

Classification Vehicle Tire Quality using Convolutional Neural Networks

Vila Rusantia Pratiwi^{1*}, Nova Rijati^{2**}

*Teknik Informatika, Universitas Dian Nuswantoro Semarang

**Teknik Informatika, Universitas Dian Nuswantoro Semarang

111202012587@mhs.dinus.ac.id¹, nova.rijati@dsn.dinus.ac.id²

Article Info

Article history:

Received 2024-07-01

Revised 2024-07-10

Accepted 2024-07-12

Keyword:

Classification,
Vehicle Tires,
Convolutional Neural Network.

ABSTRACT

Tires are a very important component in a vehicle because they are related to driving safety. Defective tires often cause accidents ranging from minor to fatal accidents. Convolutional Neural Network (CNN) is a type of neural network that is used to detect and recognize objects in an image. CNN can imitate the image recognition system in the human visual cortex, making it suitable for identification and classification of image data. This research aims to develop and evaluate a CNN model that is able to classify vehicle tires as 'defective' or 'good'. Model uses a total of 1856 tire images from kaggle.com and is labeled 'defective' or 'good'. Dataset is split using four different data split ratios (60:40, 70:30, 80:20, and 90:10) to determine the optimal distribution that improves the generalization ability of the model. Model evaluation uses accuracy, precision and recall matrices, which are calculated based on the confusion matrix results from testing on 300 data samples. Research results show that the model achieves the best performance at a split ratio of 80:20, with an accuracy of 76.67%, precision of 77.33%, and recall of 76.32%.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Penelitian ini mengkaji pentingnya ban sebagai komponen utama pada kendaraan, yang tidak hanya mengurangi dampak getaran dari jalan tidak rata dan melindungi roda dari keausan, tetapi juga meningkatkan stabilitas dan kenyamanan saat berkendara [1]. Karena sering digunakan, ban bisa mengalami kerusakan seperti retak. Retakan ini bisa muncul karena beberapa faktor, seperti usia ban atau kondisi jalan yang buruk [2]. Namun, kerusakan ban seperti retak sering terjadi karena faktor-faktor seperti usia ban dan kondisi jalan yang buruk. Meskipun demikian, pemeriksaan kualitas ban masih umum dilakukan secara manual dalam kehidupan sehari-hari untuk membedakan antara ban yang baik dan buruk [3].

Convolutional Neural Network (CNN) merupakan jenis khusus dari Jaringan Saraf Tiruan (JST) atau *Neural Network*. Pada dasarnya, *Neural Network* adalah sekumpulan node komputasi yang saling terhubung, yang disebut neuron, dan konsep ini terinspirasi dari cara sistem saraf biologi berfungsi, seperti yang terjadi dalam otak manusia [4]. Pengembangan model klasifikasi citra ban menggunakan CNN menjadi fokus utama dalam penelitian ini. CNN dikenal efektif dalam mengidentifikasi cacat pada ban dengan memproses data

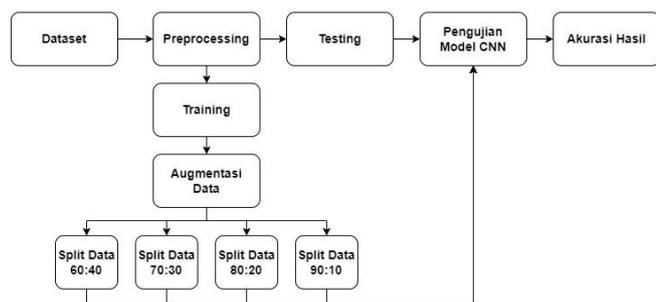
piksel secara otomatis, mengurangi ketergantungan pada keterampilan manusia [5]. Metode ini menggunakan operasi matematika Convolutional untuk mengekstraksi fitur penting dari citra, yang efisien dan efektif dalam pengolahan citra kendaraan dari mobil pribadi hingga truk komersial [6].

Penelitian [7], berhasil mengidentifikasi cacat pada produk menggunakan berbagai teknik, termasuk machine learning dan deep learning. Penelitian lalu [8], menerapkan metode CNN dengan kombinasi *Long-Short-Term Memory (LSTM)* yang menghasilkan tingkat akurasi sebesar 99%, sedangkan klasifikasi kecacatan ban untuk mengendalikan kualitas produk menggunakan Model CNN dengan Arsitektur VGG-16 dilakukan oleh [9]. Dan [10] menggunakan teknik *Classification and Regression Tree (CART)* untuk mencari akar masalah dari radial run out ban. Hasil penelitian menunjukkan bahwa faktor-faktor seperti lot komponen *Tread*, *Bead*, dan *Sidewall* mempengaruhi nilai radial run out. Penelitian [11] melakukan klasifikasi retak ban kendaraan menggunakan Arsitektur ResNet50. Dan penelitian [12] melakukan deteksi kerusakan dan kelayakan ban mobil berdasarkan *ekstraksitektur* menggunakan *gray level cooccurrence matrix* dengan metode jaringan syaraf tiruan, terhadap 100 data citra menghasilkan akurasi sebesar 80%.

Sedangkan [13], melakukan deteksi kerusakan ban mobil menggunakan CNN dengan Arsitektur ResNet-34, model ini dilatih dengan parameter seperti *optimizer Adam*, *learning rate* 0.0001, *batch size* 32, dan *epoch* 50. Model CNN ini mampu memprediksi baik kondisi ban normal maupun rusak dengan akurat, memiliki akurasi sebesar 0.916, presisi 0.907, recall 0.927, dan F1-score 0.917. Penelitian [14] mampu melakukan deteksi kerusakan ban pada angkutan umum dengan metode *local binary pattern* dan klasifikasi nearest neighbor berhasil mendeteksi kerusakan pada permukaan ban angkutan umum dengan akurasi tertinggi 91,82%. Tujuan penelitian ini adalah mengembangkan model yang dapat meningkatkan keselamatan dan efisiensi transportasi dengan mengklasifikasikan kualitas ban secara akurat menggunakan CNN. Hasil dari penelitian ini diharapkan dapat memberikan wawasan mendalam tentang potensi metode CNN dalam menentukan kualitas ban, serta menunjukkan tingkat akurasi optimal dalam evaluasi model yang telah dikembangkan [15].

II. METODE

Metode yang diusulkan dalam penelitian ini dapat di gambarkan seperti pada Gambar 1.



Gambar 1. Metode yang diusulkan

A. Dataset

Dataset dalam penelitian ini adalah 1856 citra ban, yang diambil melalui *website Kaggle.com*. Data citra ban terdiri atas 2 kelas yaitu, *defective* (cacat) dan *good* (bagus). Kelas 'defective' terdiri dari 1028 gambar dan menggambarkan ban-ban yang memiliki satu atau lebih cacat yang bisa berdampak negatif terhadap fungsionalitas dan keamanan. Cacat tersebut dapat berupa keretakan, pemakaian yang tidak merata, paku atau benda tajam yang tertanam, pengelupasan, atau kerusakan pada dinding samping.



Gambar 2. Contoh Ban Cacat (Defective)

Sedangkan kelas 'good' mencakup 828 gambar ban yang memenuhi atau melampaui kriteria kinerja dan keamanan yang ditetapkan. Kelas ini menggambarkan ban yang dalam kondisi prima, menunjukkan pola tapak yang utuh, kedalaman tapak yang memadai, tidak ada kerusakan terlihat, dan ketiadaan deformasi atau cacat. Ban-ban ini ditandai sebagai layak pakai dan diharapkan akan beroperasi sesuai dengan spesifikasi pabrikan dalam kondisi pengendalian standar.



Gambar 3. Contoh Ban Bagus (Good)

B. Preprocessing

Proses ini melibatkan transformasi data mentah untuk memenuhi spesifikasi input yang diharapkan oleh algoritma pembelajaran mesin, termasuk pembagian dataset menjadi dataset pelatihan dan pengujian, normalisasi, dan augmentasi data. Distribusi yang tepat antara data pelatihan, validasi, dan pengujian memungkinkan peneliti untuk mengevaluasi model dengan lebih akurat, memastikan bahwa model yang dikembangkan memiliki kemampuan generalisasi yang kuat, dan tidak terlalu menyesuaikan diri hanya dengan contoh data yang diberikan selama pelatihan. Dalam penelitian ini, dataset terdiri dari 1856 gambar citra ban kendaraan yang telah diberi label menjadi dua kategori: 'defective' dan 'good condition'. Pembagian dataset dilakukan dengan strategi yang memastikan validasi yang efektif dan keandalan model dalam kinerja prediksinya.

TABEL I
DISTRIBUSI DATASET

Split Ratio	Data Train	Data Validasi	Data Testing	Total Data
60:40	934	622	300	1856
70:30	1090	466	300	1856
80:20	1246	310	300	1856
90:10	1402	154	300	1856

Pembagian dataset ini diatur sedemikian rupa agar setiap split ratio memberikan jumlah data yang cukup untuk pelatihan, validasi, dan pengujian, memastikan bahwa model mendapatkan variasi yang cukup dalam data latihannya, serta memiliki data validasi yang memadai untuk *tuning hyperparameter*, dan data pengujian yang cukup untuk evaluasi akhir.

C. Pengujian Model CNN

Pelatihan yang efektif menghasilkan model yang dapat mengenali dan mengklasifikasikan dengan akurat kualitas ban kendaraan sebagai 'defective' atau 'good'. Model CNN akan dilatih menggunakan 4 skenario split ratio dataset yang berbeda-beda yang ada pada Tabel 1. Proses pelatihan dalam penelitian ini dirancang untuk memaksimalkan kapasitas pembelajaran model sambil meminimalisir risiko overfitting. Langkah-langkah dalam proses ini diatur untuk menyesuaikan model dengan karakteristik dataset pelatihan, sambil mempertahankan kemampuan model untuk menggeneralisasi ke data baru. Pseudocode untuk arsitektur pemodelan CNN.

```
BEGIN
// Inisialisasi model sequential
SET model = Sequential()

// Tambahkan lapisan konvolusi
model.add(Conv2D(256, (3, 3), activation='relu', input_shape=(150, 150, 3)))
model.add(MaxPooling2D(2, 2))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D(2, 2))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(2, 2))
model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(MaxPooling2D(2, 2))

// Ratakan output untuk lapisan terhubung penuh
model.add(Flatten())

// Tambahkan lapisan dropout
model.add(Dropout(0.2))

// Tambahkan lapisan terhubung penuh
model.add(Dense(512, activation='relu'))
model.add(Dense(2, activation='softmax'))

// Kompilasi model
model.compile(loss='categorical_crossentropy',
optimizer=Adam(learning_rate=0.00003), metrics=['accuracy'])
END
```

Performa model CNN yang dikembangkan untuk klasifikasi kualitas ban kendaraan telah dipantau dan dievaluasi secara menyeluruh selama fase pelatihan dan validasi. Grafik akurasi dan loss pelatihan dan validasi memberikan pengetahuan yang penting tentang proses pembelajaran model.

D. Evaluasi Model CNN

Evaluasi model merupakan tahap penting dalam proses penelitian untuk menentukan seberapa baik model CNN telah mempelajari tugas klasifikasi dan seberapa efektif model tersebut dalam menggeneralisasikan pembelajarannya pada data yang tidak pernah dilihat sebelumnya. Pada penelitian ini, evaluasi akhir dilakukan menggunakan sebanyak 300 data yang ada pada set pelatihan yang diuji menggunakan model yang sudah ditraining dengan 4 skenario berbeda sebelumnya.

Confusion matrix adalah salah satu alat evaluasi yang paling informatif dan banyak digunakan dalam klasifikasi.

Dengan memetakan prediksi model terhadap label sebenarnya, *confusion matrix* memberikan visualisasi yang jelas tentang di mana model berhasil dan di mana ia membuat kesalahan. Pseudocode untuk split data.

```
BEGIN
// Hitung Confusion Matrix
SET cm = confusion_matrix(true_classes, predicted_classes)
PRINT("Confusion Matrix:\n", cm)

// Plot Confusion Matrix
plt.figure(figsize=(5, 4))
sns.heatmap(cm, xticklabels=class_labels, yticklabels=class_labels,
annot=True, fmt='g', cmap='Blues', annot_kws={"size": 14})
plt.xlabel('Predicted')
plt.ylabel('True')
plt.title('Confusion Matrix')
plt.show()
END
```

Dalam penilaian kinerja model klasifikasi, penggunaan metrik yang tepat adalah esensial untuk mendapatkan pemahaman tentang efektivitas model. Model CNN dalam penelitian ini dievaluasi menggunakan metrik evaluasi akurasi, presisi, dan *recall*.

III. HASIL DAN PEMBAHASAN

A. Pembagian Dataset untuk Pengujian

Dalam penelitian ini, dataset pengujian terdiri dari sampel acak yang diambil dari dataset utama, menggunakan pseudocode pada Gambar 6. Sebanyak 150 gambar untuk masing-masing kelas 'defective' dan 'good' dengan total 300 data testing dipilih secara acak untuk membentuk set pengujian. Pseudocode untuk split data.

```
BEGIN
// Import library yang diperlukan
IMPORT ImageDataGenerator
// Inisialisasi objek ImageDataGenerator
SET datagen = ImageDataGenerator(
rescale=1./255,
validation_split=0.2,
rotation_range=45,
zoom_range=0.2,
horizontal_flip=True,
vertical_flip=True,
brightness_range=[0.5, 1.5],
fill_mode='nearest'
)
// Penggunaan datagen:
// - rescale: Normalisasi nilai piksel
// - validation_split: Pembagian data menjadi train dan validation set
// - rotation_range: Rotasi gambar hingga 45 derajat
// - zoom_range: Zoom in/out gambar
// - horizontal_flip: pencerminan horizontal
// - vertical_flip: pencerminan vertikal
// - brightness_range: Perubahan kejernihan
// - fill_mode: Strategi pengisian piksel baru setelah augmentasi
END
```

Pembagian dataset yang adil dan representatif ini mendukung integritas penilaian model. Dengan cara ini, set pengujian yang terpisah dan tidak bias memberikan

kesempatan untuk mengevaluasi model CNN secara objektif dalam mengukur kemampuannya dalam klasifikasi kualitas ban kendaraan.

B. Augmentasi Data

Augmentasi data adalah langkah penting dalam pra-pelatihan data, terutama di bidang computer vision dan deep learning. Tujuannya adalah untuk menciptakan variasi sintesis dalam dataset pelatihan, meningkatkan generalisasi model, mengurangi risiko *overfitting*, dan memperkaya dataset tanpa mengumpulkan data tambahan. Pseudocode untuk Augmentasi Data.

```
BEGIN
// Inisialisasi objek ImageDataGenerator
SET datagen = ImageDataGenerator(
    rescale=1./255,
    validation_split=0.2,
    rotation_range=45,
    zoom_range=0.2,
    horizontal_flip=True,
    vertical_flip=True,
    brightness_range=[0.5, 1.5],
    fill_mode='nearest'
)
END
```

C. Dataset Pelatihan dan Validasi

Pembagian dataset menjadi subset pelatihan dan validasi merupakan langkah penting dalam siklus pengembangan model pembelajaran mesin. Dalam konteks penelitian ini, dataset pelatihan akan digunakan untuk memfasilitasi proses pembelajaran model CNN, sedangkan dataset validasi akan digunakan untuk memonitor dan mengevaluasi generalisasi model serta mencegah *overfitting* selama proses pelatihan. Pseudocode Untuk Membuat Set Pelatihan dan Validasi.

```
BEGIN
// Initialize training and validation data generators
train_generator = datagen.flow_from_directory(
    base_dir,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical',
    #color_mode='grayscale',
    subset='training' // set as training data
)

validation_generator = datagen.flow_from_directory(
    base_dir,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical',
    #color_mode='grayscale',
    subset='validation'
)
END
```

Untuk memastikan bahwa model memiliki kemampuan generalisasi yang tinggi, dataset dibagi menggunakan skema validasi silang, di mana data dibagi menjadi dataset pelatihan dan validasi dengan perbandingan 80:20. (dilakukan 4 jenis) Strategi ini memungkinkan pengecekan konsistensi dalam

pembelajaran model dengan menggunakan 80% dari data untuk pelatihan dan sisanya sebagai data validasi yang tidak dilihat model selama fase pelatihan.

D. Pengujian Model CNN

Pengukuran performa model CNN merupakan tahapan penting dalam memahami keberhasilan model dalam tugas klasifikasi. Performa model CNN untuk klasifikasi kualitas ban kendaraan dipantau dan dievaluasi selama pelatihan dan validasi. Grafik akurasi dan loss memberikan wawasan penting tentang proses pembelajaran model. Pseudocode Untuk pengujian model CNN.

```
BEGIN
// Create a sequential model
SET model = Sequential()

// Add convolutional layers
model.add(Conv2D(256, (3, 3), activation='relu', input_shape=(150, 150, 3)))
model.add(MaxPooling2D(2, 2))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D(2, 2))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(2, 2))
model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(MaxPooling2D(2, 2))

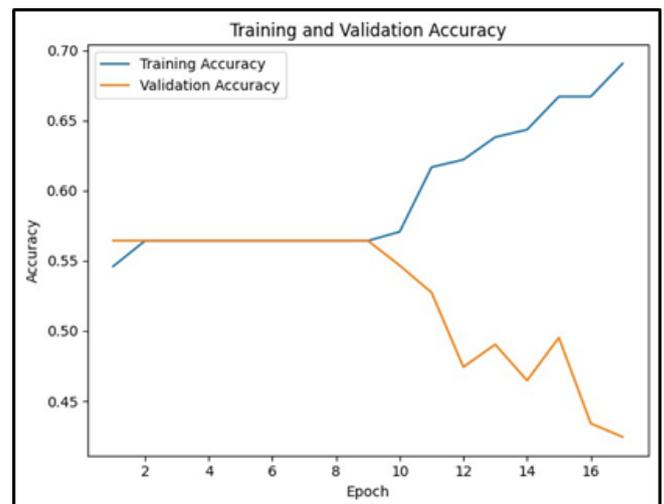
// Flatten the output for fully connected layers
model.add(Flatten())

// Add dropout layer
model.add(Dropout(0.2))

// Add fully connected layers
model.add(Dense(512, activation='relu'))
model.add(Dense(2, activation='softmax'))

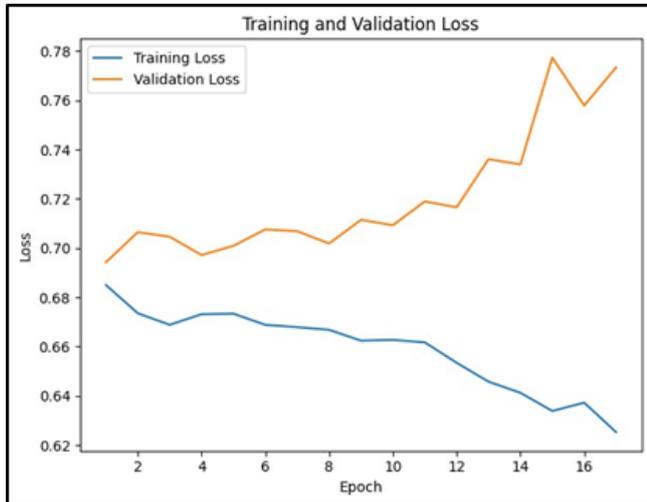
// Compile the model
model.compile(loss='categorical_crossentropy',
optimizer=Adam(learning_rate=0.00003), metrics=['accuracy'])
END
```

1) Model CNN Split Rasio 60:40



Gambar 4. Grafik akurasi pelatihan dan validasi CNN split ratio 60:40

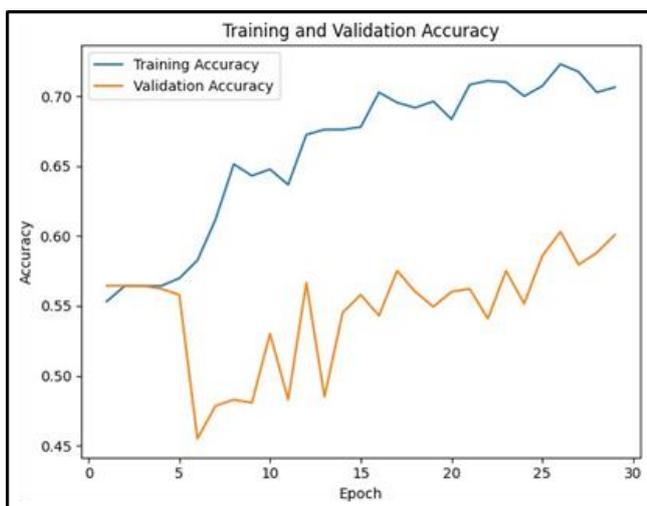
Gambar 4 menunjukkan grafik akurasi pelatihan dan validasi CNN dengan split ratio 60:40. Akurasi pelatihan meningkat dari sekitar 0.55 pada epoch awal hingga sekitar 0.68 pada epoch ke-17, menandakan model belajar dengan baik dari data pelatihan. Sebaliknya, akurasi validasi stabil pada sekitar 0.55 hingga epoch ke-10, lalu menurun drastis hingga sekitar 0.43 pada epoch ke-17, mengindikasikan adanya overfitting.



Gambar 5. Grafik loss pelatihan dan validasi CNN split ratio 60:40

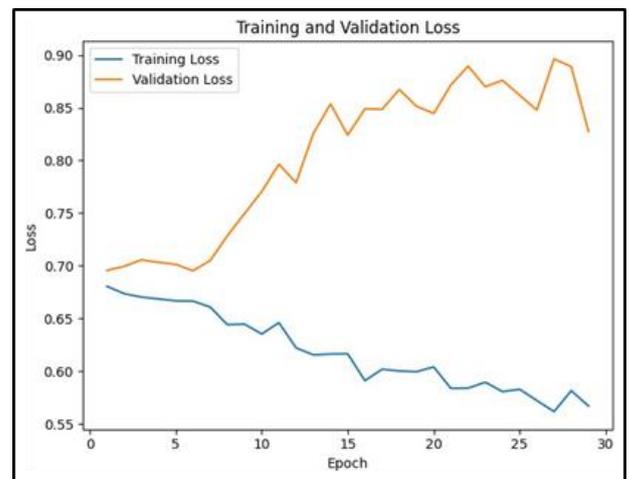
Gambar 5, grafik loss pelatihan dan validasi CNN dengan split ratio 60:40 menunjukkan bahwa loss pelatihan konsisten menurun dari sekitar 0.68 menjadi 0.62 pada epoch ke-17, mengindikasikan peningkatan kemampuan model dalam memprediksi data pelatihan. Namun, loss validasi yang awalnya stabil mulai meningkat setelah epoch ke-10, mencapai sekitar 0.78 pada epoch ke-17, yang menunjukkan adanya overfitting karena model tidak mampu mempertahankan performa yang baik pada data validasi.

2) Model CNN Split Ratio 70:30



Gambar 6. Grafik akurasi pelatihan dan validasi CNN split ratio 70:30

Pada gambar 6 grafik akurasi pelatihan dan validasi CNN dengan split ratio 70:30, terlihat bahwa akurasi pelatihan model meningkat secara bertahap. Pada epoch awal, akurasi pelatihan dimulai dari sekitar 0.55 dan terus meningkat hingga mencapai sekitar 0.70 pada epoch ke-29. Ini menunjukkan bahwa model belajar dengan baik dari data pelatihan dan menjadi semakin akurat seiring dengan bertambahnya epoch. Namun, berbeda dengan akurasi pelatihan, akurasi validasi justru menunjukkan fluktuasi. Akurasi validasi terlihat fluktuatif mulai dari epoch ke-6 hingga akhirnya pada epoch ke-9 akurasi validasi hanya bisa mencapai akurasi sekitar 0.60. Volatilitas akurasi validasi ini juga mengindikasikan adanya masalah overfitting seperti pada model dengan split ratio 60:40 sebelumnya.

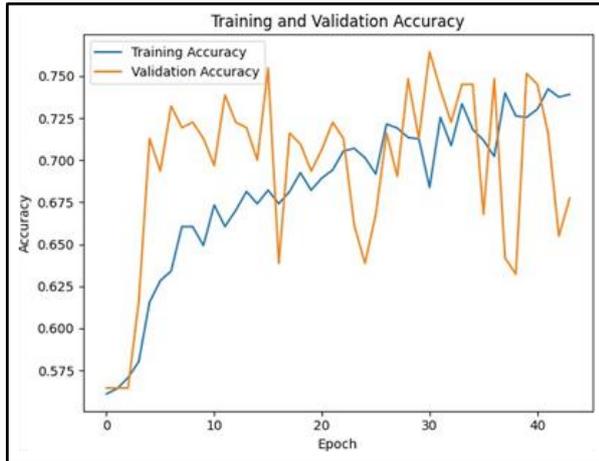


Gambar 7. Grafik loss pelatihan dan validasi CNN split ratio 70:30

Pada gambar 7 dari grafik loss pelatihan dan validasi CNN dengan split ratio 70:30, terlihat bahwa loss pelatihan menurun secara konsisten dari sekitar 0.68 hingga sekitar 0.57 pada epoch ke-29, menunjukkan kemampuan model dalam memprediksi data pelatihan yang semakin baik. Namun, loss validasi menunjukkan tren kenaikan setelah epoch ke-6, mencapai sekitar 0.85 pada epoch ke-29, mengindikasikan adanya overfitting pada model CNN ini.

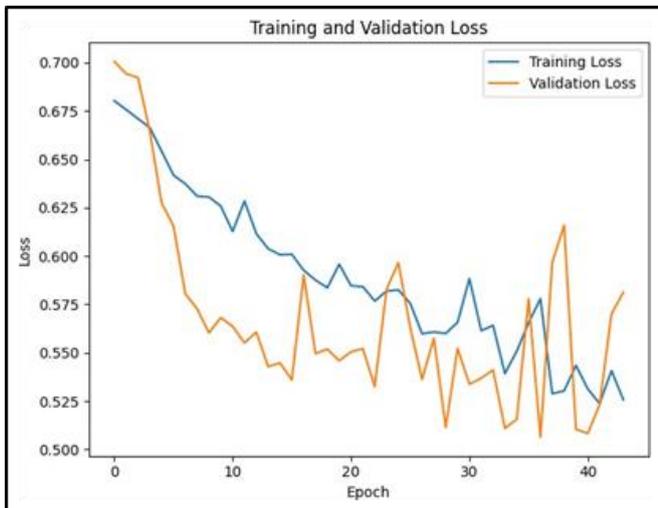
3) Model CNN Split Ratio 80:20

Pada gambar 8, Grafik Pelatihan dan Validasi CNN dengan rasio 80:20 menunjukkan tren yang signifikan. Akurasi pelatihan secara konsisten meningkat dari awal, mencerminkan keberhasilan model dalam mengekstraksi fitur penting dari data pelatihan dengan efektif hingga mencapai sekitar 0.700 hingga 0.750 setelah epoch ke-20. Meskipun demikian, akurasi validasi menunjukkan fluktuasi yang lebih besar dan variabilitas yang signifikan, terutama pada bagian akhir grafik dari epoch ke-30 hingga 40, menandakan potensi overfitting dan kemungkinan penurunan kemampuan model dalam mengadaptasi fitur-fitur baru pada dataset validasi.



Gambar 8. Grafik akurasi pelatihan dan validasi CNN split ratio 80:20

Dengan perbedaan yang mencolok antara akurasi pelatihan dan validasi, hal ini mengindikasikan tantangan dalam menggeneralisasi pengetahuan model di luar dataset pelatihan yang spesifik.

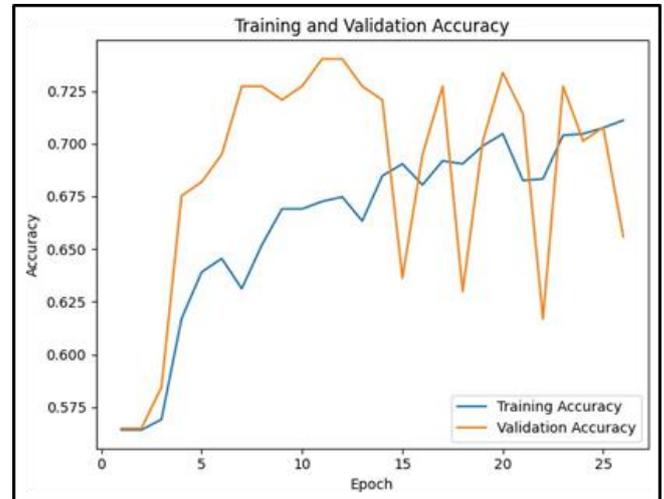


Gambar 9. Grafik loss pelatihan dan validasi CNN split ratio 80:20

Selanjutnya pada gambar 9, terlihat bahwa loss pelatihan menunjukkan penurunan stabil, menandakan peningkatan dalam kemampuan model untuk mengklasifikasikan sampel dari dataset pelatihan dengan akurasi yang meningkat. Namun, loss pada validasi menunjukkan volatilitas yang signifikan dengan lonjakan tajam setelah epoch ke- 20, mengindikasikan kesulitan model dalam mempertahankan konsistensi performa saat berinteraksi dengan data validasi yang mungkin memiliki karakteristik yang berbeda dari data pelatihan. Divergensi ini mengonfirmasi adanya overfitting, di mana meskipun model berhasil dalam pelatihan, kinerja pada data validasi tidak seimbang, ditandai dengan lonjakan yang tajam dalam loss validasi yang disebabkan oleh sampel-sampel yang mungkin tidak sesuai dengan distribusi umum yang dipelajari oleh model.

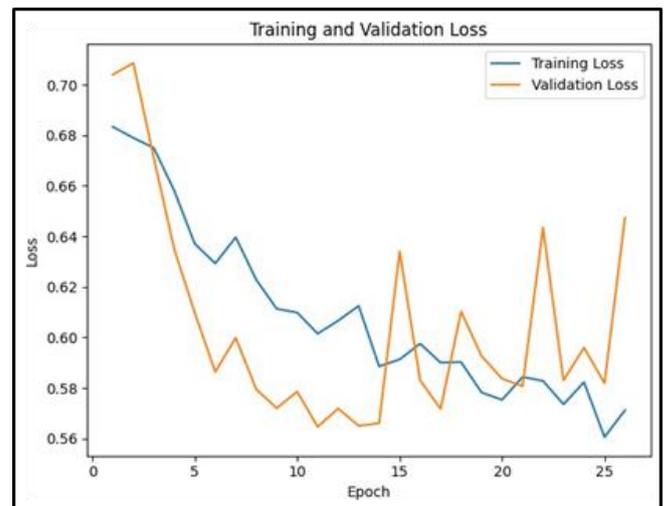
4) *Model CNN Split Ratio 90:10*

Pada gambar 10 terlihat kurva akurasi untuk pelatihan dan validasi. Awalnya, akurasi pelatihan meningkat secara konsisten dari sekitar 0.575 hingga mendekati 0.675 pada epoch ke-10, menandakan bahwa model berhasil mempelajari fitur-fitur data pelatihan dengan baik.



Gambar 10. Grafik akurasi pelatihan dan validasi CNN split ratio

Namun, setelah epoch ke- 10, akurasi pelatihan mulai fluktuatif, tetap meningkat hingga sekitar 0.700 pada epoch ke-25. Di sisi lain, akurasi validasi menunjukkan fluktuasi yang lebih besar sepanjang pelatihan, dengan lonjakan awal mencapai sekitar 0.725 pada epoch ke-5, diikuti oleh variasi yang signifikan. Hal ini mencerminkan kesulitan model dalam menggeneralisasi pengetahuan terhadap data validasi yang mungkin memiliki distribusi yang berbeda dari data pelatihan.



Gambar 11. Grafik akurasi pelatihan dan validasi CNN split ratio

Selanjutnya pada gambar 11 menampilkan kurva loss untuk pelatihan dan validasi. Loss pelatihan menunjukkan penurunan yang stabil, dari sekitar 0.70 pada epoch pertama

hingga mendekati 0.56 pada epoch ke-25. Penurunan ini menunjukkan peningkatan kemampuan model dalam mengklasifikasikan data pelatihan dengan lebih akurat seiring bertambahnya epoch. Namun, loss validasi menunjukkan volatilitas yang signifikan. Meskipun terjadi penurunan umum pada epoch 1-5, setelah itu terjadi fluktuasi yang lebih besar. Lonjakan tajam pada loss validasi, terutama setelah epoch 10, mengindikasikan bahwa model menghadapi kesulitan dalam mempertahankan konsistensi kinerja pada data validasi, mungkin karena distribusi data yang berbeda atau adanya noise yang tidak terlihat pada data pelatihan.

5) Analisis Hasil Evaluasi Model CNN

Evaluasi kinerja model CNN yang dikembangkan untuk klasifikasi kualitas ban kendaraan menggunakan metrik akurasi, presisi, dan recall memberikan informasi tentang efektivitas dan keandalan model ini. Melalui penggunaan nilai-nilai yang diperoleh dari *confusion matrix*, dapat menilai kemampuan model dalam memenuhi kriteria kinerja yang telah ditetapkan dan mengidentifikasi area potensial untuk peningkatan lebih lanjut.

TABEL II.
EVALUASI MODEL CNN

Split Ratio	Akurasi	Presisi	Recall
60:40	60.67%	62.67%	60.26%
70:30	65%	68%	64.15%
80:20	76.67%	77.33%	76.32%
90:10	72.67%	84%	68.48%

Berdasarkan tabel 2, pada split ratio 60:40 model mencapai akurasi 60.67%, presisi 62.67%, dan recall 60.26%. Meskipun nilai akurasi ini mencerminkan kemampuan dasar model dalam membedakan antara ban 'defective' dan 'good', performanya relatif rendah dibandingkan dengan split ratio lainnya. Rendahnya presisi dan *recall* menunjukkan bahwa model sering salah dalam mengklasifikasikan ban, baik dalam hal false positives maupun false negatives. Rasio ini mungkin lebih cocok untuk tahap eksplorasi awal, namun kurang optimal untuk pelatihan model akhir.

Selanjutnya dengan split ratio 70:30, model menunjukkan peningkatan performa dengan akurasi 65%, presisi 68%, dan recall 64.15%. Peningkatan ini menunjukkan bahwa lebih banyak data pelatihan membantu model dalam belajar pola yang lebih kompleks dari dataset. Namun, meskipun ada peningkatan dibandingkan dengan split ratio 60:40, hasilnya masih menunjukkan adanya ruang untuk perbaikan, terutama dalam meningkatkan recall untuk memastikan lebih sedikit ban cacat yang lolos deteksi.

Rasio 80:20 memberikan hasil terbaik di antara semua split ratio yang diuji, dengan akurasi 76.67%, presisi 77.33%, dan recall 76.32%. Tingginya nilai-nilai ini menunjukkan bahwa model mampu melakukan prediksi yang lebih akurat dan andal. Tingkat presisi yang tinggi mengindikasikan bahwa sebagian besar prediksi 'defective' adalah benar, sedangkan nilai recall yang tinggi menunjukkan bahwa model juga

efektif dalam mendeteksi hampir semua ban cacat. Oleh karena itu, rasio ini memberikan keseimbangan terbaik antara pelatihan dan validasi, memastikan model dapat melakukan generalisasi dengan baik pada data baru.

Pada split ratio 90:10, model mencapai akurasi 72.67%, presisi 84%, dan recall 68.48%. Meskipun akurasi dan recall sedikit lebih rendah dibandingkan dengan split ratio 80:20, presisi model mencapai nilai tertinggi pada rasio ini. Tingginya presisi menunjukkan bahwa model sangat efektif dalam mengidentifikasi ban 'defective', namun nilai recall yang lebih rendah mengindikasikan adanya jumlah false negatives yang lebih tinggi. Rasio ini mungkin kurang optimal karena data validasi yang lebih sedikit dapat menyebabkan model *overfitting*, mengurangi kemampuannya untuk menangani data baru secara efektif

IV. KESIMPULAN

Dari penelitian menggunakan model CNN untuk klasifikasi kualitas ban kendaraan, disimpulkan bahwa split ratio 80:20 memberikan hasil terbaik dengan akurasi 76.67%, presisi 77.33%, dan recall 76.32%. Temuan ini menegaskan pentingnya jumlah data pelatihan yang memadai untuk meningkatkan kemampuan generalisasi model dalam mengenali pola cacat dan menghasilkan prediksi yang akurat. Model CNN juga berhasil mendeteksi cacat ban dengan tingkat presisi dan recall yang tinggi pada split ratio optimal, meskipun masih diperlukan penelitian lebih lanjut untuk mengatasi *trade-off* antara presisi dan recall serta untuk mengoptimalkan distribusi data dan arsitektur model yang digunakan.

DAFTAR PUSTAKA

- [1] A. Ufriandi, "Analisis Tingkat Keausan Terhadap Pemakaian Ban Merek A, B Dan C Menggunakan Ban Standar 90/90-14 46 P," *J. Surya Tek.*, vol. 8, no. 1, pp. 282–288, Jul. 2021, doi: 10.37859/jst.v8i1.2678.
- [2] A. Arthono and V. A. Permana, "Perencanaan Perkerasan Lentur Jalan Raya Menggunakan Metode Analisa Komponen SNI 1732-1989-F Ruas Jalan Raya Mulya Sari Kecamatan Pamanukan Sampai Kecamatan Binong Kabupaten Subang Propinsi Jawa Barat," *J. Komposit*, vol. 6, no. 1, p. 41, 2022, doi: 10.32832/komposit.v6i1.6740.
- [3] M. Ushuludin, S. F. C. Haviana, and I. M. I. Subroto, "Sistem Deteksi Masker Pada Wajah Menggunakan Convolutional Neural Network Arsitektur VGG16," *Transm. J. Ilm. Tek. Elektro*, vol. 25, no. 4, pp. 179–185, Dec. 2023, doi: 10.14710/transmisi.25.4.179-185.
- [4] A. Saxena, "An Introduction to Convolutional Neural Networks," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, pp. 943–947, Dec. 2022, doi: 10.22214/ijrasat.2022.47789.
- [5] L. Puspitasari, "Analisa Performance Ban Pada Unit Produksi Overburden Hd-785 Terhadap Produktivitas Tambang Batubara," *KURVATEK*, vol. 5, no. 1, pp. 69–79, May 2020, doi: 10.33579/krvtk.v5i1.1775.
- [6] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging*, vol. 9, no. 4, pp. 611–629, Aug. 2018, doi: 10.1007/s13244-018-0639-9.
- [7] N. Rahmayuna, K. Adi, and R. Kusumaningrum, "Tableware Ceramics Defect Detection Using Morphological Operation Approach," in *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, Dec.

- 2021, pp. 412–416. doi: 10.1109/ISRITI54043.2021.9702806.
- [8] M. Hatami Varjovi, M. F. Talu, and K. Hanbay, “Fabric Defect Detection Using Customized Deep Convolutional Neural Network for Circular Knitting Fabrics,” *Türk Doğa ve Fen Derg.*, vol. 11, no. 3, pp. 160–165, Sep. 2022, doi: 10.46810/tdfd.1108264.
- [9] A. L. A. Shidiq, E. SUhartono, and S. Saidah, “Klasifikasi Kecacatan Ban Untuk Mengendalikan Kualitas Produk Menggunakan Model CNN Dengan Arsitektur VGG-16 Classification Of Tire Defect To Control Product Quality Using Cnn Model With VGG-16 Architecture,” *e-Proceeding Eng.*, vol. 8, no. 6, pp. 3216–3225, 2022.
- [10] B. Biantoro and Hernadewita, “K-means clustering on quality of radial run out tires,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1034, no. 1, p. 012122, Feb. 2021, doi: 10.1088/1757-899X/1034/1/012122.
- [11] I. E. Hendarawan, Riza Ibnu Adam, and Chaerur Rozikin, “Klasifikasi Retak Ban Kendaraan Menggunakan Arsitektur ResNet50,” *SATIN - Sains dan Teknol. Inf.*, vol. 9, no. 1, pp. 22–32, Jun. 2023, doi: 10.33372/stn.v9i1.902.
- [12] Nugroho Febriyanto, Cahya Rahmad, and C. B. Vista, “Deteksi Kerusakan Dan Kelayakan Ban Mobil Berdasarkan Ekstraksi Tekstur Menggunakan Gray Level Cooccurrence Matrix Dengan Metode Jaringan Syaraf Tiruan,” *J. Inform. Polinema*, vol. 7, no. 4, pp. 27–32, Aug. 2021, doi: 10.33795/jip.v7i4.469.
- [13] Hendri Candra Mayana and Desmarita Leni, “Deteksi Kerusakan Ban Mobil Menggunakan Convolutional Neural Network dengan Arsitektur ResNet-34,” *J. Surya Tek.*, vol. 10, no. 2, pp. 842–851, Dec. 2023, doi: 10.37859/jst.v10i2.6336.
- [14] A. Nurjati, R. Magdalena, and I. N. A. Ramatryana, “Deteksi Kerusakan Permukaan Ban Pada Angkutan Umum Dengan Metode Local Binary Pattern Dan Klasifikasi Nearest Neighbor,” vol. 3, no. 2, pp. 1986–1993, 2016.
- [15] M. F. Herlambang, A. N. Hermana, and K. R. Putra, “Pengenalan Karakter Huruf Braille dengan Metode Convolutional Neural Network,” *Syst. Inf. Syst. Informatics J.*, vol. 6, no. 2, pp. 20–26, Jan. 2021, doi: 10.29080/systemic.v6i2.969.