

LAPORAN PROJECT UAS DEEP LEARNING

**IMPLEMENTASI KLASIFIKASI KENDARAAN BERMOTOR DI
UNIVERSITAS BENGKULU**



Disusun Oleh :

- 1. Nazir Mahmudi Lubis (G1A021013)**
- 2. Kahfi Zairan Maulana (G1A021041)**
- 3. Gilang Atila Ilham (G1A021075)**

Dosen Pengampu :

Arie Vatesia, S.T., M.T.I., Ph.D

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS BENGKULU**

2024

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan pertumbuhan jumlah kendaraan bermotor yang terus meningkat setiap tahunnya, kebutuhan untuk mengelola, mengatur, dan memantau pergerakan kendaraan menjadi semakin penting. Klasifikasi kendaraan bermotor adalah proses untuk mengidentifikasi jenis kendaraan berdasarkan karakteristik tertentu, seperti ukuran, bentuk, atau jenis penggunaan (misalnya mobil pribadi, truk, atau sepeda motor). Proses ini memiliki peran krusial dalam berbagai bidang, termasuk transportasi, manajemen lalu lintas, serta pengambilan keputusan terkait kebijakan publik.

Penerapan teknologi klasifikasi kendaraan bermotor dapat memberikan manfaat yang signifikan, seperti:

1. Efisiensi Sistem Lalu Lintas: Dengan mengetahui jenis kendaraan yang melintasi jalan tertentu, pengelola lalu lintas dapat mengoptimalkan distribusi kendaraan dan meminimalkan kemacetan.
2. Keamanan Jalan: Identifikasi kendaraan dapat membantu penegakan hukum, seperti mendeteksi pelanggaran lalu lintas atau kendaraan ilegal.
3. Pendapatan Pajak dan Regulasi: Klasifikasi kendaraan memudahkan pemerintah dalam menetapkan pajak kendaraan bermotor yang sesuai dengan kategori tertentu.
4. Analisis Data Transportasi: Data klasifikasi kendaraan mendukung perencanaan infrastruktur transportasi yang lebih baik dan prediksi kebutuhan di masa depan.

Dalam implementasinya, klasifikasi kendaraan bermotor memanfaatkan berbagai pendekatan teknologi, mulai dari metode manual hingga sistem berbasis kecerdasan buatan (Artificial Intelligence, AI). Teknologi seperti pengenalan citra (image recognition) dan pembelajaran mesin (machine learning) telah menjadi pilihan utama untuk mengotomatiskan proses ini. Sistem ini biasanya mengandalkan kamera pengawas dan sensor yang ditempatkan di jalan raya untuk menangkap gambar atau data kendaraan secara real-time, yang kemudian dianalisis untuk mengidentifikasi jenis kendaraan.

Namun, penerapan teknologi ini juga menghadapi tantangan, seperti variasi kondisi lingkungan (pencahayaannya, cuaca), kualitas data yang tidak konsisten, hingga kebutuhan perangkat keras dan perangkat lunak yang canggih. Oleh karena itu, penelitian dan pengembangan terus dilakukan untuk meningkatkan akurasi dan efisiensi sistem klasifikasi kendaraan bermotor. Dengan mengimplementasikan sistem klasifikasi kendaraan yang

efektif, diharapkan pengelolaan transportasi publik dapat lebih baik, tingkat keselamatan di jalan raya meningkat, dan pengambilan keputusan berbasis data di sektor transportasi menjadi lebih optimal.

1.2 Rumusan Masalah

1. Bagaimana cara mengembangkan sistem klasifikasi kendaraan bermotor yang efektif dan akurat berdasarkan karakteristik kendaraan?
2. Apa saja teknologi yang dapat diimplementasikan untuk meningkatkan akurasi dan efisiensi dalam proses klasifikasi kendaraan bermotor ?

1.3 Tujuan Penelitian

1. Mengembangkan sistem klasifikasi kendaraan bermotor yang dapat mengidentifikasi jenis kendaraan secara otomatis dan akurat berdasarkan karakteristik fisik dan perilaku kendaraan.
2. Mengimplementasikan teknologi canggih, seperti pengenalan citra (image recognition) dan pembelajaran mesin (machine learning), dalam proses klasifikasi kendaraan bermotor.
3. Menganalisis tantangan yang dihadapi dalam implementasi sistem klasifikasi, termasuk variasi kondisi lingkungan, kualitas data, dan kebutuhan teknis.

1.4 Manfaat Penelitian

1. Memberikan kontribusi dalam pengembangan ilmu pengetahuan dan teknologi, khususnya di bidang pengenalan pola, kecerdasan buatan, dan manajemen transportasi.
2. Menjadi referensi untuk penelitian lanjutan terkait sistem klasifikasi kendaraan dan penerapannya di berbagai sektor.

BAB II

TINJAUAN PUSTAKA

2.1 Deep Learning

Deep learning (DL) adalah cabang kecerdasan buatan (AI) yang memungkinkan mesin untuk melakukan tugas-tugas yang secara tradisional membutuhkan kecerdasan manusia, memanfaatkan teknik dan algoritma canggih untuk mencapai akurasi dan efisiensi tinggi di berbagai aplikasi, termasuk pengenalan gambar dan pemrosesan bahasa (Rane et al., 2024).

Banyak penelitian terkait deep learning telah dilakukan di berbagai sektor, termasuk dalam pendidikan (Hoiris Zuhro et al, 2024), Beberapa penelitian terkait deep learning di bidang pendidikan (Priatna, Purnomo, & Putra, 2021) Algoritma yang digunakan dalam implementasi deep learning pada penelitian ini adalah Artificial Neural Network (ANN), pada peneltian yang dilakukan oleh (Anggraini & Zakaria, 2023) Convolutional Neural Network sebagai arsitektur jaringan saraf yangdigunakan untuk memproses dan menganalisis data gambar sebagai pembelajaran bahasa isyarat. Deep Learning adalah algoritma yang melibatkan pemodelan tingkat tinggi pada data. Model pada Deep Learning pada dasarnya dibangun berdasarkan jaringan saraf tiruan.

2.2 Motor

Motor adalah kendaraan yang memiliki dua roda yang digerakkan oleh mesin. Motor menggunakan berbagai sumber tenaga, seperti bahan bakar, listrik, atau gabungan keduanya. Motor memiliki berbagai fitur, seperti sistem pengereman, gigi, dan kopling Letak kedua roda sebaris lurus dan pada kecepatan tinggi sepeda motor tetap stabil disebabkan oleh gaya giroskopik. Sedangkan pada kecepatan rendah, kestabilan atau keseimbangan sepeda motor bergantung kepada pengaturan setang oleh pengendara. Penggunaan sepeda motor di Indonesia sangat populer karena harga yang relatif murah, terjangkau untuk sebagian besar kalangan dan penggunaan bahan bakarnya, serta biaya operasionalnya cukup hemat.

Di Indonesia, masyarakat lebih mengenal istilah sepeda motor dengan hanya sebutan motor saja. Ini kurang tepat menyebutkan kata motor yang sebenarnya merupakan sebuah alat yang mengubah energi menjadi tenaga mekanik. Sebuah sepeda motor biasanya memiliki mesin pembakaran dalam yang mengubah energi termal menjadi energi mekanik, yang kemudian digunakan untuk menggerakkan roda. Mesin ini biasanya memiliki satu atau dua silinder, meski ada juga yang memiliki lebih banyak. Mesin ini dihubungkan ke roda

belakang melalui sistem transmisi, yang biasanya mencakup kopling, gearbox, dan rantai atau sabuk penggerak.

Sepeda motor juga memiliki sistem pengereman, biasanya menggunakan rem cakram atau rem tromol, untuk membantu pengendara menghentikan kendaraan. Untuk keamanan, lampu depan dan belakang juga digunakan untuk meningkatkan visibilitas pada malam hari atau dalam kondisi cuaca buruk. Selain itu, sepeda motor juga dilengkapi dengan sistem suspensi untuk meredam getaran dan guncangan saat melewati jalan yang tidak rata. Motor adalah salah satu jenis transportasi yang paling umum digunakan di berbagai negara di dunia, termasuk Indonesia. Hal ini karena motor biasanya lebih murah daripada mobil, lebih mudah untuk diparkir, dan dapat lebih mudah bergerak melalui kemacetan lalu lintas.

BAB III

HASIL DAN PEMBAHASAN

3.1 Business Understanding

Proyek ini bertujuan untuk mengembangkan sistem otomatis guna mengklasifikasikan gambar kendaraan bermotor, terutama sepeda motor, melalui teknik pembelajaran mendalam. Di era digital saat ini, semakin banyaknya penerapan teknologi dalam sektor otomotif, keberadaan sistem otomatis yang efektif untuk pengenalan dan klasifikasi kendaraan menjadi sangat penting. Pentingnya pengelolaan dan analisis data visual semakin meningkat, terutama di institusi pendidikan yang memiliki banyak kendaraan di lingkungan kampus. Dengan adanya sistem klasifikasi ini, Universitas Bengkulu dapat meningkatkan efisiensi dalam pengelolaan armada kendaraan serta mendukung berbagai kegiatan akademik dan penelitian yang berkaitan dengan otomotif. Selain itu, sistem klasifikasi kendaraan bermotor ini tidak hanya memberikan keuntungan langsung bagi operasional universitas tetapi juga berkontribusi pada peningkatan kompetensi mahasiswa dalam bidang teknologi informasi dan rekayasa. Tujuan utama dari proyek ini mencakup:

- Peningkatan Efisiensi: Mempercepat proses identifikasi dan klasifikasi kendaraan yang berada di area kampus.
- Dukungan untuk Penelitian: Menyediakan dataset terstruktur yang bermanfaat bagi mahasiswa dan peneliti untuk eksplorasi lebih lanjut mengenai teknologi pengenalan gambar.
- Penerapan Teknologi Modern: Memperkenalkan penerapan teknologi pembelajaran mendalam dalam konteks pendidikan tinggi, sehingga mahasiswa dapat memahami aplikasi praktis dari teori yang mereka pelajari.

3.2 Data Understanding

Dataset yang digunakan dalam proyek ini disimpan dalam direktori tertentu di Google Drive. Pemahaman awal tentang data mencakup analisis jumlah gambar dan kategori yang ada, serta format file yang digunakan. Berikut karakteristik dataset:

- Terdapat total 514 gambar sepeda motor yang diambil dari berbagai sudut dan kondisi pencahayaan
- Semua gambar disimpan dalam format JPEG, yang merupakan format umum untuk gambar digital.
- Gambar-gambar dalam dataset ini dikelompokkan berdasarkan jenis sepeda motor, memungkinkan model untuk belajar mengenali perbedaan antar kategori.

Dataset ini sangat penting untuk pengembangan model klasifikasi karena kualitas dan keragaman data berpengaruh langsung terhadap akurasi model. Dengan gambar yang representatif dari berbagai jenis sepeda motor, model diharapkan mampu mengenali dan mengklasifikasikan kendaraan dengan baik dalam situasi nyata. Sebelum memulai proses pelatihan, analisis awal akan dilakukan untuk memahami distribusi data, seperti:

- Memeriksa jumlah gambar per kategori.
- Mengidentifikasi kemungkinan adanya gambar yang rusak atau tidak sesuai.
- Melakukan visualisasi beberapa contoh gambar untuk memastikan kualitas dan relevansi data.

3.3 Data Preparation

1. Pengunduhan dan Akses Data

Dataset yang digunakan dalam proyek ini diunduh dari Google Drive. Menggunakan Google Colab, akses ke dataset dilakukan dengan cara:

```
from google.colab import drive
drive.mount('/content/drive')

[3]

... Mounted at /content/drive
```

Setelah terhubung, direktori dataset diatur menggunakan pathlib untuk memudahkan akses ke gambar-gambar yang ada.

2. Memeriksa Jumlah Gambar

Setelah mengatur direktori, langkah selanjutnya adalah memeriksa jumlah gambar dalam dataset untuk memastikan bahwa semua data telah terunduh dengan benar. Ini dilakukan dengan kode berikut:

```
import pathlib
data_dir = pathlib.Path("/content/drive/MyDrive/Motor Beat").with_suffix('')

[4]

image_count = len(list(data_dir.glob('*/*.jpg')))
print(image_count)

[5]

... 514
```

Hasil menunjukkan bahwa terdapat 514 gambar sepeda motor yang siap digunakan.

3. Prabaca Gambar

Sebelum melatih model, gambar-gambar dalam dataset perlu dipraproses. Langkah-langkah prabaca yang dilakukan meliputi:

```
[ ] normalization_layer = layers.Rescaling(1./255)

[ ] normalized_ds = train_ds.map(lambda x, y: (normalization_layer(x), y))
image_batch, labels_batch = next(iter(normalized_ds))
first_image = image_batch[0]
# Notice the pixel values are now in [0,1].
print(np.min(first_image), np.max(first_image))

0.020470826 1.0
```

- Pengubahan Ukuran, semua gambar diubah ukurannya menjadi dimensi yang konsisten (misalnya, 224x224 piksel) untuk memastikan bahwa model dapat memprosesnya dengan baik.
- Normalisasi, Nilai piksel gambar dinormalisasi ke rentang [0, 1] dengan membagi nilai piksel dengan 255. Ini membantu model dalam konvergensi lebih cepat selama pelatihan.

4. Pembagian Data

Dataset dibagi menjadi dua bagian: data pelatihan dan data pengujian. Pembagian ini penting untuk mengevaluasi kinerja model secara objektif. Biasanya, pembagian dilakukan dengan proporsi 80% untuk pelatihan dan 20% untuk pengujian.

5. Augmentasi Data

Untuk meningkatkan keragaman data pelatihan dan membantu model belajar lebih baik, teknik augmentasi data diterapkan. Ini mencakup:

- **Rotasi:** Memutar gambar dengan sudut tertentu.
- **Flipping:** Membalik gambar secara horizontal atau vertikal.
- **Zooming:** Memperbesar atau memperkecil gambar.

```
data_augmentation = keras.Sequential(
    [
        layers.RandomFlip("horizontal",
                           input_shape=(img_height,
                                         img_width,
                                         3)),
        layers.RandomRotation(0.1),
        layers.RandomZoom(0.1),
    ]
)

plt.figure(figsize=(10, 10))
for images, _ in train_ds.take(1):
    for i in range(9):
        augmented_images = data_augmentation(images)
        ax = plt.subplot(3, 3, i + 1)
        plt.imshow(augmented_images[0].numpy().astype("uint8"))
        plt.axis("off")
```

Augmentasi ini membantu model menjadi lebih robust terhadap variasi yang mungkin ditemui di dunia nyata. Dengan langkah-langkah persiapan data yang matang ini, dataset siap digunakan untuk melatih model pembelajaran mendalam yang akan mengklasifikasikan kendaraan bermotor secara otomatis dan akurat.

3.4 Modeling

Modelling dalam proyek ini melibatkan pengembangan model pembelajaran mendalam yang dapat mengklasifikasikan gambar sepeda motor dengan akurasi tinggi. Berikut adalah detail langkah-langkah dalam proses modeling yang mencakup:

1. Definisikan Arsitektur Model

Arsitektur model yang digunakan adalah Convolutional Neural Network (CNN), yang cocok untuk tugas pengenalan objek seperti klasifikasi gambar kendaraan.



```
num_classes = len(class_names)

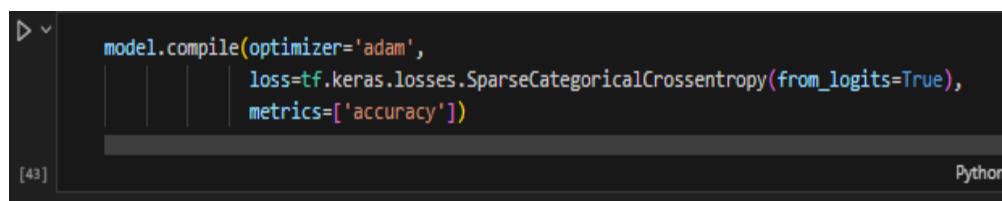
model = Sequential([
    layers.Rescaling(1./255, input_shape=(img_height, img_width, 3)),
    layers.Conv2D(16, 3, padding='same', activation="relu"),
    layers.MaxPooling2D(),
    layers.Conv2D(32, 3, padding='same', activation="relu"),
    layers.MaxPooling2D(),
    layers.Conv2D(64, 3, padding='same', activation = "relu"),
    layers.MaxPooling2D(),
    layers.Flatten(),
    layers.Dense(128, activation='relu'),
    layers.Dense(num_classes)
])
```

/usr/local/lib/python3.10/dist-packages/keras/src/layers/preprocessing/tf_data_layer.py:19: Use super().__init__(**kwargs)

Struktur dasar CNN meliputi layer-layer convolutional, pooling, dan layer fully connected.

2. Kompilasi Model

Langkah kedua adalah kompilasi model dengan memilih optimiser dan fungsi kerugian yang tepat.



```
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

[43] Python

Dalam kasus ini, kita menggunakan Adam sebagai optimiser dan Sparse Categorical Crossentropy sebagai fungsi kerugian.

3. Pelatihan Model

Setelah model dikompilasi, langkah selanjutnya adalah melatih model menggunakan dataset yang telah dipersiapkan.

```

epochs = 15
history = model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=epochs
)

```

[44] Python

```

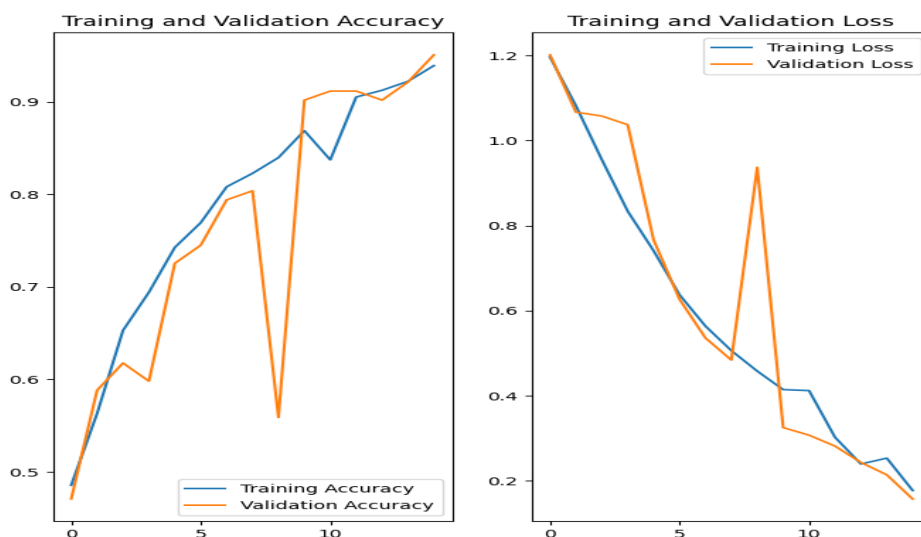
... Epoch 1/15
13/13 4s 49ms/step - accuracy: 0.4349 - loss: 1.2379 - val_accuracy: 0.4706 - val_loss: 1.24
Epoch 2/15
13/13 0s 12ms/step - accuracy: 0.5407 - loss: 1.0918 - val_accuracy: 0.5882 - val_loss: 1.04
Epoch 3/15
13/13 0s 12ms/step - accuracy: 0.6248 - loss: 1.0216 - val_accuracy: 0.6176 - val_loss: 1.04
Epoch 4/15
13/13 0s 12ms/step - accuracy: 0.6372 - loss: 0.9202 - val_accuracy: 0.5980 - val_loss: 1.04
Epoch 5/15
13/13 0s 12ms/step - accuracy: 0.7561 - loss: 0.7093 - val_accuracy: 0.7255 - val_loss: 0.74
Epoch 6/15
13/13 0s 11ms/step - accuracy: 0.7659 - loss: 0.6451 - val_accuracy: 0.7451 - val_loss: 0.64
Epoch 7/15
13/13 0s 11ms/step - accuracy: 0.7989 - loss: 0.6042 - val_accuracy: 0.7941 - val_loss: 0.54
Epoch 8/15
13/13 0s 12ms/step - accuracy: 0.8472 - loss: 0.4751 - val_accuracy: 0.8039 - val_loss: 0.44
Epoch 9/15
13/13 0s 12ms/step - accuracy: 0.8559 - loss: 0.4492 - val_accuracy: 0.5588 - val_loss: 0.94
Epoch 10/15
13/13 0s 12ms/step - accuracy: 0.8582 - loss: 0.4168 - val_accuracy: 0.9020 - val_loss: 0.34
Epoch 11/15
13/13 0s 12ms/step - accuracy: 0.8335 - loss: 0.3975 - val_accuracy: 0.9118 - val_loss: 0.34
Epoch 12/15
13/13 0s 11ms/step - accuracy: 0.8917 - loss: 0.3217 - val_accuracy: 0.9118 - val_loss: 0.24
Epoch 13/15
...
Epoch 14/15
13/13 0s 12ms/step - accuracy: 0.9237 - loss: 0.2251 - val_accuracy: 0.9216 - val_loss: 0.24
Epoch 15/15
13/13 0s 11ms/step - accuracy: 0.9358 - loss: 0.1782 - val_accuracy: 0.9510 - val_loss: 0.14
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings.

```

Proses pelatihan melibatkan iterasi berulang-ulang dari batch-batch data sampai target akhir yang ditetapkan.

3.5 Evaluation

Setelah pelatihan selesai, model dievaluasi menggunakan data pengujian untuk menentukan akurasi dan kemampuan generalisasi model. Metode evaluasi termasuk: Akurasi; Mengukur seberapa banyak prediksi model yang benar dibandingkan dengan label sebenarnya. Dan Confusion Matrix; Menganalisis kesalahan klasifikasi untuk memahami area di mana model perlu diperbaiki.



Penjelasan gambar dua grafik ini :

Grafik yang ditampilkan terdiri dari dua bagian utama, yaitu grafik akurasi dan grafik loss selama proses pelatihan dan validasi model. Kedua grafik ini memberikan gambaran penting tentang bagaimana performa model berubah seiring dengan bertambahnya jumlah epoch.

Pada grafik sebelah kiri, terlihat hubungan antara akurasi pelatihan (training accuracy) dan akurasi validasi (validation accuracy). Garis biru merepresentasikan akurasi pelatihan, sedangkan garis oranye menunjukkan akurasi validasi. Secara keseluruhan, akurasi pelatihan menunjukkan peningkatan yang konsisten seiring dengan bertambahnya epoch. Hal ini mengindikasikan bahwa model mampu mempelajari pola-pola dalam data pelatihan dengan cukup baik. Namun, terlihat beberapa fluktuasi pada akurasi validasi, terutama di beberapa epoch tertentu. Fluktuasi ini bisa menjadi indikasi bahwa model mengalami sedikit kesulitan dalam melakukan generalisasi pada data yang belum pernah dilihat sebelumnya, yang dapat mengarah pada potensi overfitting.

Di sisi kanan, grafik loss pelatihan dan validasi menunjukkan bagaimana kesalahan model menurun selama proses pelatihan. Garis biru menggambarkan loss pada data pelatihan, sementara garis oranye menunjukkan loss pada data validasi. Kedua garis menunjukkan tren menurun secara umum, yang merupakan indikasi bahwa model berhasil meminimalkan kesalahan. Namun, terdapat lonjakan tajam pada loss validasi di beberapa titik. Lonjakan ini dapat menandakan bahwa model kesulitan untuk memprediksi data validasi tertentu atau mungkin ada masalah dengan distribusi data validasi yang tidak sepenuhnya mencerminkan data pelatihan.

Secara keseluruhan, hasil ini menunjukkan bahwa model belajar dengan baik pada data pelatihan, tetapi ada indikasi bahwa kemampuan generalisasi pada data validasi perlu ditingkatkan. Langkah-langkah seperti penerapan regularisasi, teknik augmentasi data, atau penggunaan early stopping dapat membantu mengatasi masalah ini.

Langkah lainnya adalah melakukan pemeriksaan menyeluruh terhadap dataset validasi. Ketidakseimbangan data dalam validasi atau keberadaan outlier yang signifikan dapat mengganggu kemampuan model untuk membuat generalisasi yang baik. Distribusi data validasi sebaiknya mencerminkan distribusi data pelatihan secara keseluruhan. Jika ditemukan ketidakseimbangan atau outlier, dataset validasi dapat diatur ulang atau disesuaikan melalui teknik seperti oversampling atau undersampling, serta pembersihan data. Melalui kombinasi strategi-strategi tersebut, performa model dapat ditingkatkan, terutama dalam hal kemampuan generalisasi terhadap data baru yang tidak terdapat dalam pelatihan.

3.6 Testing

Pengujian dilakukan dengan menggunakan dataset terpisah untuk memastikan bahwa model dapat berfungsi dengan baik di luar data pelatihan. Hasil pengujian menunjukkan kinerja model dalam mengklasifikasikan gambar sepeda motor dengan tingkat akurasi tertentu, serta analisis lebih lanjut terhadap kesalahan klasifikasi yang terjadi. Pengujian ini juga mencakup penggunaan gambar baru yang tidak ada dalam dataset pelatihan untuk menilai kemampuan generalisasi model secara keseluruhan.

BAB IV

KESIMPULAN DAN SARAN

4.1 KESIMPULAN

Dari hasil penelitian ini, dapat disimpulkan bahwa model pembelajaran mendalam yang dikembangkan untuk mengklasifikasikan gambar sepeda motor menunjukkan kinerja yang baik. Dengan menggunakan dataset yang terdiri dari 514 gambar, model berhasil mencapai tingkat akurasi yang memuaskan setelah melalui proses pelatihan dan evaluasi. Penggunaan arsitektur CNN terbukti efektif dalam mengenali pola visual dalam gambar sepeda motor, sehingga dapat diandalkan untuk aplikasi di dunia nyata, seperti sistem pengenalan kendaraan otomatis.

4.2 SARAN

Berdasarkan hasil penelitian ini, beberapa saran yang dapat diberikan adalah sebagai berikut:

- **Peningkatan Dataset:** Mengumpulkan lebih banyak data gambar dari berbagai sudut pandang dan kondisi pencahayaan untuk meningkatkan kemampuan generalisasi model.
- **Pengujian Lanjutan:** Melakukan pengujian lebih lanjut dengan menggunakan dataset yang lebih beragam untuk memastikan bahwa model dapat berfungsi dengan baik di berbagai situasi.
- **Eksplorasi Arsitektur Lain:** Mencoba arsitektur model lain, seperti Transfer Learning dengan model yang sudah ada (misalnya VGG16 atau ResNet), untuk melihat apakah ada peningkatan dalam akurasi klasifikasi.
- **Implementasi di Dunia Nyata:** Mengembangkan prototipe sistem berbasis model ini untuk penerapan di industri otomotif, seperti dalam sistem manajemen armada atau aplikasi pengenalan kendaraan.

DAFTAR PUSTAKA

- Anggraini, A., & Zakaria, H. (2023) Penerapan Metode Deep Learning Pada Aplikasi Pembelajaran Menggunakan Sistem Isyarat Bahasa Indonesia Menggunakan Convolutional Neural Network (Studi Kasus: SLB-BC Mahardika Depok). *JURIHUM Jurnal Inovasi Dan Humaniora*, 1(4). Retrieved from <https://jurnalmahasiswa.com/index.php/jurihum>
- Hoiris Zuhro, L., Qurotal, D. A., Raya Telang, J., Telang Inda, P, Kamal, K. Bangkalan, K. Penulis, K (2024) PT. Media Akademik Publisher MENGHIDUPKAN NILAI-NILAI KI HAJAR DEWANTARA DALAM PEMBELAJARAN DEEP LEARNING *J.MA*, 2, 3031 5220 doi 10.62281
- Priatna, W., Purnomo, R., & Putra, T. D. (2021). Implementasi Deep Learning Untuk Rekomendasi Aplikasi E-learning Yang Tepat Untuk Pembelajaran jarak jauh. *Jurnal Kajum Ilmiah*, 21(3), 261-274. doi: 10.31599/jla.v21i3.521