

PERBANDINGAN KINERJA K-NEAREST NEIGHBORS DAN CONVOLUTIONAL NEURAL NETWORK DALAM KLASIFIKASI CITRA KONDISI PERMUKAAN JALAN

Fenny Jong¹

¹ Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara,
Jln. Letjen S. Parman No. 1, Jakarta, 11440, Indonesia

E-mail: ¹fenny.535210001@stu.untar.ac.id

Abstrak

Peningkatan kualitas infrastruktur jalan merupakan aspek penting dalam pembangunan transportasi dan keselamatan pengguna jalan. Penilaian kondisi permukaan jalan secara otomatis dapat mempercepat pemeliharaan dan perbaikan jalan. Penelitian ini membandingkan dua metode klasifikasi yaitu K-Nearest Neighbors (KNN) dan Convolutional Neural Network (CNN) untuk menilai kondisi permukaan jalan berdasarkan citra digital. Fitur tekstur diekstraksi menggunakan Gray Level Co-occurrence Matrix (GLCM), meliputi Contrast, Homogeneity, Energy, dan lainnya, untuk meningkatkan akurasi klasifikasi pada KNN, sementara pada CNN ekstraksi fitur dan klasifikasi dilakukan secara otomatis. Penelitian ini menggunakan 1.500 gambar permukaan jalan dalam tiga kondisi berbeda: mulus, retak, dan berlubang masing-masing kondisi memiliki 500 gambar dengan resolusi 300x300 piksel. Hasil penelitian menunjukkan algoritma KNN mencapai akurasi 57,2%, sementara CNN menunjukkan performa paling terbaik dengan akurasi 93,8% untuk 80% latihan dan 20% pengujian.

Kata kunci— Convolutional Neural Network (CNN), Gray Level Co-occurrence Matrix (GLCM), Infrastruktur jalan, K-Nearest Neighbors (KNN), Klasifikasi.

Abstract

Improving road infrastructure quality is an important aspect of transportation development and road user safety. Automatically assessing road surface conditions can accelerate maintenance and repair efforts. This study compares two classification methods, K-Nearest Neighbors (KNN) and Convolutional Neural Network (CNN), to evaluate road surface conditions based on digital images. Texture features are extracted using the Gray Level Co-occurrence Matrix (GLCM), including Contrast, Homogeneity, Energy, and others, to enhance the classification accuracy in KNN, while feature extraction and classification in CNN are performed automatically. The dataset used in this research consists of 1500 images of road surfaces with three different conditions: smooth, cracked, and potholes. Each condition contains 500 images with a resolution of 300x300 pixels. The results show that the KNN algorithm achieves an accuracy of 57.2%, while CNN demonstrates the best performance with an accuracy of 93.8% for 80% training data and 20% testing data.

Keywords— Convolutional Neural Network (CNN), Road Infrastructure, Gray Level Co-occurrence Matrix (GLCM), K-Nearest Neighbors (KNN), Classification.

1. PENDAHULUAN

Peningkatan kualitas infrastruktur jalan merupakan aspek penting dalam pembangunan transportasi dan keselamatan pengguna jalan. Permukaan jalan memiliki peran krusial dalam mendukung kelancaran mobilitas serta aktivitas masyarakat sehari-hari [21]. Jalan yang baik tidak hanya meningkatkan efisiensi perjalanan, tetapi juga berpengaruh pada keselamatan dan kenyamanan pengguna. Kondisi permukaan jalan yang baik dapat memperpanjang umur pakai jalan tersebut dan mengurangi potensi risiko kecelakaan. Oleh karena itu, pemantauan dan klasifikasi kondisi permukaan jalan secara berkala sangat penting untuk memastikan kualitas jalan tetap terjaga. Oleh karena itu, pemantauan dan klasifikasi kondisi permukaan jalan menjadi hal yang penting untuk dilakukan secara berkala [1]. Pengecekan kualitas jalan raya biasanya dilakukan secara manual yang memerlukan banyak waktu dan tenaga. Jumlah jalan yang banyak dan proses pengecekan manual merupakan masalah utama dalam perawatan jalan raya [2]. Salah satu cara untuk mendeteksi dan menganalisis kondisi jalan adalah dengan menggunakan teknik pembelajaran mesin untuk mengklasifikasikan gambar atau citra permukaan jalan.

Hubungan spasial antar piksel diamati oleh ekstraksi fitur Gray Level Co-occurrence Matrix (GLCM) yang kemudian diklasifikasikan dengan K-Nearest Neighbors (K-NN) dan Convolutional Neural Network (CNN) merupakan dua algoritma pembelajaran mesin yang populer dan sering digunakan dalam tugas klasifikasi, termasuk klasifikasi citra. K-NN merupakan metode pembelajaran berbasis *instance* yang sederhana namun efektif dalam berbagai masalah klasifikasi [3][4], sementara CNN sebagai salah satu bentuk arsitektur jaringan saraf dalam, terbukti sangat unggul dalam pemrosesan dan pengenalan citra berkat kemampuannya dalam mengekstraksi fitur secara otomatis [5].

Penelitian sebelumnya telah banyak membahas penggunaan algoritma K-NN dan CNN untuk berbagai aplikasi klasifikasi citra, seperti pengenalan objek, pengklasifikasian citra medis, hingga analisis citra jalan. Namun, perbandingan kinerja kedua algoritma ini dalam klasifikasi kondisi permukaan jalan masih terbatas. Beberapa penelitian menunjukkan bahwa CNN lebih unggul dalam hal akurasi dan kemampuan generalisasi dibandingkan metode lain, namun metode yang lebih sederhana seperti K-NN tetap relevan dalam situasi tertentu dengan dataset yang terbatas atau ketika interpretabilitas model sangat penting. Misalnya, Penelitian Ridol Liusman dkk yang mengembangkan model CNN berbasis arsitektur Xception untuk klasifikasi kualitas permukaan jalan, dengan hasil akurasi mencapai 90,11% [6]. Penelitian oleh Budiarto dkk. mengenai sistem klasifikasi permukaan jalan dan penghindaran jalan berlubang pada kursi roda pintar menggunakan metode MobileNetV3-Small mencapai akurasi 88,076% dengan waktu komputasi rata-rata 0,96048 detik [7]. Selain itu, Penelitian oleh Muchtar dkk. berjudul penerapan analisis berbasis fraktal dalam klasifikasi citra retakan pada permukaan jembatan beton menggunakan K-Nearest Neighbor (K-NN) berhasil mencapai akurasi 97,6%, dengan presisi 97,659% dan recall 97,6% [8].

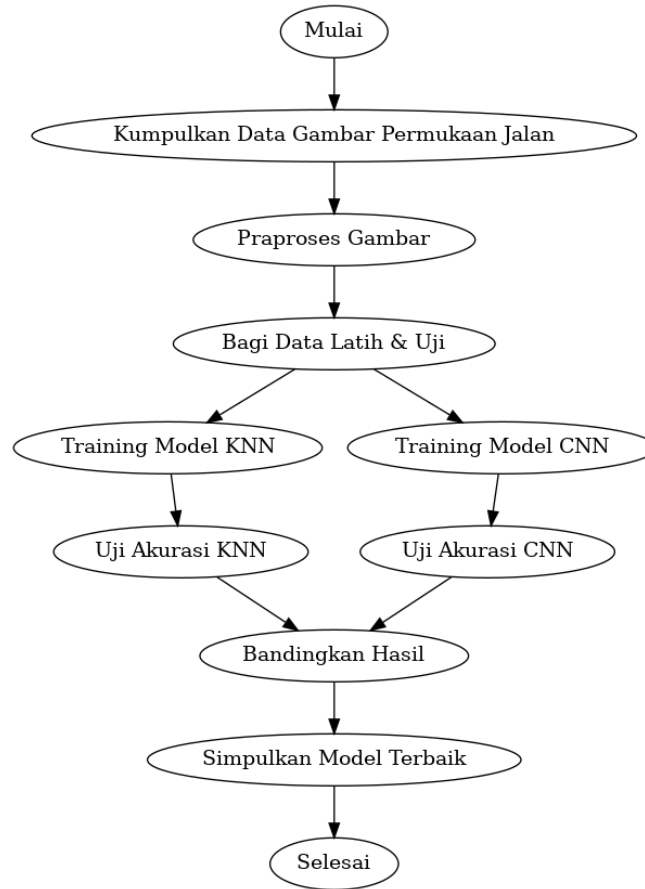
Penelitian ini bertujuan untuk membandingkan algoritma KNN dan CNN dalam klasifikasi kondisi permukaan jalan. Diharapkan metode ini akan berfungsi sebagai dasar untuk pengembangan sistem pemantauan kondisi jalan yang menggunakan pembelajaran mesin yang lebih efisien.

2. METODE PENELITIAN

2.1 Tahap Penelitian

Tahapan proses pengembangan model untuk pengenalan gambar permukaan jalan digambarkan dalam flowchart yang ditunjukkan pada Gambar 1. Dimulai dengan data gambar permukaan jalan dikumpulkan, yang kemudian diproses melalui tahap praproses gambar. Selanjutnya, dikelompokkan data latihan dan data pengujian. Model K-Nearest Neighbors (KNN) dan Convolutional Neural Network (CNN) kemudian dilatih dengan data latih dan kemudian diuji

akurasi dengan data pengujian. Hasil dari kedua model dievaluasi sebelum memilih model yang lebih baik dan pada akhirnya, model yang paling akurat diputuskan. Setelah itu, proses diakhiri.



Gambar 1 Flowchart Penelitian

2.2 Gray-Level Co-occurrence Matrix (GLCM)

Gray-Level Co-occurrence Matrix (GLCM) adalah metode ekstraksi fitur untuk analisis tekstur yang menghitung probabilitas hubungan antara dua piksel berdasarkan intensitas, jarak, dan orientasi sudut (0° , 45° , 90° , 135°) dengan jarak antar piksel 1. Tahapannya meliputi pembentukan matriks awal, penggabungan matriks, perhitungan probabilitas elemen, dan ekstraksi fitur dari semua arah sudut. Persamaan rumus GLCM sebagai berikut [10]:

$$Contrast = \sum_{i,j=0}^{N-1} P_{(ij)} (i - j)^2 \quad (1)$$

$$Energy = \sum_{i,j=0}^{N-1} (P_{(ij)})^2 \quad (2)$$

$$Entropy = \sum_{i,j=0}^{N-1} -\ln(P_{(ij)}) P_{(ij)} \quad (3)$$

$$Homogeneity = \sum_{i,j=0}^{N-1} \frac{P_{(ij)}}{(1+(j-i)^2)} \quad (4)$$

$$correlation = \frac{\sum_i \sum_j (i-j) P(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (5)$$

2.3 K-Nearest Neighbors (KNN)

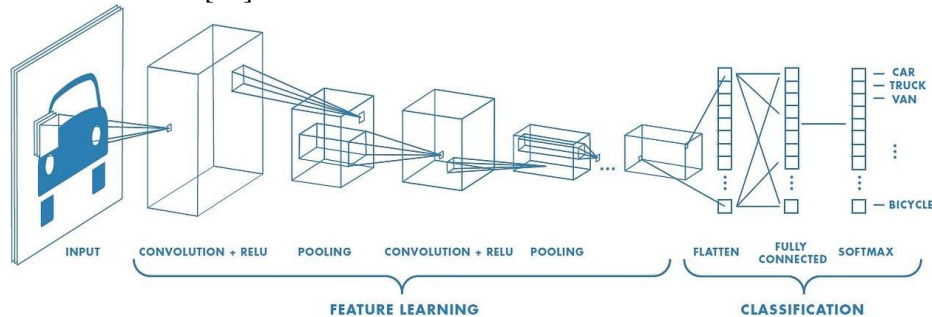
K-Nearest Neighbors (KNN) adalah algoritma pembelajaran mesin untuk regresi dan klasifikasi. KNN mengklasifikasikan titik data menggunakan K tetangga terdekat dan mayoritas kelas tetangga. Algoritma ini menentukan kedekatan titik data dengan menggunakan jarak antara mereka. Ada persamaan berikut untuk rumus KNN [10][11]:

$$d(A, B) = \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad (6)$$

d adalah jarak data, i adalah variabel data, n adalah dimensi data A_i adalah data latih B_i adalah data pengujian.

2.4 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah metode populer untuk pengolahan citra karena kemampuannya mengolah data gambar besar dengan akurasi tinggi dan efisiensi proses. CNN unggul dalam menyimpan informasi spasial, dengan struktur yang mencakup input, hidden layer, klasifikasi, dan output, serta menggunakan fungsi aktivasi seperti Softmax untuk menghasilkan klasifikasi [12]. Gambar 1 adalah visualisasi CNN.



Gambar 1 Convolutional Neural Network (CNN)

Berikut ini penjelasan tiap struktur CNN:

a. *Input Layer*

Lapisan input adalah lapisan pertama yang menyimpan nilai piksel gambar yang dimasukkan [13].

b. *Convolution Layer*

Lapisan ini mengekstrak fitur penting dari citra dengan operasi konvolusi menggunakan filter. Setiap convolutional layer mengidentifikasi fitur sederhana pada lapisan awal dan fitur lebih kompleks pada lapisan dalam [14]. Rumus *Convolution Layer* sebagai berikut [13]:

$$(of)_{ij} = (w_f * R)_{ij} + b_{ij} \quad (7)$$

Pada persamaan ini, o_{ij} adalah nilai keluaran di posisi i, j yang diperoleh dari operasi konvolusi antara filter w_j dan input R , menghasilkan $(w_f * R)_{ij}$. Nilai b_{ij} kemudian ditambahkan untuk meningkatkan kemampuan model menangkap informasi kompleks.

c. *Activation Layer*

Fungsi aktivasi menentukan apakah sinyal diteruskan ke lapisan berikutnya [14]. Fungsi umum yang digunakan ReLU yaitu mengubah nilai input negatif menjadi nol, mempercepat komputasi, dan mengurangi risiko overfitting [15]. Sigmoid yaitu fungsi non-linear yang menghasilkan nilai antara 0 dan 1, sering digunakan untuk klasifikasi biner [16][25]. Softmax yaitu menghitung probabilitas dan menentukan kelas berdasarkan nilai yang dinormalisasi [16].

d. *Pooling Layer*

Pooling layer digunakan untuk mengurangi *feature map* konvolusi, mempercepat komputasi, dan mengekstraksi fitur dominan [17].

e. *Flatten*

Proses ini mengubah *feature map* menjadi vektor yang berdimensi satu untuk memudahkan klasifikasi [18].

f. *Fully Connected Layer*

Setelah proses convolution dan pooling hasilnya kemudian diubah menjadi vektor input untuk *fully connected layer* yang setiap nilai dalam vektor ini mewakili probabilitas untuk kelas tertentu dari gambar yang diklasifikasikan [19].

2.5 Metode Evaluasi

Metode evaluasi digunakan untuk menilai kinerja model dalam tugas pembelajaran mesin. Beberapa metrik evaluasi klasifikasi yang digunakan antara lain akurasi, precision, recall, dan F1-score. Metode ini membantu dalam memahami seberapa baik model dalam memprediksi hasil yang benar dan menghindari kesalahan. Confusion Matrix menggambarkan jumlah prediksi untuk setiap kombinasi kelas aktual dan kelas prediksi (*True Positive*, *True Negative*, *False Positive*, *False Negative*) [15][22][23]. Gambar 2 adalah visualisasi confusion matrix.

	Prediction outcome		
	positive	negative	
positive	TP	FN	TP + FN
negative	FP	TN	FP + TN
	TP + FP	FN + TN	

Gambar 2 Confusion Matrix

Akurasi mengukur persentase prediksi yang benar dari total data. Rumus akurasi sebagai berikut [20]:

$$akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

Precision adalah ukuran jumlah prediksi positif yang benar-benar relevan. Rumus precision sebagai berikut [20][24]:

$$precision = \frac{TP}{TP + FP} \quad (9)$$

Kemampuan model untuk menemukan semua kasus positif diukur dengan recall. Rumus recall sebagai berikut [20]:

$$recall = \frac{TP}{TP + FN} \quad (10)$$

F1-Score adalah skor rata-rata untuk presisi dan recall yang mengimbangi keduanya. Rumus F1-Score sebagai berikut [20]:

$$F1 - Score = \frac{2 * Precision * recall}{Precision + recall} \quad (11)$$

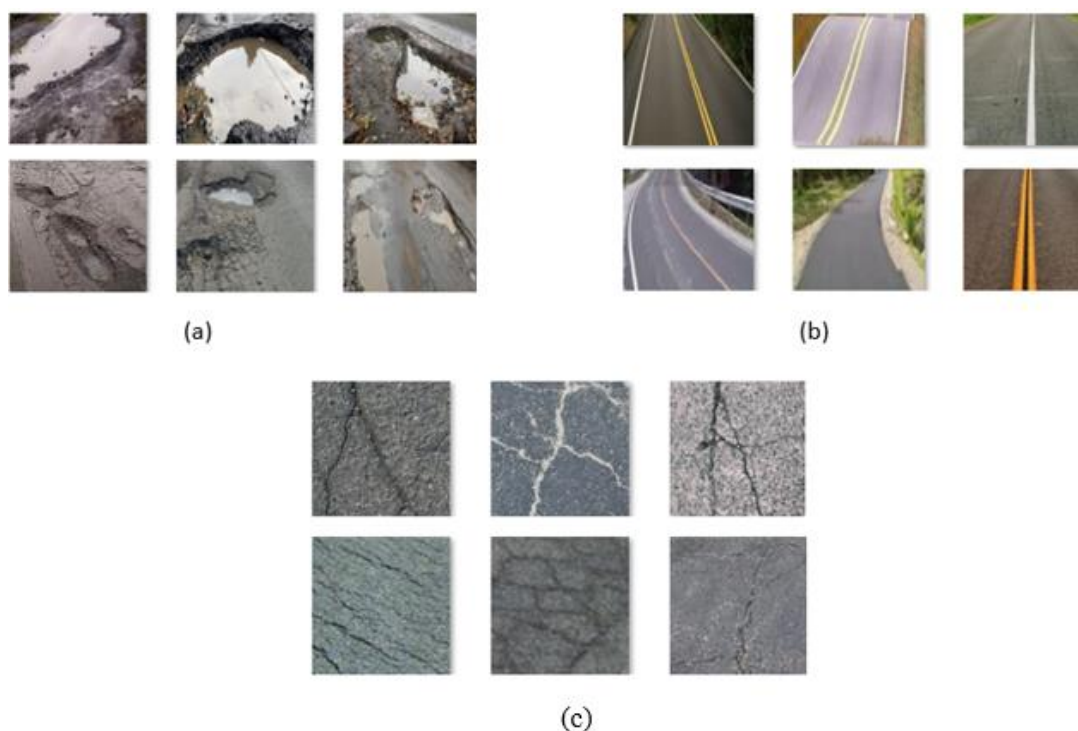
3. HASIL DAN PEMBAHASAN

3.1 Sumber Data

Dataset dalam penelitian ini diperoleh melalui *web scraping* dari berbagai sumber *online* yang menyediakan gambar kondisi permukaan jalan. Dataset terdiri dari 500 gambar per kelas, meliputi kondisi jalan mulus, retak, dan berlubang, yang digunakan untuk klasifikasi. Penjelasan lebih lanjut tentang kelas-kelas ini terdapat pada Tabel 1, sementara contoh gambar dataset disajikan pada Gambar 3. Penggunaan data yang digunakan dalam penelitian ini pada referensi [21].

Tabel 1 Kelas Kondisi Permukaan Jalan

Kelas	Jumlah Data	Deksripsi
Jalan Mulus	500 Gambar	Gambar yang menunjukkan permukaan jalan yang rata dan tidak terdapat kerusakan atau lubang.
Jalan Retak	500 Gambar	Gambar yang menggambarkan jalan dengan kerusakan permukaan seperti retakan atau degradasi ringan.
Jalan Berlubang	500 Gambar	Gambar yang memperlihatkan jalan dengan lubang besar atau kerusakan yang cukup parah.



Gambar 3 Contoh Dataset (a) Jalan Berlubang, (b) Jalan Mulus, dan (c) Jalan Retak.

3.2 Preprocessing data

Tahapan *preprocessing* dilakukan untuk memastikan data sesuai kebutuhan model. Semua gambar diubah ukurannya menjadi 300x300 piksel untuk menyamakan dimensi dan mengurangi kebutuhan memori tanpa kehilangan detail penting. Dataset dibagi dalam dua skenario pelatihan dan pengujian: 80:20 dan 70:30. Pada skenario 80:20, 80% dialokasikan untuk

dilatih dan 20% untuk pengujian. Sedangkan pada skenario 70:30, 70% data dialokasikan untuk dilatih dan 30% untuk pengujian.

3.3 Pengujian *K-Nearest Neighbors (KNN)*

Pengujian *K-Nearest Neighbors (KNN)* dilakukan dengan $k = 3$ dan berbagai nilai random state (1, 10, 100, 1000, dan 10.000). Selain itu, data pelatihan dan pengujian dibagi menjadi dataset 80:20 dan 70:30 untuk mendapatkan informasi tekstual dari gambar, Gray-Level Co-occurrence Matrix (GLCM) digunakan untuk mengekstraksi fitur. Konfigurasi KNN dengan pembagian dataset 70% data Latihan dan 30% data pengujian ditunjukkan dalam Tabel 3, dan Tabel 2 menunjukkan 80% data latihan dan 20% data pengujian.

Tabel 2 Hasil Pengujian KNN 80% Data Latihan dan Data pengujian 20%

Random State	Precision	Recall	F1-score	Akurasi	Waktu Pelatihan (detik)
1	0.57	0.56	0.55	0.56	0.0039
10	0.58	0.58	0.58	0.59	0.0000
100	0.58	0.58	0.58	0.59	0.0105
1000	0.57	0.58	0.57	0.56	0.0028
10000	0.57	0.56	0.56	0.56	0.0021

Pembagian dataset 80:20 pada Tabel 2 menunjukkan kinerja yang cukup stabil dengan nilai precision dan recall berkisar antara 0.56 hingga 0.58, dan nilai tertinggi tercatat pada random state 10 dan 100, masing-masing mencapai 0.58. F1-score juga bervariasi antara 0.55 hingga 0.58, dengan performa terbaik pada random state 10 dan 100. Sementara itu, akurasi pada pengujian ini berkisar antara 0.56 hingga 0.59, dengan random state 10 dan 100 mencatatkan nilai akurasi tertinggi sebesar 0.59.

Tabel 3 Hasil Pengujian KNN 70% Data Latihan dan Data pengujian 30%

Random State	Precision	Recall	F1-score	Akurasi	Waktu Pelatihan (detik)
1	0.58	0.56	0.56	0.56	0.0012
10	0.58	0.58	0.57	0.58	0.0000
100	0.55	0.55	0.55	0.55	0.0000
1000	0.57	0.57	0.57	0.56	0.0000
10000	0.58	0.58	0.58	0.58	0.0041

Pembagian dataset 70:30 pada Tabel 3 menunjukkan hasilnya sedikit lebih rendah dibandingkan dengan pembagian 80:20, namun masih menunjukkan kinerja yang relatif stabil. Precision dan recall berada dalam kisaran 0.55 hingga 0.58, dengan nilai terbaik tercatat pada random state 10 dan 10000, yang mencapai 0.58. Begitu juga dengan F1-score, yang berkisar antara 0.55 hingga 0.58, dengan performa tertinggi pada random state 10 dan 10000. Akurasi pada pengujian ini berada di kisaran 0.55 hingga 0.58, dengan performa terbaik mencapai 0.58 pada random state 10 dan 10000.

3.4 Pengujian *Convolutional Neural Network (CNN)*

Pengujian *Convolutional Neural Network (CNN)* dengan ResNet152V2 dilakukan dengan berbagai jumlah epoch (10, 20, 30, 40, dan 50). Tujuan dari eksperimen adalah untuk mengevaluasi pengaruh jumlah epoch terhadap kinerja model dalam mengklasifikasikan kondisi permukaan jalan. Diharapkan kemampuan model untuk mendeteksi fitur gambar dengan lebih baik dapat ditingkatkan dengan meningkatkan jumlah epoch.

Konfigurasi CNN dengan pembagian data 80% data latihan dan 20% data pengujian ditunjukkan dalam Tabel 4 dan konfigurasi CNN dengan pembagian data 70% data latihan dan 30% data pengujian ditunjukkan di Tabel 5. Pembagian 80:20 menghasilkan jumlah 38 untuk setiap langkah epoch, sementara pembagian 70:30 menghasilkan jumlah 33 untuk setiap langkah epoch, memberikan gambaran tentang perbedaan dalam distribusi data antara kedua pengaturan tersebut.

Tabel 4 Hasil Pengujian CNN 80% Data Latihan dan Data pengujian 20%

Epoch	Precision	Recall	F1-score	Akurasi	Waktu pelatihan (detik)
10	0.94	0.93	0.93	0.93	962
20	0.94	0.94	0.94	0.94	1547
30	0.94	0.93	0.93	0.93	1897
40	0.96	0.96	0.96	0.96	2705
50	0.94	0.93	0.93	0.93	3054

Tabel 4 menunjukkan hasil pengujian model CNN dengan pembagian dataset 80% untuk latihan dan 20% untuk pengujian. Hasil pengujian menunjukkan seiring bertambahnya epoch, nilai Precision, Recall, F1-score, dan akurasi cenderung stabil pada angka yang tinggi, dengan nilai tertinggi dicapai pada epoch ke-40 yakni 0.96 untuk Precision, Recall, F1-score, dan akurasi, meskipun waktu pelatihan juga semakin lama.

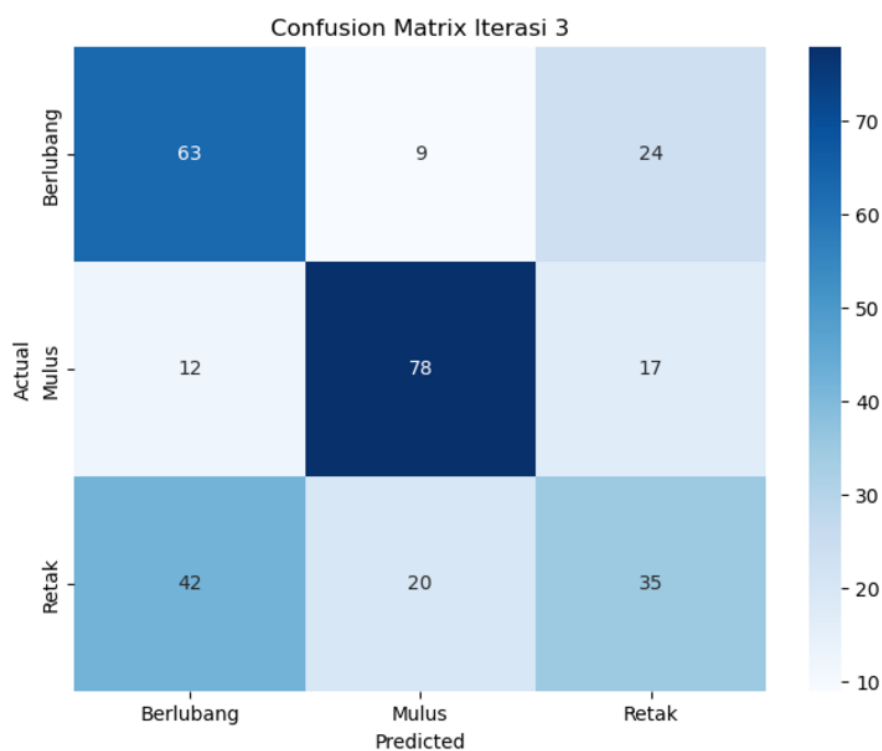
Tabel 5 Hasil Pengujian CNN 70% Data Latihan dan Data pengujian 30%

Epoch	Precision	Recall	F1-score	Akurasi	Waktu pelatihan (detik)
10	0.91	0.91	0.91	0.91	633
20	0.94	0.94	0.94	0.94	1183
30	0.93	0.93	0.93	0.93	1870
40	0.94	0.94	0.94	0.94	2733
50	0.93	0.93	0.93	0.93	3276

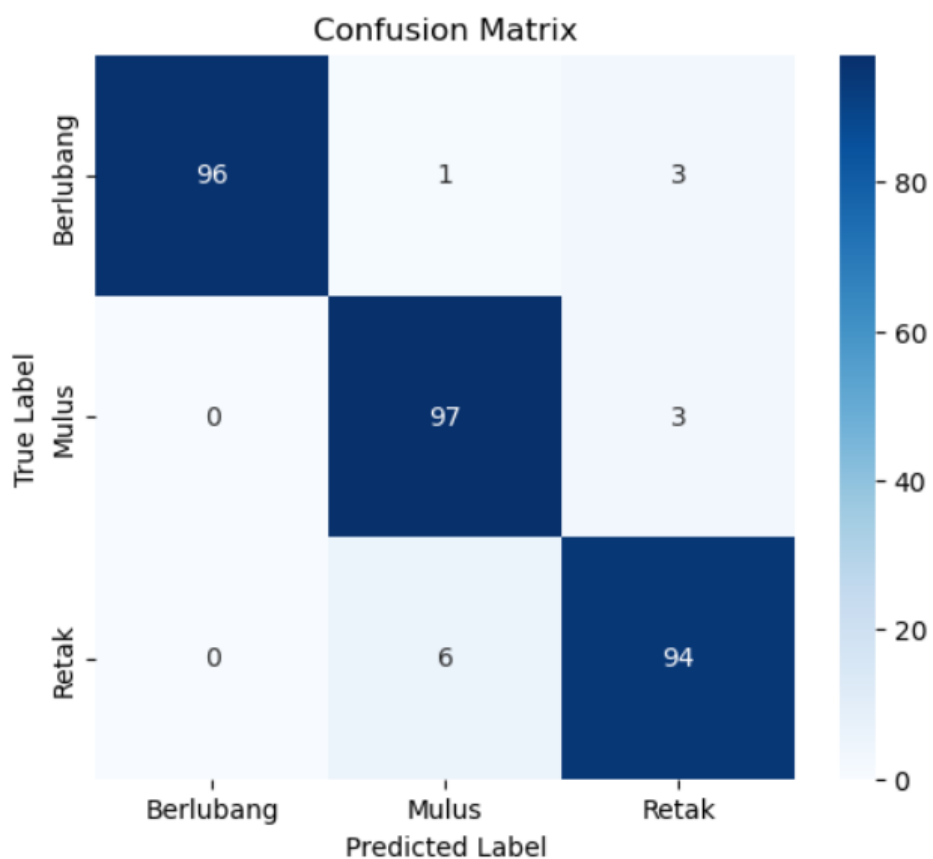
Tabel 5 menunjukkan hasil pengujian dengan 70% untuk latihan dan 30% untuk pengujian. Meskipun hasil Precision, Recall, F1-score, dan akurasi sedikit lebih rendah dibandingkan dengan pembagian 80%:20%, model ini tetap menunjukkan performa yang cukup baik dengan nilai tertinggi pada epoch ke-20 yakni 0.94 untuk Precision, Recall, F1-score, dan akurasi. Waktu pelatihan juga lebih cepat dibandingkan dengan pembagian 80%:20%, terutama pada epoch pertama yang memerlukan waktu hanya 633 detik.

3.5 Hasil Eksperimen

Eksperimen ini mengevaluasi perbandingan dua metode klasifikasi yang digunakan untuk mengklasifikasikan kondisi permukaan jalan, yaitu K-Nearest Neighbors (KNN) dan Convolutional Neural Network (CNN). Confusion matrix digunakan untuk menampilkan performa klasifikasi dari kedua model, yang memvisualisasikan prediksi benar dan salah untuk masing-masing kelas yaitu berlubang, mulus, dan retak.



Gambar 4 Confusion Matrix KNN Pembagian 80% Data Latihan dan 20% Data pengujian



Gambar 5 Confusion Matrix CNN Pembagian 80% Data Latihan dan 20% Data pengujian

Gambar 4 menunjukkan confusion matrix dari model K-Nearest Neighbors (KNN) dengan pembagian data latih 80% dan data pengujian 20%. Model ini memiliki akurasi rendah, terutama pada data dengan label retak yang sering salah diprediksi sebagai berlubang menunjukkan keterbatasan KNN dalam menangani klasifikasi berbasis citra kompleks. Sebaliknya Gambar 5 menampilkan confusion matrix dari model Convolutional Neural Network (CNN) yang memberikan performa terbaik dengan pembagian data yang sama. CNN mampu mengklasifikasikan data dengan tingkat akurasi tinggi, dengan prediksi benar dominan pada kategori berlubang (96), mulus (97), dan retak (94). Hal ini menunjukkan keunggulan CNN dalam menangkap pola visual kompleks, menjadikannya metode yang lebih andal dibandingkan KNN.

Tabel 6 Rata rata Hasil Pengujian

Algoritma	Data Latih	Data pengujian	Rata-rata Precision	Rata-rata Recall	Rata-rata F1-score	Rata-rata Akurasi	Rata-rata Waktu pelatihan (detik)
KNN ($k = 3$)	80	20	0.574	0.572	0.568	0.572	0.00386
KNN ($k = 3$)	70	30	0.572	0.568	0.566	0.566	0.00106
CNN (ResNet152V2)	80	20	0.944	0.938	0.938	0.938	2033
CNN (ResNet152V2)	70	30	0.93	0.93	0.93	0.93	1939

Tabel 6 hasil pengujian menunjukkan bahwa model CNN dengan model ResNet152V2 jauh lebih baik dibandingkan dengan KNN. CNN menghasilkan rata-rata Precision, Recall, F1-score, dan akurasi masing masing sebesar 0.944, 0.938, 0.938, dan 0.938 dengan waktu pelatihan rata-rata 2033 untuk 80% latihan dan 20% uji. Sebaliknya, KNN dengan $k = 3$ hanya untuk pembagian dataset 80% latihan dan 20% uji hanya memperoleh nilai rata-rata Precision, Recall, F1-score, dan akurasi masing masing sebesar 0.574, 0.572, 0.568, dan 0.572, dengan waktu pelatihan yang sangat cepat yaitu hanya 0.00386 detik. Demikian pula, pada pembagian dataset 70% latih dan 30% uji, CNN tetap menunjukkan performa yang lebih unggul dengan rata-rata Precision, Recall, F1-score, dan akurasi 0.93, dibandingkan KNN yang hanya mencapai 0.572 untuk Precision dan 0.566 untuk F1-score. Meskipun KNN memerlukan waktu pelatihan yang lebih singkat, CNN menunjukkan keunggulan yang signifikan dalam akurasi dan kinerja klasifikasinya pada pembagian dataset 80% latihan dan 20% pengujian. Pada pembagian data ini, menghasilkan nilai rata-rata Precision sebesar 0.944, Recall sebesar 0.938, F1-score sebesar 0.938, dan akurasi sebesar 0.938.

4. KESIMPULAN

Eksperimen ini menunjukkan bahwa Convolutional Neural Network (CNN) memiliki akurasi sebesar 93,8%, sedangkan K-Nearest Neighbors (KNN) memiliki akurasi 57,8%. CNN menunjukkan prediksi yang lebih akurat pada semua kelas terutama pada kondisi retak dengan lebih sedikit kesalahan klasifikasi dibandingkan KNN. Prediksi CNN konsisten mengenali pola visual kompleks, sementara KNN cenderung menghasilkan kesalahan lebih tinggi terutama pada kelas retak dan berlubang.

Hasil ini menegaskan bahwa CNN lebih unggul untuk klasifikasi kondisi permukaan jalan karena kemampuannya dalam ekstraksi fitur otomatis, meskipun membutuhkan waktu pelatihan yang lebih lama. Sebaliknya, KNN lebih sederhana namun kurang efektif dalam menangani pola yang kompleks.

Pengembangan selanjutnya dapat dilakukan dengan menggunakan arsitektur CNN yang lebih canggih untuk meningkatkan akurasi dan efisiensi model. Selain itu, augmentasi data dapat digunakan untuk meningkatkan variasi data latih agar model lebih andal. Pendekatan hybrid, seperti menggabungkan CNN untuk ekstraksi fitur dan model KNN atau SVM sebagai

pengklasifikasi juga dapat menjadi solusi untuk mengoptimalkan kelebihan kedua metode. Evaluasi pada data jalan yang lebih beragam juga diperlukan untuk memastikan generalisasi model di kondisi nyata. Dengan pengembangan ini, sistem klasifikasi kondisi permukaan jalan dapat menjadi lebih akurat dan efektif untuk diterapkan dalam sistem pemantauan jalan otomatis.

DAFTAR PUSTAKA

- [1] R. L. Gaho, I. T. Ali, and E. Prakasa, "Klasifikasi Kualitas Permukaan Jalan Raya Menggunakan Metode CNN Berbasis Arsitektur Xception," *Jurnal Inovtek Polbeng Seri Informatika*, vol. 9, no. 1, 2024.
- [2] E. Setiadi and A. Wibowo, "Klasifikasi dan deteksi keretakan pada trotoar menggunakan metode convolutional neural network," 2023.
- [3] I. Maylani, V. W. Ambarwati, B. Wasykuru, A. Alqaroni, and F. T. B. K. Wati, "Grey Level Co-Occurrence Matrix (GLCM) & Hybrid Klasifikasi untuk Mendeteksi Kerusakan Jalan Aspal," in *Proc. Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika (SNESTIK)*, vol. 1, no. 1, pp. 225–231, Apr. 2023.
- [4] S. Ulya, M. A. Soeleman, and F. Budiman, "Optimasi Parameter K Pada Algoritma K-NN Untuk Klasifikasi Prioritas Bantuan Pembangunan Desa," *Techno. Com*, vol. 20, no. 1, 2021.
- [5] M. H. Fadlun, M. Martanto, and U. Hayati, "Klasifikasi Tumor Otak menggunakan Convolutional Neural Network dan Transfer Learning," *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 6, no. 1, pp. 289–295, 2024.
- [6] R. L. Gaho, I. T. Ali, and E. Prakasa, "Klasifikasi Kualitas Permukaan Jalan Raya Menggunakan Metode CNN Berbasis Arsitektur Xception," *Jurnal Inovtek Polbeng Seri Informatika*, vol. 9, no. 1, 2024.
- [7] M. R. W. Budiarto and F. Utaminingrum, "Sistem Klasifikasi Permukaan Jalan dan Penghindaran Jalan Berlubang pada Kursi Roda Pintar dengan Metode MobileNetV3-Small," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 5, pp. 2249–2254, 2023.
- [8] M. Muchtar, M. N. Sutoyo, A. Paliling, S. Sunyanti, and J. N. Iin, "Penerapan Analisis Berbasis Fraktal dalam Klasifikasi Citra Retakan pada Permukaan Jembatan Beton," *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, vol. 9, no. 1, pp. 21–29, 2024.
- [9] U. Shodiq, M. Maimunah, and M. H. Avizenna, "Klasifikasi Jalan Rusak Menggunakan Transfer Learning Arsitektur VGG16," *JOISIE (Journal of Information Systems and Informatics Engineering)*, vol. 8, no. 1, pp. 75–85, 2024.
- [10] R. A. Saputra, S. Wasiyanti, D. F. Saefudin, A. Supriyatna, and A. Wibowo, "Rice leaf disease image classifications using KNN based on GLCM feature extraction," in *Journal of Physics: Conference Series*, vol. 1641, no. 1, p. 012080, Nov. 2020.
- [11] S. R. Cholil, T. Handayani, R. Prathivi, and T. Ardianita, "Implementasi algoritma klasifikasi k-nearest neighbor (KNN) untuk klasifikasi seleksi penerima beasiswa," *IJCIT (Indonesian Journal on Computer and Information Technology)*, vol. 6, no. 2, pp. 118–127, 2021.
- [12] D. Kurnia and A. T. Wibowo, "Klasifikasi Spesies Tanaman Kaktus Grafting Berdasarkan Citra Scion Menggunakan Metode Convolutional Neural Network (CNN)," *eProceedings of Engineering*, vol. 8, no. 4, 2021.
- [13] K. Azmi, S. Defit, and S. Sumijan, "Implementasi convolutional neural network (CNN) untuk klasifikasi batik tanah liat Sumatera Barat," *Jurnal Unitek*, vol. 16, no. 1, pp. 28–40, 2023.
- [14] M. Khairunnas, "Penggunaan Metode Deep Learning EfficientNetB1 untuk Mengenali Sampah Organik dan Sampah Anorganik," M.S. thesis, Universitas Islam Negeri Sultan Syarif Kasim Riau, 2021.

- [15] S. Sikandar, R. Mahum, and A. Alsalman, "A novel hybrid approach for a content-based image retrieval using feature fusion," *Applied Sciences*, vol. 13, no. 7, p. 4581, 2023.
- [16] D. Aditya, "Klasifikasi Motif Songket Indonesia Menggunakan Metode Convolutional Neural Network (CNN)," M.S. thesis, Universitas Islam Negeri Sultan Syarif Kasim Riau, 2024.
- [17] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, "Implementasi metode convolutional neural network menggunakan arsitektur LeNet-5 untuk pengenalan doodle," *Jurnal Algoritme*, vol. 1, no. 1, pp. 45–56, 2020.
- [18] D. R. R. Putra and R. A. Saputra, "Implementasi Convolutional Neural Network (CNN) untuk Mendeteksi Penggunaan Masker pada Gambar," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 3, 2023.
- [19] K. R. Wardani and L. Leonardi, "Klasifikasi Penyakit pada Daun Anggur menggunakan Metode Convolutional Neural Network," *Jurnal Tekno Insentif*, vol. 17, no. 2, pp. 112–126, 2023.
- [20] M. R. A. Nasution and M. Hayaty, "Perbandingan akurasi dan waktu proses algoritma K-NN dan SVM dalam analisis sentimen Twitter," *J. Inform*, vol. 6, no. 2, pp. 226–235, 2019.
- [21] Y. Sari, A. R. Baskara, P. B. Prakoso, and M. A. Rahman, "Penerapan Active Contour Model pada Pengolahan Citra untuk Deteksi Kerusakan Jalan," *Jurnal Jalan Jembatan*, vol. 38, no. 2, pp. 134–142, 2021.
- [22] B. P. Pratiwi, A. S. Handayani, and S. Sarjana, "Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi WSN Menggunakan Confusion Matrix," *Jurnal Informatika Upgris*, vol. 6, no. 2, 2020.
- [23] R. R. Adhitya, W. Witanti, and R. Yuniarti, "Perbandingan Metode CART dan Naïve Bayes untuk Klasifikasi Customer Churn," *INFOTECH journal*, vol. 9, no. 2, pp. 307–318, 2023.
- [24] R. R. Sani, Y. A. Pratiwi, S. Winarno, E. D. Udayanti, and F. Alzami, "Analisis Perbandingan Algoritma Naive Bayes Classifier dan Support Vector Machine untuk Klasifikasi Berita Hoax pada Berita Online Indonesia," *Jurnal Masyarakat Informatika*, vol. 13, no. 2, pp. 85–98, 2022.
- [25] T. L. Malau and T. Joseph, "Analisis Metode Logistik Regresi Ensemble untuk Klasifikasi dengan Pra-Pemrosesan Menggunakan Principal Component Analysis," *IJM: Indonesian Journal of Multidisciplinary*, vol. 1, no. 2, 2023.
- [26] Google Drive, "Dataset Permukaan Jalan." [Online]. Available: https://drive.google.com/drive/folders/1kyRJT3bmx2oKr5Q1bG_XIAe1dab30O3v?usp=drive_link. [Accessed: Dec. 4, 2024].