat RGB dengan resolusi 224x224 piksel. Selanjutnya, semua lapisan dalam base model dibekukan (trainable=False) untuk mencegah parameter model pre-trained diperbarui selama pelatihan, sehingga hanya lapisan tambahan yang dilatih. Hal ini memanfaatkan fitur umum yang telah dipelajari oleh VGG19 sekaligus menghemat waktu pelatihan dan mencegah overfitting.

Setelah membekukan base model, ditambahkan beberapa lapisan baru untuk klasifikasi. Lapisan pertama adalah GlobalAveragePooling2D, yang meratakan output spasial VGG19 menjadi vektor fitur. Kemudian, sebuah lapisan Dense dengan 256 unit dan aktivasi ReLU digunakan untuk mendeteksi pola kompleks dari fitur yang dihasilkan. Untuk mengurangi risiko overfitting, ditambahkan lapisan Dropout dengan tingkat penghapusan 50%. Akhirnya, lapisan output berupa Dense dengan 3 unit dan aktivasi softmax ditambahkan untuk menghasilkan probabilitas untuk setiap kelas, karena model bertujuan mengklasifikasikan gambar ke dalam 3 kategori.

Model ini dikompilasi menggunakan optimizer Adam, yang merupakan metode adaptif untuk mempercepat konvergensi dan cocok untuk transfer learning. Fungsi loss yang digunakan adalah Sparse Categorical Cross Entropy, yang cocok untuk tugas klasifikasi multi-kelas dengan label integer, dan evaluasi performa dilakukan menggunakan metrik accuracy. Secara keseluruhan, pendekatan ini menggunakan VGG19 sebagai extractor fitur yang memanfaatkan pengetahuan dari dataset besar (ImageNet), sedangkan lapisan tambahan dirancang untuk belajar pola spesifik pada dataset Anda. Pendekatan ini efisien untuk mempercepat pelatihan dan meningkatkan performa klasifikasi, terutama pada dataset kecil.

3.2.7 Model CNN

Arsitektur model yang digunakan dalam penelitian ini adalah *Convolutional Neural Network* (CNN). CNN adalah jenis jaringan saraf yang efektif untuk pengolahan gambar.. Setelah pembangunan arsitektur CNN selesai, langkah berikutnya adalah kompilasi model. Proses kompilasi ini melibatkan penentuan optimizer, loss function, dan metrics yang akan digunakan untuk melatih model. Dalam hal ini, kita memilih optimizer Adam, yang dikenal efektif dalam pembelajaran mendalam karena dapat menyesuaikan tingkat pembelajaran untuk setiap parameter.

- Optimizer '*adam*': Adam adalah optimizer yang populer dan efektif yang menggabungkan keuntungan dari dua metode sebelumnya, yaitu AdaGrad dan RMSProp. Adam menyesuaikan tingkat pembelajaran berdasarkan rata-rata bergerak dari gradien pertama dan kedua. Dengan ini, model dapat beradaptasi dengan lebih baik terhadap perubahan dalam data selama pelatihan.
- Loss Function '*categorical_crossentropy*': Fungsi loss ini digunakan untuk masalah klasifikasi multi-kelas, di mana setiap gambar akan dikategorikan ke dalam salah satu dari empat kelas. Kategori tersebut adalah SUV, Sedan, Hatchback, dan Truck.
- Metrics '*accuracy*': Akurasi adalah metrik yang digunakan untuk mengevaluasi kinerja model selama pelatihan. Ini mengukur persentase gambar yang berhasil diklasifikasikan dengan benar oleh model.

Gambar 2. Model CNN

Model: "sequential_3"		
Layer (type)	Output Shape	Param #
conv2d_12 (Conv2D)	(None, 222, 222, 32)	896
max_pooling2d_9 (MaxPooling2D)	(None, 111, 111, 32)	0
conv2d_13 (Conv2D)	(None, 109, 109, 64)	18,496
max_pooling2d_10 (MaxPooling2D)	(None, 54, 54, 64)	0
conv2d_14 (Conv2D)	(None, 52, 52, 128)	73,856
max_pooling2d_11 (MaxPooling2D)	(None, 26, 26, 128)	0
flatten_2 (Flatten)	(None, 86528)	0
dense_6 (Dense)	(None, 128)	11,075,712
dropout_6 (Dropout)	(None, 128)	0
dense_7 (Dense)	(None, 4)	516
Total params: 11,169,476 (42.61 MB)		
Trainable params: 11,169,476 (42.61 MB)		
Non-trainable params: 0 (0.00 B)		

3.2.8 Training Data

Proses pelatihan model dilakukan dengan menggunakan data pelatihan yang telah diproses, termasuk penerapan augmentasi gambar untuk meningkatkan variasi data. Augmentasi gambar, seperti rotasi, flipping, dan zoom, diterapkan secara otomatis untuk meningkatkan kemampuan generalisasi model. Pelatihan dilakukan selama 10 epoch, menggunakan data pelatihan dari generator data yang telah diatur.

Data validasi digunakan untuk memantau performa model di setiap epoch. Hasilnya mencatat nilai loss dan accuracy baik untuk data pelatihan maupun validasi. Penurunan nilai loss dan peningkatan nilai accuracy diharapkan menunjukkan bahwa model semakin mampu memprediksi dengan baik.

3.2.9 Testing Dataset

Setelah pelatihan data selesai, model diuji menggunakan dataset uji yang tidak pernah digunakan selama pelatihan maupun validasi. Dataset uji memberikan gambaran performa generalisasi model terhadap data baru.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Percobaan Model

Pada tahap ini, dilakukan percobaan menggunakan berbagai arsitektur model untuk menentukan model terbaik dalam proses klasifikasi gambar. Model yang dicoba meliputi EfficientNet, CNN-LSTM, MobileNet, dan Vision Transformer (ViT), namun model tersebut tidak digunakan karena berbagai alasan. Model yang dipilih untuk implementasi akhir adalah ResNet50, VGG19, dan CNN, karena memberikan akurasi dan efisiensi terbaik.

Percobaan Model yang Tidak Digunakan:

1. EfficientNet
Meskipun menghasilkan akurasi baik, waktu komputasi yang lama membuat EfficientNet kurang ideal untuk kasus ini.
2. CNN-LSTM
CNN-LSTM tidak relevan untuk data gambar statis, dan waktu pelatihan yang lama tidak sebanding dengan peningkatan akurasi yang diperoleh.
3. MobileNet
MobileNet efisien secara komputasi, tetapi akurasinya lebih rendah dibandingkan model lain, terutama pada kelas dengan fitur visual serupa.
4. Vision Transformer (ViT)
ViT membutuhkan sumber daya komputasi yang besar dan waktu pelatihan lama, sementara akurasinya tidak optimal serta penggunaannya yang sangat berat.

Model yang Dipilih:

1. ResNet50
Memberikan akurasi tinggi dan generalisasi baik melalui arsitektur residual, meskipun waktu pelatihannya lebih lama.
2. VGG19
Menghasilkan akurasi tinggi dengan arsitektur sederhana yang stabil dan efektif untuk fitur visual yang jelas.
3. CNN (Convolutional Neural Network)
Sebagai baseline, CNN memberikan akurasi baik dengan waktu pelatihan yang singkat, meskipun di bawah ResNet50 dan VGG19.

ResNet50, VGG19, dan CNN dipilih untuk implementasi akhir karena akurasi, generalisasi, dan efisiensinya. Model seperti EfficientNet, CNN-LSTM, MobileNet, dan ViT tidak digunakan karena keterbatasan seperti waktu komputasi lama, akurasi rendah, atau relevansi yang kurang dengan tugas ini. Kombinasi model terpilih memberikan keseimbangan optimal untuk proyek ini.

4.2 Hasil Training

Pelatihan dilakukan menggunakan platform Google Colab dengan runtime GPU T4, yang memberikan akselerasi signifikan dalam pemrosesan data untuk deep learning. Kami

menggunakan 10 epoch untuk melatih tiga model berbeda, yaitu ResNet50, VGG19, dan CNN. Model ini dibuat untuk mengklasifikasikan tiga tipe kendaraan, dengan setiap model dievaluasi berdasarkan akurasi dan loss-nya.

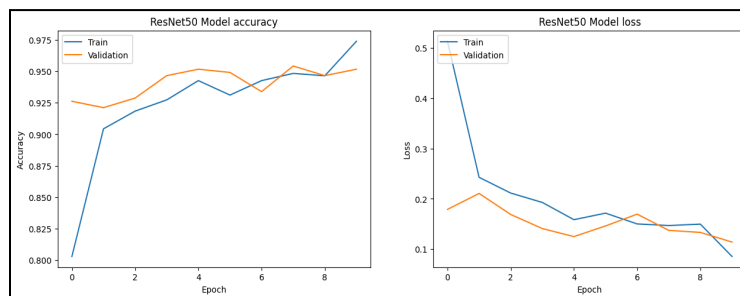
Pemanfaatan penggunaan GPU mempercepat proses pelatihan jaringan saraf mendalam, terutama untuk arsitektur model yang kompleks seperti ResNet50 dan VGG19, yang membutuhkan banyak komputasi untuk proses forward dan backward propagation.

Parameter Training:

- Jumlah epoch: 10, yang dirancang untuk menghindari overfitting pada dataset yang tergolong kecil.
- Batch size: Diasumsikan nilai default (32) atau disesuaikan dengan kapasitas memori GPU.

4.2.1 Hasil Training Model ResNet50

Gambar 3. Grafik Model ResNet50



1. Grafik Akurasi Training:

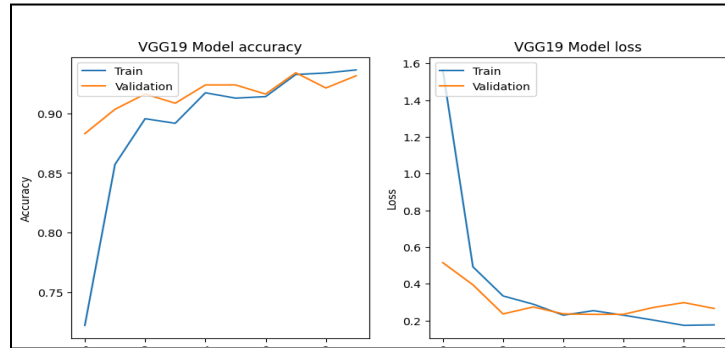
- Garis biru (Train) menunjukkan akurasi pada data pelatihan, yang meningkat secara konsisten hingga mencapai sekitar 97%.
- Garis oranye (Validation) menunjukkan akurasi pada data validasi, yang juga meningkat tetapi dengan fluktuasi lebih besar, mencapai sekitar 95%.

2. Grafik Loss:

- Garis biru (Train) menunjukkan nilai loss pada data pelatihan, yang menurun seiring bertambahnya epoch, menunjukkan bahwa model belajar dengan baik.
- Garis oranye (Validation) juga menunjukkan penurunan loss, tetapi dengan nilai yang lebih rendah dibandingkan dengan data pelatihan, menunjukkan bahwa model tidak mengalami overfitting.

4.2.2 Hasil Training Model VGG19

Gambar 4. Grafik Model VGG19



1. Grafik Akurasi Training:

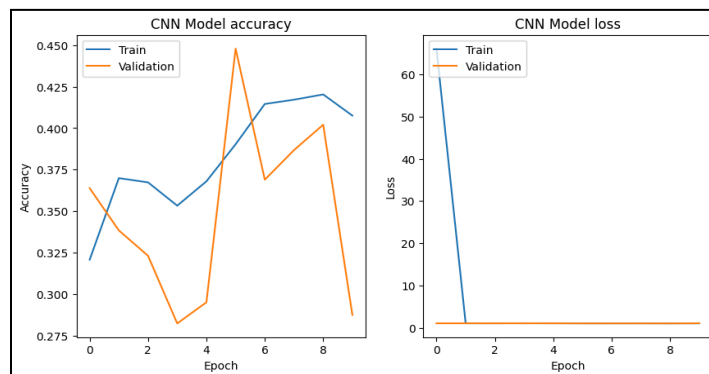
- Garis biru (Train) menunjukkan akurasi pada data pelatihan, yang meningkat secara signifikan, mencapai sekitar 92%.
- Garis oranye (Validation) menunjukkan akurasi pada data validasi, yang lebih rendah dan mengalami stagnasi di sekitar 85%, yang menunjukkan bahwa model mungkin tidak generalisasi dengan baik pada data yang tidak terlihat.

2. Grafik Loss:

- Garis biru (Train) menunjukkan nilai loss pada data pelatihan, yang menurun tajam hingga mencapai nilai yang rendah, menunjukkan pembelajaran yang efektif.
- Garis oranye (Validation) menunjukkan penurunan loss yang lebih lambat, tetapi tetap lebih tinggi dibandingkan dengan loss pelatihan, menunjukkan potensi overfitting.

4.2.3 Hasil Training Model CNN

Gambar 5. Grafik Model CNN



1. Grafik Akurasi:

- Garis biru (Train) menunjukkan akurasi pada data pelatihan, yang memiliki fluktuasi yang signifikan, dengan nilai tertinggi sekitar 0.45 dan terendah sekitar 0.275.

- Garis oranye (Validation) menunjukkan akurasi pada data validasi, yang tetap rendah dan juga fluktuatif, dengan nilai tertinggi sekitar 0.375.
2. Grafik Loss:
- Garis biru (Train) menunjukkan nilai loss pada data pelatihan, yang bervariasi dan memiliki nilai yang jauh lebih tinggi, mencapai sekitar 60, yang menunjukkan bahwa model mungkin tidak belajar dengan baik.
 - Garis oranye (Validation) menunjukkan loss yang juga tinggi, tetapi dengan nilai yang lebih konsisten dibandingkan loss pelatihan.

4.3 Hasil Testing

4.3.1 Hasil Testing Model ResNet50

Tabel 2. *Classification Report ResNet50*

ResNet50	Precision	Recall	F1-Score	Support
Mobil Biasa	0.96	0.98	0.97	140
Bus	0.95	0.93	0.94	105
Truck	0.94	0.94	0.94	148
Accuracy			0.95	393
Macro Avg	0.95	0.95	0.95	393
Weight Avg	0.95	0.95	0.95	393

Dari tabel diatas, model ResNet50 menunjukkan kinerja yang sangat baik dengan nilai akurasi sebesar 95% dan juga nilai precision dan recall yang sangat tinggi pada ke-3 kelas. Hal ini bisa kita simpulkan bahwa model ResNet50 sudah sangat baik dalam mengklasifikasikan 3 kelas.

4.3.2 Hasil Testing Model VGG19

Tabel 3. *Classification Report VGG19*

VGG19	Precision	Recall	F1-Score	Support
Mobil Biasa	0.97	0.99	0.98	140
Bus	0.84	0.97	0.90	105
Truck	0.98	0.84	0.91	148
Accuracy			0.93	393
Macro Avg	0.93	0.94	0.93	393
Weight Avg	0.94	0.93	0.93	393

Dari tabel diatas, model VGG19 menunjukkan kinerja yang sangat baik dengan nilai akurasi sebesar 93% dan juga nilai precision dan recall yang tinggi pada kelas “Mobil Biasa”. Secara keseluruhan model ini menunjukkan kemampuan yang baik dalam mengklasifikasikan data.

4.3.3 Hasil Testing Model CNN

Tabel 4. *Classification Report CNN*

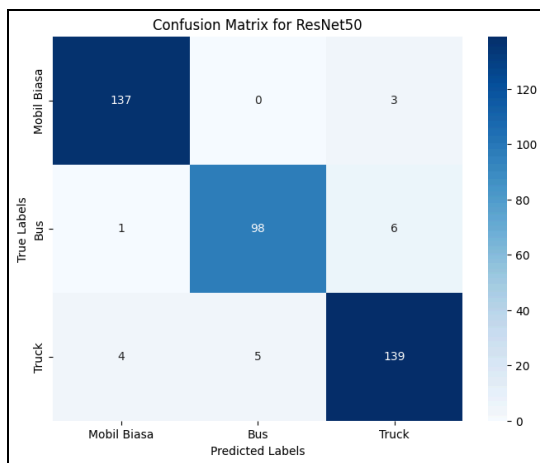
CNN	Precision	Recall	F1-Score	Support
Mobil Biasa	0.55	0.04	0.08	140
Bus	0.27	0.95	0.42	105
Truck	0.78	0.05	0.09	148
Accuracy			0.29	393
Macro Avg	0.53	0.35	0.20	393
Weight Avg	0.56	0.29	0.17	393

Dari tabel diatas, model CNN menunjukkan kinerja yang kurang baik dengan nilai akurasi sebesar 29%. Nilai precision dan f1-score yang rendah menunjukkan model kesulitan dalam mengklasifikasikan data dengan benar.

4.4 Confusion Matrix

4.4.1 Confusion Matrix ResNet50

Gambar 6. Confusion Matrix Resnet50

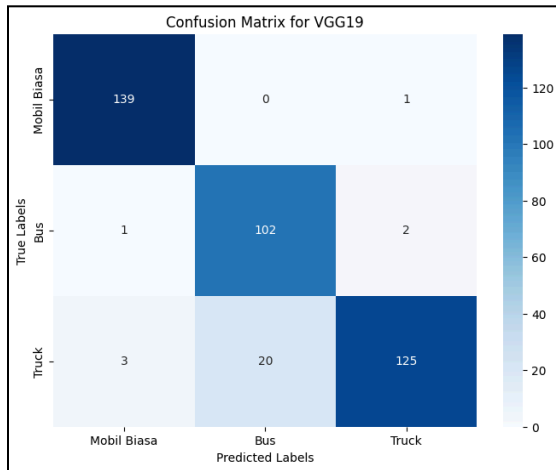


Confusion Matrix menunjukkan bahwa model ResNet50 memiliki performa yang baik dalam mengklasifikasikan tiga jenis kendaraan: Mobil Biasa, Bus, dan Truck. Model berhasil mengklasifikasikan 137 gambar Mobil truck dengan benar. Kesalahan klasifikasi sebagian besar terjadi pada Bus dan Truck yang saling tertukar, seperti 6 gambar Bus diklasifikasikan sebagai Truck dan 5 gambar Truck diklasifikasikan sebagai Bus, kemungkinan karena kemiripan bentuk dan ukuran. Secara keseluruhan, model menunjukkan kemampuan klasifikasi yang baik, namun masih dapat ditingkatkan

melalui penambahan data atau fine-tuning untuk mengurangi kesalahan pada kelas yang saling mirip.

4.4.2 Confusion Matrix VGG19

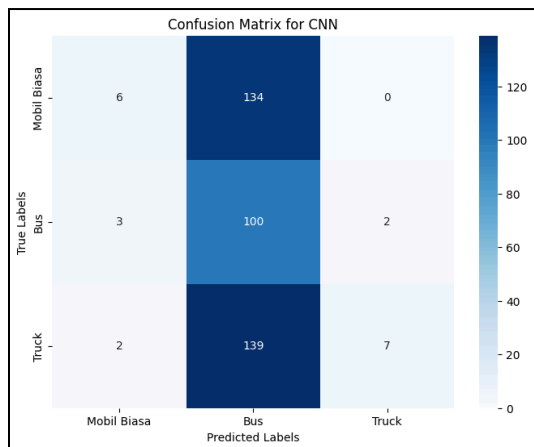
Gambar 7. Confusion Matrix VGG19



Confusion Matrix untuk model VGG19 menunjukkan bahwa model ini berhasil mengklasifikasikan kategori "Mobil Biasa," "Bus," dan "Truck" dengan baik, terutama pada "Mobil Biasa" yang memiliki 139 prediksi benar. Namun, terdapat beberapa kesalahan klasifikasi, dengan "Mobil Biasa" salah dikategorikan sebagai "Bus" dan "Truck," serta "Bus" sering salah sebagai "Truck." Ini menunjukkan bahwa meskipun akurasi model tinggi, masih ada kebingungan antara kategori "Bus" dan "Truck," yang menunjukkan perlunya perbaikan lebih lanjut dalam model atau data pelatihan untuk meningkatkan kinerjanya.

4.4.3 Confusion Matrix CNN

Gambar 8. Confusion Matrix CNN



Confusion Matrix untuk model CNN menunjukkan bahwa model ini mengalami kesulitan dalam klasifikasi, dengan "Mobil Biasa" hanya memiliki 6 prediksi benar dan banyak salah dikategorikan sebagai "Bus" (134) dan "Truck" (0). Sementara itu, kategori "Bus" memiliki 100 prediksi benar, tetapi juga mengalami beberapa kesalahan dengan 3 salah sebagai "Mobil Biasa" dan 2 sebagai "Truck." Kategori "Truck" menunjukkan kinerja yang lebih baik dengan 139 prediksi benar, namun masih terdapat 7 kesalahan klasifikasi. Secara keseluruhan, model CNN menunjukkan kinerja yang kurang optimal, terutama dalam membedakan "Mobil Biasa" dari kategori lainnya.

4.5 Implementasi Model

Melakukan percobaan untuk menerapkan model yang sudah dibuat dengan cara mencoba mengklasifikasikan gambar lain selain data test.

Tabel 5. Implementasi Model

Sampel Gambar		
		
Gambar mobil dalam kondisi gambar terdapat sedikit <i>noise</i> .	Gambar mobil dalam kondisi pencahayaan terang.	Gambar truk dalam kondisi malam hari.
Hasil Prediksi		
ResNet50: Bus	ResNet50: Mobil biasa	ResNet50: Truk
VGG19: Bus	VGG19: Mobil biasa	VGG19: Truk
CNN: Bus	CNN: Mobil biasa	CNN: Bus

Model seperti ResNet50 dan VGG19 menunjukkan kinerja lebih baik dalam mengenali gambar dengan pencahayaan terang, terlihat dari keberhasilannya memprediksi mobil di dalam ruangan dan truk di malam hari. Sebaliknya, CNN lebih rentan terhadap kesalahan, seperti salah mengklasifikasikan truk menjadi bus, kemungkinan akibat kemiripan bentuk visual. *Noise* dan pencahayaan yang kurang ideal juga mempengaruhi akurasi semua model. Peningkatan preprocessing, seperti pengurangan noise, dan augmentasi data dapat membantu meningkatkan performa model dalam berbagai kondisi.

BAB V

KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan model klasifikasi tipe kendaraan menggunakan teknik Deep Learning, khususnya arsitektur Convolutional Neural Networks (CNN), ResNet50, dan VGG19. Model ini dirancang untuk mengatasi rumusan masalah yang telah diidentifikasi, termasuk pengembangan sistem yang mampu mengenali jenis dan model kendaraan secara akurat dalam berbagai kondisi, seperti variasi desain, sudut pandang, pencahayaan, dan kualitas gambar.

Hasil yang diperoleh menunjukkan bahwa model ResNet50 memiliki akurasi tertinggi sebesar 95%, diikuti oleh VGG19 dengan akurasi 93%, sementara model CNN menunjukkan performa yang kurang memuaskan dengan akurasi hanya 29%. Untuk mencapai tujuan penelitian, yaitu meningkatkan akurasi klasifikasi tipe kendaraan melalui pemanfaatan dataset yang relevan, model berhasil dilatih menggunakan gabungan dari Stanford Cars Dataset dan hasil scraping. Ini menunjukkan efektivitas penggunaan arsitektur Deep Learning dalam meningkatkan akurasi.

Meskipun demikian, tantangan seperti ketidakseimbangan data dan kesulitan dalam membedakan antara kategori kendaraan yang mirip tetap ada. Penelitian ini juga berhasil menunjukkan potensi aplikasi teknologi pengenalan kendaraan dalam mendukung sistem penggolongan kendaraan pada gerbang tol otomatis, meskipun masih perlu pengembangan lebih lanjut untuk memastikan sistem tetap relevan dengan pembaruan model kendaraan baru.

Saran untuk penelitian selanjutnya sebagai berikut:

1. Melakukan augmentasi data yang lebih beragam untuk meningkatkan jumlah dan variasi data pelatihan, sehingga model dapat belajar dari lebih banyak kondisi yang berbeda.
2. Mempertimbangkan untuk mencoba arsitektur model lain yang lebih canggih, seperti Vision Transformer (ViT), untuk melihat apakah dapat memberikan hasil yang lebih baik.
3. Melakukan fine-tuning pada model yang sudah ada, terutama pada lapisan terakhir, untuk meningkatkan akurasi klasifikasi pada kategori yang sulit dibedakan.
4. Melakukan pengujian model di lingkungan nyata untuk mengevaluasi kinerjanya dalam situasi dunia nyata, seperti variasi pencahayaan dan sudut pandang yang berbeda.
5. Mengumpulkan lebih banyak data dari berbagai sumber untuk mengatasi masalah ketidakseimbangan dan memastikan model tetap relevan dengan model kendaraan terbaru.

Secara keseluruhan, penelitian ini memberikan kontribusi signifikan terhadap pengembangan teknologi pengenalan gambar otomotif dan membuka peluang untuk aplikasi lebih lanjut dalam sistem penentuan golongan kendaraan untuk Gerbang Tol Otomatis.

DAFTAR PUSTAKA

Almukadi, W. S., & Rekabdar, M. (2023). Vehicle Classification Using Deep Feature Fusion and Genetic Algorithms. *Electronics*, 12(2), 280. Diakses pada 19 November 2024, dari <https://www.mdpi.com/2079-9292/12/2/280>

Gholamhosseinian, A., & Seitz, J. (2021). Vehicle Classification in Intelligent Transport Systems: An Overview, Methods and Software Perspective. *IEEE Open Journal of Intelligent Transportation Systems*, 2(July), 173–194. <https://doi.org/10.1109/OJITS.2021.3096756>

Hattiya, T., Dittakan, K., & Musikasuwan, S. (2021). Diabetic Retinopathy Detection using Convolutional Neural Network: A Comparative Study on Different Architectures. *Engineering Access*, 7(1), 50–60. <https://doi.org/10.14456/mijet.2021.8>

Hicham, B., Ahmed, A., & Mohammed, M. (2018, October). Vehicle type classification using convolutional neural network. In *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)* (pp. 313-316). IEEE.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84–90. <https://doi.org/10.1145/3065386>

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June-2015, 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>

Wibisono, Ari, et al. "Embedded Deep Learning System for Classification of Car Make and Model." *Jurnal Ilmu Komputer dan Informasi* 16.1 (2023): 69-75.


Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018(1), 7068349.

Zha, W., Liu, Y., Wan, Y., Luo, R., Li, D., Yang, S., & Xu, Y. (2022). Forecasting monthly gas field production based on the CNN-LSTM model. *Energy*, 260, 124889.

Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O. (2021). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3), 107–115. <https://doi.org/10.1145/3446776>

LAMPIRAN

Kode:

 DM2_A2_014_034_046_094.ipynb

Data:

 Data_UAS