

**PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM  
KLASIFIKASI SAMPAH BERBASIS CITRA**

**SKRIPSI**

Oleh :  
**ADILA QURROTA A'YUN**  
**NIM. 210605110105**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

**PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM  
KLASIFIKASI SAMPAH BERBASIS CITRA**

**SKRIPSI**

Diajukan Kepada:  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :  
**ADILA QURROTA A'YUN**  
NIM. 210605110105

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

**HALAMAN PERSETUJUAN**

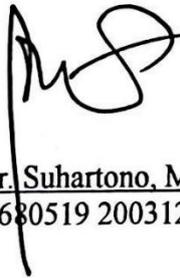
**PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM  
KLASIFIKASI SAMPAH BERBASIS CITRA**

**SKRIPSI**

Oleh :  
**ADILA QURROTA A'YUN**  
**NIM. 210605110105**

Telah Diperiksa dan Disetujui untuk Diuji:  
Tanggal: 17 Maret 2025

Pembimbing I,



Prof. Dr. Suhartono, M.Kom  
NIP. 19680519 200312 1 001

Pembimbing II,



Tri Mukti Lestari, M.Kom  
NIP. 19911108 202012 2 005

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Ir. Fachrul Kurniawan, M.MT., IPU  
NIP. 19771020 200912 1 001

## HALAMAN PENGESAHAN

### PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM KLASIFIKASI SAMPAH BERBASIS CITRA

#### SKRIPSI

Oleh :

**ADILA QURROTA A'YUN**  
**NIM. 210605110105**

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima sebagai Salah Satu Persyaratan  
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal: 06 Mei 2025

#### Susunan Dewan Penguji

Ketua Penguji	: <u>Fajar Rohman Hariri, M.Kom</u> NIP. 19890515 201801 1 001	(  )
Anggota Penguji I	: <u>Roro Inda Melani, MT., M.Sc</u> NIP. 19780925 200501 2 008	(  )
Anggota Penguji II	: <u>Prof. Dr. Suhartono, M.Kom</u> NIP. 19680519 200312 1 001	(  )
Anggota Penguji III	: <u>Tri Mukti Lestari, M.Kom</u> NIP. 19911108 202012 2 005	(  )

Mengetahui dan Mengesahkan,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Ir. Fachrul Kurniawan, M.MT., IPU  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Adila Qurrota A'yun  
NIM : 210605110105  
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika  
Judul Skripsi : Penerapan *Convolutional Neural Network* dalam  
Klasifikasi Sampah Berbasis Citra

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 06 Mei 2025

Yang membuat pernyataan,



Adila Qurrota A'yun

NIM. 210605110105

**MOTTO**

*“Everybody is a genius”*

## **HALAMAN PERSEMBAHAN**

Dengan penuh rasa syukur, penulis mempersembahkan karya ini kepada:

Ayah dan Mama

Atas doa, dukungan, dan kasih sayang yang tidak pernah henti.

Kakak, Adik, dan Kerabat

Atas dukungan dan doa yang selalu menyertai.

Dosen

Atas ilmu dan bimbingannya selama masa studi.

Teman

Yang selalu kebersamai dalam setiap perjalanan studi.

Diri Sendiri

Yang telah berusaha dan bertahan hingga di titik ini.

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penerapan *Convolutional Neural Network* dalam Klasifikasi Sampah Berbasis Citra” dengan baik.

Penyusunan skripsi ini tidak terlepas dari bimbingan, dukungan, serta doa dari berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Dr. H. M. Zainuddin, MA selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT., IPU selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. Suhartono, M.Kom selaku Dosen Pembimbing I dan Ibu Tri Mukti Lestari, M.Kom selaku Dosen Pembimbing II atas bimbingan, arahan, dan motivasi yang sangat berarti bagi penulis.
5. Bapak Fajar Rohman Hariri, M.Kom selaku Dosen Penguji I dan Ibu Roro Inda Melani, MT., M.Sc selaku Dosen Penguji II atas masukan dan saran yang membangun.
6. Ibu Khadijah Fahmi Hayati Holle, M.Kom dan Dr. Zainal Abidin, M.Kom selaku Dosen Wali atas arahan dan bimbingannya selama masa studi.
7. Segenap dosen dan staf akademik Program Studi Teknik Informatika atas ilmu yang diberikan selama masa studi.

8. Bapak Muhajir dan Ibu Eni Ulfatin selaku kedua orang tua penulis atas cinta, doa, dukungan tanpa henti; serta kakak, adik penulis yang selalu memberikan semangat.
9. Orang terdekat yang menjadi sumber semangat, sarana berdiskusi, dan bertukar pikiran; Pejuang S.Kom yang telah kebersamai penulis sejak awal perkuliahan; serta Bimbingan Prof Suhar yang selalu menemani dan membantu selama proses penyusunan skripsi.
10. Saudara Teknik Informatika khususnya ASTER '21 yang telah menjadi bagian dari perjalanan akademik serta senantiasa saling mendukung.
11. HIMATIF ENCODER khususnya Divisi Public Relation dan Komunitas GDSC Maliki '22 '23 atas pengalaman serta kesempatan untuk berproses.
12. Seluruh pihak yang tidak dapat penulis sebutkan satu per satu atas bantuannya selama masa studi berlangsung.
13. Diri sendiri atas setiap langkah kecil yang dijalani dengan usaha, kesabaran, dan doa hingga di titik ini.

Penulis menyadari bahwa penyusunan skripsi ini masih terdapat kekurangan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan untuk perbaikan di masa mendatang. Besar harapan penulis agar skripsi ini dapat memberikan manfaat serta wawasan bagi pembaca.

Malang, 30 April 2025

Penulis

## DAFTAR ISI

<b>HALAMAN PENGAJUAN</b> .....	<b>ii</b>
<b>HALAMAN PERSETUJUAN</b> .....	<b>iii</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN</b> .....	<b>v</b>
<b>MOTTO</b> .....	<b>vi</b>
<b>HALAMAN PERSEMBAHAN</b> .....	<b>vii</b>
<b>KATA PENGANTAR</b> .....	<b>viii</b>
<b>DAFTAR ISI</b> .....	<b>x</b>
<b>DAFTAR GAMBAR</b> .....	<b>xii</b>
<b>DAFTAR TABEL</b> .....	<b>xiii</b>
<b>ABSTRAK</b> .....	<b>xiv</b>
<b>ABSTRACT</b> .....	<b>xv</b>
<b>مستخلص البحث</b> .....	<b>xvi</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	5
1.3 Batasan Masalah .....	5
1.4 Tujuan Penelitian .....	6
1.5 Manfaat Penelitian .....	6
<b>BAB II STUDI PUSTAKA</b> .....	<b>7</b>
2.1 Penelitian Terkait .....	7
2.2 Sampah.....	12
2.3 <i>Convolutional Neural Network (CNN)</i> .....	13
<b>BAB III METODE PENELITIAN</b> .....	<b>16</b>
3.1 Desain Penelitian .....	16
3.2 Pengumpulan Data .....	17
3.3 Desain Sistem.....	18
3.3.1 <i>Input Citra</i> .....	19
3.3.2 <i>Preprocessing Data</i> .....	19
3.4 Implementasi Metode.....	22
3.4.1 <i>Convolutional Layer</i> .....	24
3.4.2 <i>Max Pooling Layer</i> .....	25
3.4.3 <i>Fully Connected Layer</i> .....	26
3.5 Skenario Pengujian .....	29
3.6 Evaluasi.....	31
<b>BAB IV HASIL DAN PEMBAHASAN</b> .....	<b>33</b>
4.1 Hasil Uji Coba.....	33
4.1.1 Skenario Uji 1A .....	33
4.1.2 Skenario Uji 1B.....	39
4.1.3 Skenario Uji 2A .....	40
4.1.4 Skenario Uji 2B.....	46
4.2 Pengujian <i>K-Fold Cross Validation</i> .....	49
4.2.1 Skenario Uji 1A .....	50

4.2.2 Skenario Uji 1B .....	51
4.2.3 Skenario Uji 2A .....	53
4.2.4 Skenario Uji 2B .....	54
4.3 Pembahasan.....	55
4.4 Integrasi Sains dan Islam .....	62
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>65</b>
5.1 Kesimpulan .....	65
5.2 Saran .....	65
<b>DAFTAR PUSTAKA</b>	

## DAFTAR GAMBAR

Gambar 2.1 Arsitektur <i>Convolutional Neural Network</i> (CNN).....	14
Gambar 3.1 Desain Penelitian.....	16
Gambar 3.2 Desain Sistem.....	18
Gambar 3.3 <i>Zoom</i> .....	21
Gambar 3.4 <i>Rotate</i> .....	21
Gambar 3.5 <i>Horizontal Flip</i> .....	22
Gambar 3.6 Arsitektur CNN .....	23
Gambar 3.7 Perhitungan Proses <i>Convolutional</i> .....	25
Gambar 3.8 Penerapan Proses <i>Max Pooling</i> .....	26
Gambar 3.9 Penerapan Proses <i>Flatten</i> .....	27
Gambar 4.1 Grafik <i>Training Accuracy</i> dan <i>Loss</i> Skenario Uji 1A .....	34
Gambar 4.2 <i>Confusion Matrix</i> Skenario Uji 1A <i>Threshold</i> 0.3 .....	34
Gambar 4.3 <i>Confusion Matrix</i> Skenario Uji 1A <i>Threshold</i> 0.4 .....	35
Gambar 4.4 <i>Confusion Matrix</i> Skenario Uji 1A <i>Threshold</i> 0.5 .....	36
Gambar 4.5 <i>Confusion Matrix</i> Skenario Uji 1A <i>Threshold</i> 0.6 .....	37
Gambar 4.6 <i>Confusion Matrix</i> Skenario Uji 1A <i>Threshold</i> 0.7 .....	38
Gambar 4.7 Grafik <i>Training Accuracy</i> dan <i>Loss</i> Skenario Uji 1B .....	39
Gambar 4.8 <i>Confusion Matrix</i> Skenario Uji 1B.....	40
Gambar 4.9 Grafik <i>Training Accuracy</i> dan <i>Loss</i> Skenario Uji 2A .....	41
Gambar 4.10 <i>Confusion Matrix</i> Skenario Uji 2A <i>Threshold</i> 0.3 .....	42
Gambar 4.11 <i>Confusion Matrix</i> Skenario Uji 2A <i>Threshold</i> 0.4 .....	43
Gambar 4.12 <i>Confusion Matrix</i> Skenario Uji 2A <i>Threshold</i> 0.5 .....	44
Gambar 4.13 <i>Confusion Matrix</i> Skenario Uji 2A <i>Threshold</i> 0.6 .....	45
Gambar 4.14 <i>Confusion Matrix</i> Skenario Uji 2A <i>Threshold</i> 0.7 .....	46
Gambar 4.15 Grafik <i>Training Accuracy</i> dan <i>Loss</i> Skenario Uji 2B .....	47
Gambar 4.16 <i>Confusion Matrix</i> Skenario Uji 2B .....	48
Gambar 4.17 Grafik Akurasi Setiap <i>Fold</i> Skenario Uji 1A.....	51
Gambar 4.18 Grafik Akurasi Setiap <i>Fold</i> Skenario Uji 1B .....	52
Gambar 4.19 Grafik Akurasi Setiap <i>Fold</i> Skenario Uji 2A.....	53
Gambar 4.20 Grafik Akurasi Setiap <i>Fold</i> Skenario Uji 2B .....	55
Gambar 4.21 Akurasi dan Metrik Evaluasi <i>Threshold</i> Skenario 1A .....	57
Gambar 4.22 Akurasi dan Metrik Evaluasi <i>Threshold</i> Skenario 2A .....	58
Gambar 4.23 Akurasi dan Metrik Evaluasi Antar Skenario .....	59

## DAFTAR TABEL

Tabel 2.1 Penelitian Terkait .....	10
Tabel 3.1 Contoh Sampel Citra Sampah <i>Paper</i> dan <i>Plastic</i> .....	17
Tabel 3.2 Skenario Augmentasi Citra .....	29
Tabel 3.3 Skenario Metode Klasifikasi .....	30
Tabel 3.4 Kombinasi Skenario Pengujian .....	30
Tabel 3.5 Nilai Parameter .....	30
Tabel 4.1 Metrik Evaluasi Skenario Uji 1A <i>Threshold</i> 0.3 .....	35
Tabel 4.2 Metrik Evaluasi Skenario Uji 1A <i>Threshold</i> 0.4 .....	36
Tabel 4.3 Metrik Evaluasi Skenario Uji 1A <i>Threshold</i> 0.5 .....	37
Tabel 4.4 Metrik Evaluasi Skenario Uji 1A <i>Threshold</i> 0.6 .....	38
Tabel 4.5 Metrik Evaluasi Skenario Uji 1A <i>Threshold</i> 0.7 .....	39
Tabel 4.6 Metrik Evaluasi Skenario Uji 1B .....	40
Tabel 4.7 Metrik Evaluasi Skenario Uji 2A <i>Threshold</i> 0.3 .....	42
Tabel 4.8 Metrik Evaluasi Skenario Uji 2A <i>Threshold</i> 0.4 .....	43
Tabel 4.9 Metrik Evaluasi Skenario Uji 2A <i>Threshold</i> 0.5 .....	44
Tabel 4.10 Metrik Evaluasi Skenario Uji 2A <i>Threshold</i> 0.6 .....	45
Tabel 4.11 Metrik Evaluasi Skenario Uji 2A <i>Threshold</i> 0.7 .....	46
Tabel 4.12 Metrik Evaluasi Skenario Uji 2B .....	48
Tabel 4.13 Contoh Hasil Klasifikasi Citra .....	49
Tabel 4.14 Hasil Akurasi Setiap <i>Fold</i> Skenario Uji 1A .....	50
Tabel 4.15 Hasil Akurasi Setiap <i>Fold</i> Skenario Uji 1B .....	52
Tabel 4.16 Hasil Akurasi Setiap <i>Fold</i> Skenario Uji 2A .....	53
Tabel 4.17 Hasil Akurasi Setiap <i>Fold</i> Skenario Uji 2B .....	54
Tabel 4.18 Rata-Rata Akurasi Pengujian <i>K-Fold</i> Antar Skenario .....	59

## ABSTRAK

A'yun, Adila Qurrota. 2025. **Penerapan *Convolutional Neural Network* dalam Klasifikasi Sampah Berbasis Citra**. Skripsi. Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, M.Kom (II) Tri Mukti Lestari, M.Kom.

**Kata Kunci:** *Convolutional Neural Network, Klasifikasi Citra, Sampah*

Permasalahan pengelolaan sampah yang semakin kompleks, khususnya dalam proses pemilahan, menuntut adanya solusi berbasis teknologi yang efisien dan akurat. Penelitian ini bertujuan untuk menerapkan metode *Convolutional Neural Network* (CNN) dalam klasifikasi sampah berbasis citra, dengan fokus pada dua kelas yaitu *paper* dan *plastic*. Dataset yang digunakan sebanyak 2000 citra dengan proporsi 80% untuk pelatihan dan 20% untuk pengujian. Penelitian ini menguji empat skenario kombinasi augmentasi citra dan metode klasifikasi yaitu *threshold* dan *one-hot encoding*, serta mengevaluasi kinerja model menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil terbaik diperoleh pada skenario tanpa augmentasi citra dengan metode klasifikasi *one-hot encoding* dengan akurasi 86%, *precision* 86%, *recall* 86%, dan *F1-score* 88,5%. Temuan ini menunjukkan bahwa penerapan CNN dapat meningkatkan efektivitas klasifikasi sampah berbasis citra dan mendukung upaya daur ulang melalui sistem pemilahan sampah yang lebih cerdas dan otomatis.

## ABSTRACT

A'yun, Adila Qurrota. 2025. **Implementation of Convolutional Neural Network in Image-Based Waste Classification.** Undergraduate Thesis. Informatics Engineering Study Program, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Supervisor: (I) Prof. Dr. Suhartono, M.Kom (II) Tri Mukti Lestari, M.Kom.

**Keywords:** Convolutional Neural Network, Image Classification, Waste

The increasingly complex issue of waste management, particularly in the sorting process, demands efficient and accurate technology-based solution. This study aims to implement the Convolutional Neural Network (CNN) method for image-based waste classification, focusing on two classes paper and plastic. The dataset used consists of 2,000 images, with an 80% proportion for training and 20% for testing. This study tested four scenarios combining image augmentation and classification methods, namely threshold and one-hot encoding, and evaluated model performance using accuracy, precision, recall, and F1-score metrics. The best results were obtained in the scenario without image augmentation using the one-hot encoding classification method, with an accuracy of 86%, precision of 86%, recall of 86%, and F1-score of 88,5%. These findings indicate that implementation of CNN can enhance the effectiveness of image-based waste classification and support recycling efforts through a smarter and more automated sorting system.

## مستخلص البحث

أعين، عادل قرة. 2025. تطبيق الشبكة العصبية الالتفافية في تصنيف النفايات المعتمدة على الصور. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: أ. د. سهارتونو، ماجستير. المشرف الثاني: تري موكتي ليستاري، ماجستير.

**الكلمات الرئيسية:** الشبكة العصبية التلافيفية، تصنيف الصور، النفايات

تتطلب مشكلة إدارة النفايات المتزايدة التعقيد، خاصة في عملية الفرز، وجود حلول تقنية فعالة ودقيقة. تهدف هذه الدراسة إلى تطبيق طريقة الشبكة العصبية التلافيفية (CNN) في تصنيف النفايات القائم على الصور، مع التركيز على فئتين هما الورق والبلاستيك. تم استخدام 2000 صورة في مجموعة البيانات، بنسبة 80% للتدريب و 20% للاختبار. اختبرت هذه الدراسة أربعة سيناريوهات من مزيج توسيع الصور وطرق التصنيف، وهي *threshold* و *one-hot encoding*، كما قيمت أداء النموذج باستخدام مقاييس الدقة والضبط والاسترجاع ودرجة F1. أفضل النتائج تم الحصول عليها في السيناريو بدون زيادة الصور باستخدام طريقة التصنيف *one-hot encoding* بدقة 86%، ودقة 86%، واسترجاع 86%، ودرجة F1 88.5%. تشير هذه النتائج إلى أن تطبيق CNN يمكن أن يحسن فعالية تصنيف النفايات القائم على الصور ويدعم جهود إعادة التدوير من خلال نظام فرز النفايات الأكثر ذكاءً وتلقائيةً.

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pengelolaan sampah telah menjadi isu lingkungan yang mendesak di seluruh dunia (Deslanda et al., 2023). Peningkatan jumlah sampah akibat pertumbuhan populasi dan urbanisasi yang cepat tidak hanya menyebabkan pencemaran tanah, air, dan udara, tetapi juga menimbulkan berbagai masalah kesehatan, seperti polusi udara dan suara (Prabhu, 2022). Menurut data dari World Bank Group (2022), produksi sampah global pada tahun 2020 diperkirakan mencapai 2,24 miliar ton, setara dengan 0,79 kilogram per orang per hari. Jika tidak dikelola dengan baik, tumpukan sampah akan semakin mempercepat kerusakan ekosistem dan membahayakan kesehatan masyarakat. Dengan pertumbuhan populasi dan urbanisasi yang pesat, diperkirakan produksi sampah akan meningkat sebesar 73% pada tahun 2050, mencapai 3,88 miliar ton.

Berbagai negara dihadapkan pada tantangan besar dalam mengelola sampah. Sistem pengelolaan sampah yang tidak memadai dapat memperburuk dampak lingkungan, seperti peningkatan emisi gas rumah kaca (Lasaiba, 2024). Di negara-negara berkembang, termasuk Indonesia, penanganan sampah masih menghadapi berbagai hambatan, mulai dari infrastruktur yang tidak memadai hingga kesadaran masyarakat yang rendah (Julia Lingga et al., 2024). Menurut Kementerian Lingkungan Hidup dan Kehutanan (2023), Indonesia menghasilkan sekitar 38,79

juta ton sampah setiap tahunnya, namun hanya sekitar 48,45% yang dapat dikelola secara optimal.

Salah satu tantangan utama dalam pengelolaan sampah di Indonesia adalah rendahnya efektivitas sistem pemilahan sampah. Data dari Kementerian Lingkungan Hidup dan Kehutanan (KLHK) menunjukkan bahwa lebih dari 60% sampah berakhir di Tempat Pembuangan Akhir (TPA), di mana pengelolaannya belum memadai (Sumartiningtyas, 2020). TPA yang penuh dan sistem pemilahan manual menyebabkan peningkatan risiko pencemaran lingkungan dan ancaman kesehatan masyarakat. Selain itu, jenis sampah seperti plastik yang sulit terurai menambah tantangan dalam pengelolaannya. Menurut Sustainable Waste Indonesia (SWI), sampah plastik menguasai sekitar 5% dari total sampah nasional, yang setara dengan 3,2 juta ton setiap tahunnya (Waluyo, 2023). Plastik yang tidak dikelola akan mencemari lautan dan masuk ke rantai makanan melalui mikroplastik (Aqilla et al., 2023).

Upaya dalam mencari solusi atas permasalahan pengelolaan sampah merupakan wujud tanggung jawab manusia dalam menjaga kelestarian bumi dan lingkungan yang telah Allah ciptakan. Dalam ajaran Islam, merawat alam merupakan amanah yang dipercayakan kepada manusia sebagai khalifah di muka bumi. Salah satu cara untuk menjalankan amanah tersebut adalah dengan mendukung proses daur ulang sampah. Hal ini sejalan dengan firman Allah dalam surah Al-A'raf ayat 56 yang berbunyi:

وَلَا تُفْسِدُوا فِي الْأَرْضِ بَعْدَ إِصْلَاحِهَا وَادْعُوهُ حَوْفًا وَقَطْمًا إِنَّ رَحْمَتَ اللَّهِ قَرِيبٌ مِّنَ الْمُحْسِنِينَ

*“Janganlah kamu berbuat kerusakan di bumi setelah diatur dengan baik. Berdoalah kepada-Nya dengan rasa takut dan penuh harap. Sesungguhnya rahmat Allah sangat dekat dengan orang-orang yang berbuat baik.” (QS. Al-A’raf: 56).*

Menurut penjelasan dalam Tafsir Al-Muyassar yang diterbitkan oleh Kementrian Agama Saudi Arabia, manusia dilarang melakukan berbagai bentuk kerusakan di bumi setelah Allah memperbaikinya melalui pengutusan para rasul dan pengajaran amal ketaatan kepada-Nya untuk memakmurkan bumi. Allah memerintahkan agar kita berdoa kepada-Nya dengan penuh keikhlasan, disertai rasa takut akan azab-Nya serta mengharap pahala-Nya. Sesungguhnya, rahmat Allah sangat dekat dengan orang-orang yang senantiasa berbuat kebaikan (Al-Qarni, 2016).

Mengingat volume sampah yang terus meningkat, metode pemilahan manual tidak lagi efisien dan meningkatkan risiko kesalahan. Oleh karena itu, penerapan teknologi pemilahan sampah secara otomatis penting untuk mengatasi masalah ini. Salah satu teknologi yang efektif adalah *Convolutional Neural Network (CNN)*, sebuah algoritma *deep learning* yang secara khusus dikembangkan untuk memproses data berupa citra (Yan, 2024). CNN dapat mengenali berbagai jenis sampah berdasarkan citra secara otomatis, sehingga dapat meningkatkan efisiensi dan meminimalkan kesalahan dalam proses pemilahan.

Penelitian yang dilakukan oleh Dacipta & Putra (2022) mengeksplorasi penerapan metode *Convolutional Neural Network (CNN)* dengan arsitektur VGGNet yang digunakan untuk melakukan klasifikasi limbah. Dataset diambil dari *platform* Kaggle dan Google Drive dengan total 8371 citra yang terdiri dari 9 jenis limbah, yaitu baterai, pakaian, e-limbah, kaca, bola lampu, logam, organik, kertas,

dan plastik. Penelitian ini menerapkan *K-Fold Cross Validation* guna memastikan keandalan model pada proses *training* dan *testing*. Hasil dari penelitian ini mencapai akurasi tertinggi 69,77% pada proses *training* setelah 28 *epoch*, dan akurasi 64,45% pada proses *testing*. Meskipun penelitian ini berhasil menunjukkan efektivitas CNN dalam klasifikasi limbah, namun penelitian ini masih memiliki beberapa keterbatasan yaitu akurasi masih tergolong rendah, tidak adanya augmentasi data untuk meningkatkan keragaman dataset, serta evaluasi model yang hanya berfokus pada akurasi tanpa mempertimbangkan evaluasi lainnya yang mencakup *precision*, *recall*, dan *F1-score*.

Selain digunakan untuk klasifikasi limbah, metode *Convolutional Neural Network* (CNN) juga telah diterapkan pada klasifikasi sampah botol dengan membandingkan kinerja arsitektur VGG16 dan ResNet34 oleh Hutamaputra et al., (2022). Dataset yang digunakan berasal dari GitHub dan Kaggle sebanyak 4449 gambar dengan 2 kelas yaitu botol plastik dan botol bukan plastik. Data tersebut dibagi menjadi 90% untuk *training* dan 10% untuk validasi. Hasil penelitian ini menunjukkan bahwa arsitektur VGG16 mencapai akurasi 96,39%, sedangkan ResNet34 mencapai akurasi 91,37%. Namun, penelitian ini memiliki beberapa keterbatasan, salah satunya tidak disertakan modifikasi arsitektur untuk meningkatkan performa model, tidak diterapkannya teknik augmentasi data untuk mencegah *overfitting*, serta tidak dijelaskannya ukuran *input* citra dan *kernel* pada arsitektur ResNet34 yang dapat memengaruhi efisiensi model.

Penelitian ini bertujuan untuk menerapkan metode CNN dalam mengklasifikasikan sampah berbasis citra guna meningkatkan efektivitas

pengelolaan sampah. Dengan meningkatnya volume sampah, terutama plastik yang sulit terurai, diperlukan solusi inovatif yang mampu mempercepat proses daur ulang sekaligus mengurangi dampak negatif terhadap lingkungan. Implementasi metode CNN diharapkan dapat meningkatkan ketepatan dalam pemilahan jenis sampah, sehingga mendukung proses daur ulang yang lebih efektif dan berkelanjutan.

Pentingnya penelitian ini terletak pada kontribusi terhadap sistem pengelolaan sampah yang lebih cerdas dan otomatis. Dengan memanfaatkan kemampuan CNN, penelitian ini berpotensi memberikan solusi teknologi yang relevan bagi tantangan global dalam pemilahan sampah. Berdasarkan latar belakang di atas, peneliti tertarik untuk melakukan penelitian terkait penerapan CNN dalam klasifikasi sampah, yang diharapkan dapat menjadi solusi dalam menghadapi tantangan pengelolaan sampah.

## **1.2 Rumusan Masalah**

Bagaimana performa metode *Convolutional Neural Network* (CNN) dalam mengklasifikasikan jenis sampah berbasis citra?

## **1.3 Batasan Masalah**

Guna memperoleh hasil yang optimal, penelitian ini menetapkan beberapa batasan, diantaranya:

- a. Dataset yang digunakan merupakan data citra yang diperoleh dari *platform* Kaggle yaitu “*Real Waste Image Classification*” dan “*Trash Images*” dengan format JPG.

- b. Jenis sampah yang diklasifikasikan adalah sampah anorganik yang dapat didaur ulang berupa *paper* (kertas) dan *plastic* (plastik).
- c. Penelitian ini berfokus pada pengujian dengan teknik augmentasi citra serta penggunaan klasifikasi *threshold* dan *one-hot encoding*.

#### **1.4 Tujuan Penelitian**

Mengevaluasi performa metode *Convolutional Neural Network* (CNN) dalam mengklasifikasikan jenis sampah berbasis citra.

#### **1.5 Manfaat Penelitian**

Penelitian ini diharapkan dapat memberikan solusi inovatif untuk mempermudah proses pemilahan sampah, sehingga mendukung upaya daur ulang dan pengelolaan sampah yang lebih efektif.

## BAB II

### STUDI PUSTAKA

#### 2.1 Penelitian Terkait

Penelitian terkait digunakan untuk mengkaji penelitian sebelumnya yang dijadikan acuan oleh peneliti untuk memahami topik penelitian dan membandingkannya dengan penelitian yang sedang dilakukan.

Leonardo et al., (2020) melakukan penelitian tentang klasifikasi sampah daur ulang dengan memanfaatkan metode *Support Vector Machine* serta fitur *Local Binary Pattern*. Adapun dataset yang digunakan bersumber dari GitHub sebanyak 2015 citra dengan format JPG. Jenis sampah yang diklasifikasikan adalah kardus, kaca, logam, kertas, dan plastik. Pengujian dilakukan menggunakan *K-Fold Cross Validation* dengan jenis *Leave One Out* (LOO) yang membagi dataset menjadi data *training* dan data *testing*. Penelitian ini menerapkan *kernel* linear, polinomial, dan gaussian pada SVM. Hasil yang diperoleh menunjukkan bahwa *kernel* polinomial memberikan hasil terbaik dibandingkan dengan *kernel* lainnya, dengan rata-rata akurasi 87,82%.

Penelitian yang dilakukan oleh Hutamaputra et al., (2022) membahas penerapan metode *Convolutional Neural Network* (CNN) yang digunakan untuk mengklasifikasikan sampah botol dengan fokus pada perbandingan dua arsitektur, yakni VGG16 dan ResNet34. Dataset yang digunakan bersumber dari GitHub dan Kaggle, terdiri dari 4449 citra yang diklasifikasikan ke dalam 2 kategori yaitu botol plastik dan botol bukan plastik. Dataset ini dibagi menjadi 90% untuk data latih dan

10% untuk data validasi. Akurasi yang diperoleh dari penelitian menunjukkan bahwa akurasi VGG16 mencapai akurasi 96,39%, sementara ResNet34 memiliki akurasi 91,37%.

Yujie He et al., (2020) melakukan penelitian mengenai klasifikasi sampah menggunakan metode *Convolutional Neural Network* (CNN) dengan menggunakan dataset yang mencakup 2527 citra sampah yang diklasifikasikan dalam 6 kategori yaitu kardus, kaca, logam, kertas, plastik, serta sampah umum. Arsitektur CNN yang digunakan adalah AlexNet yang dimodifikasi, di mana dua lapisan dihilangkan untuk efisiensi. Pembagian data dilakukan dengan proporsi 75% untuk pelatihan dan 25% untuk pengujian. Penelitian ini memanfaatkan beberapa teknik seperti *dropout*, *data augmentation*, dan penurunan laju pembelajaran. Penelitian ini juga membandingkan dua jenis pengklasifikasian di lapisan terakhir yaitu *softmax* dan *Support Vector Machine* (SVM). Hasil paling optimal diperoleh dari model yang menerapkan SVM dan teknik *partial data augmentation*, dengan akurasi 79,94%.

Penelitian yang dilakukan oleh Dacipta & Putra (2022) membahas penggunaan metode *Convolutional Neural Network* (CNN) dengan arsitektur VGGNet untuk melakukan klasifikasi limbah, dengan dataset yang berisi 8371 citra dari Kaggle dan Google Drive mencakup 9 kategori limbah yaitu baterai, pakaian, e-limbah, kaca, bola lampu, logam, organik, kertas, dan plastik. Model ini dievaluasi menggunakan *K-Fold Cross Validation* untuk memastikan performa dalam proses pelatihan maupun pengujian. Akurasi tertinggi yang dicapai pada

proses *training* sebesar 69,77% setelah 28 *epoch*, sedangkan pada proses *testing* mencapai akurasi 64,45%.

Sutanty & Kusuma Astuti (2023) melakukan penelitian tentang klasifikasi jenis sampah menggunakan model Arsitektur VGG16 dengan menggunakan dataset *Garbage Classification* dari Kaggle yang terdiri dari 8134 citra berformat JPG. Dataset ini terdiri dari 10 kelas yaitu kaca cokelat, kardus, kaca hijau, logam, kertas, plastik, kaca putih, baterai, masker, dan sampah organik. Dataset tersebut dibagi menjadi tiga bagian yaitu 70% untuk pelatihan, 15% untuk validasi, dan 15% untuk pengujian. Proses *training* dilakukan dengan 40 *epoch* dan hasil akurasi 82,89% pada data pelatihan dan 84,62% pada data validasi.

Penelitian yang dilakukan oleh Nisa et al., (2022) mengklasifikasikan citra sampah dengan metode *Support Vector Machine* (SVM) yang menggunakan ekstraksi fitur *Gray Level Co-occurrence Matrix* (GLCM) dan *color moments*. Dataset yang digunakan bersumber dari GitHub yang mencakup 2527 citra berformat JPG. Dataset tersebut mencakup 6 kategori yaitu kardus, kaca, logam, plastik, kertas, dan sampah lainnya. Proses ekstraksi fitur menggunakan metode GLCM untuk mendapatkan informasi tekstur serta *color moments* untuk menangkap karakteristik warna. Evaluasi dilakukan menggunakan teknik *10-Fold Cross Validation* dan hasilnya menunjukkan bahwa kombinasi fitur GLCM dan *color moments* memberikan rata-rata akurasi terbaik sebesar 85,43% yang dicapai pada *fold* ke-3.

Kurniawan et al., (2023) melakukan penelitian tentang klasifikasi sampah anorganik menggunakan metode *Convolutional Neural Network* (CNN) dengan

dataset yang diperoleh dari Kaggle dan beberapa data tambahan dari penulis dengan total 2647 citra dengan 6 kelas sampah anorganik yaitu kardus, kaca, logam, kertas, plastik, dan sampah. Dataset dibagi menjadi 70% untuk pelatihan, 15% untuk validasi, dan 15% untuk pengujian. Model CNN yang digunakan dirancang menggunakan arsitektur *Xception* dan dilatih menggunakan *optimizer Adam* dengan *learning rate* 0,001. Hasil penelitian menunjukkan bahwa model CNN mampu mengklasifikasikan sampah anorganik dengan akurasi 87,81%.

Penelitian terkait yang dijadikan sebagai referensi untuk melakukan penelitian mengenai penerapan CNN dalam klasifikasi sampah berbasis citra Tabel 2.1.

Tabel 2.1 Penelitian Terkait

No	Nama Peneliti	Judul Penelitian	Metode	Dataset	Jenis Sampah	Hasil Penelitian
1	(Leonardo et al., 2020)	Klasifikasi Sampah Daur Ulang Menggunakan <i>Support Vector Machine</i> dengan Fitur <i>Local Binary Pattern</i>	<i>Support Vector Machine</i> serta fitur <i>Local Binary Pattern</i>	Bersumber dari GitHub, 2015 citra	Kardus, kaca, logam, kertas, plastik	<i>Kernel</i> polinomial memberikan akurasi terbaik dengan rata-rata 87,82%.
2	(Hutamaputra et al., 2022)	Perbandingan Kinerja <i>Convolutional Neural Network</i> VGG16 dan ResNet34 pada Sistem Klasifikasi Sampah Botol	<i>Convolutional Neural Network</i> dengan arsitektur VGG16 dan ResNet34	Bersumber dari GitHub dan Kaggle, 4449 citra	Botol plastik, botol bukan plastik	VGG16 mencapai akurasi 96,39%, sedangkan ResNet mencapai akurasi 91,37%.
3	(Yujie He et al., 2020)	<i>Trash Classification Using Convolutional Neural Networks Project Category:</i>	<i>Convolutional Neural Network</i> dengan arsitektur AlexNet	2527 citra	Kardus, kaca, logam, kertas, plastik, sampah umum	Klasifikasi dengan <i>Support Vector Machine</i> dan <i>partial data augmentation</i> mencapai

No	Nama Peneliti	Judul Penelitian	Metode	Dataset	Jenis Sampah	Hasil Penelitian
		<i>Computer Vision</i>				akurasi 79,94%.
4	(Dacipta & Putra, 2022)	Sistem Klasifikasi Limbah Menggunakan Metode <i>Convolutional Neural Network</i> (CNN) pada <i>Web Service</i> Berbasis <i>Framework Flask</i>	<i>Convolutional Neural Network</i> dengan arsitektur VGGNet	Bersumber dari Kaggle dan Google Drive, 8371 citra	Baterai, pakaian, e-limbah, kaca, bola lampu, logam, organik, kertas, plastik	Akurasi tertinggi pada proses <i>training</i> mencapai 69,77%, sedangkan pada proses <i>testing</i> mencapai 64,45%.
5	(Sutanty & Kusuma Astuti, 2023)	Penerapan Model Arsitektur VGG16 untuk Klasifikasi Jenis Sampah	<i>Convolutional Neural Network</i> dengan arsitektur VGG16	Bersumber dari Kaggle, 8134 citra	Kaca coklat, kardus, kaca hijau, logam, kertas, plastik, kaca putih, baterai, masker, sampah organik	Hasil akurasi 82,89% pada data pelatihan dan 84,62% pada data validasi.
6	(Nisa et al., 2022)	Klasifikasi Citra Sampah Menggunakan <i>Support Vector Machine</i> dengan Ekstraksi Fitur <i>Gray Level Co-occurrence Matrix</i> dan <i>Color Moments</i>	<i>Support Vector Machine</i> dengan ekstraksi fitur <i>Gray Level Co-occurrence Matrix</i> dan <i>color moments</i>	Bersumber dari GitHub, 2527 citra	Kardus, kaca, logam, plastik, kertas, sampah lainnya	Kombinasi fitur <i>Gray Level Co-occurrence Matrix</i> serta <i>color moments</i> menghasilkan rata-rata akurasi 85,43% pada <i>fold</i> ke-3.
7	(Kurniawan et al., 2023)	Implementasi Arsitektur <i>Xception</i> pada Model <i>Machine Learning</i> Klasifikasi Sampah Anorganik	<i>Convolutional Neural Network</i> dengan arsitektur <i>Xception</i>	Bersumber dari Kaggle serta data tambahan dari penulis, 2647 citra	Kardus, kaca, logam, kertas, plastik, sampah	Model <i>Convolutional Neural Network</i> dengan arsitektur <i>Xception</i> mencapai akurasi 87,81%.

Berdasarkan penelitian yang relevan, dapat disimpulkan bahwa metode CNN terbukti mampu menghasilkan akurasi yang baik dalam hal pengklasifikasian citra. Oleh karena itu, penelitian ini berfokus pada penerapan CNN dalam mengklasifikasikan sampah anorganik khususnya *paper* dan *plastic* berbasis citra.

## **2.2 Sampah**

Sampah merupakan material sisa yang dihasilkan dari berbagai aktivitas manusia maupun proses alamiah yang sudah tidak memiliki manfaat. Secara umum, sampah diklasifikasikan menjadi dua kategori utama, yakni sampah organik dan anorganik, yang masing-masing memiliki karakteristik dan dampak lingkungan yang berbeda. Menurut Undang-Undang No. 18 Tahun 2008, sampah didefinisikan sebagai sisa aktivitas manusia. Jika pengelolaannya tidak dilakukan dengan tepat, permasalahan sampah dapat menimbulkan penurunan kualitas lingkungan (Majida et al., 2023). Pemahaman tentang sifat dan jenis sampah penting untuk mendukung pengelolaan sampah serta mengurangi risiko kerusakan lingkungan akibat penumpukan sampah.

Sampah organik adalah jenis sampah yang berasal dari makhluk hidup, terutama tumbuhan dan hewan yang mampu terurai secara alami oleh mikroorganisme. Jenis sampah ini sering disebut sebagai sampah basah karena dalam kondisi tertentu mudah membusuk dan terdekomposisi dengan cepat. Contoh dari sampah organik yaitu sisa makanan, daun, ranting, dan kulit buah. Proses penguraian ini sering kali menghasilkan kompos yang bermanfaat untuk meningkatkan kualitas tanah di bidang pertanian dan penghijauan (Hamidah et al.,

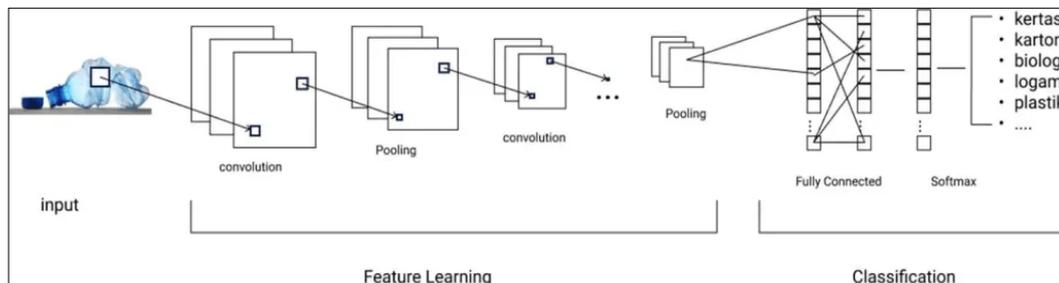
2023). Selain itu, pengelolaan sampah organik juga dapat memberikan manfaat tambahan seperti biogas yang bisa dimanfaatkan sebagai energi terbarukan.

Di sisi lain, sampah anorganik adalah material yang sulit terurai secara alami dan memerlukan waktu yang sangat lama untuk hancur. Sampah ini berasal dari bahan-bahan seperti plastik, logam, kaca, kertas yang lebih sulit ditangani karena sifatnya yang tahan lama. Plastik sebagai salah satu jenis sampah anorganik paling umum yang membutuhkan waktu lama untuk terurai dan sering kali menyebabkan pencemaran lingkungan, terutama di lautan (Zahra et al., 2024). Oleh karena itu, penanganan sampah anorganik melibatkan proses daur ulang yang lebih kompleks, terutama untuk *paper* dan *plastic* yang menjadi fokus dalam penelitian ini. Melalui proses daur ulang yang tepat, material-material tersebut dapat diproses kembali menjadi produk yang lebih ramah lingkungan.

### **2.3 Convolutional Neural Network (CNN)**

CNN merupakan salah satu metode *deep learning* yang umum digunakan dalam pengolahan citra. Sebagai bagian dari jaringan saraf, CNN dirancang untuk meniru proses pembelajaran otak manusia menggunakan program komputer (Suhartono, 2012). CNN telah diaplikasikan dalam berbagai pengenalan citra, seperti klasifikasi, segmentasi, serta deteksi objek. Fungsi utama dari CNN adalah untuk mengekstrak dan mengenali pola visual dalam citra melalui proses yang mendalam dan bertahap. Salah satu keunggulan CNN yaitu dapat bekerja secara otomatis untuk mengekstrak fitur dari data *input* tanpa memerlukan pra-pemrosesan manual yang rumit. CNN juga mampu melakukan klasifikasi objek berdasarkan fitur-fitur yang diekstraksi, sehingga menjadi solusi yang tepat dalam pengenalan

objek dan klasifikasi citra (Sunanto & Utomo, 2022). Contoh arsitektur *Convolutional Neural Network* (CNN) Gambar 2.1.



Gambar 2.1 Arsitektur *Convolutional Neural Network* (CNN)

Arsitektur CNN tersusun dari beberapa lapisan utama yang berperan dalam proses pembelajaran fitur dan klasifikasi. Lapisan utama dalam CNN meliputi *convolutional layer*, *pooling layer*, dan *fully connected layer*. *Convolutional layer* berperan dalam mengenali fitur lokal seperti garis tepi, pola, dan tekstur dengan menggunakan *filter* atau *kernel* yang melakukan proses konvolusi pada setiap bagian citra. Hasil dari proses ini adalah peta fitur (*feature map*) yang merepresentasikan pola-pola visual pada citra. Setelah itu, *pooling layer* berfungsi untuk menurunkan dimensi dari *feature map* dengan cara mengambil nilai maksimum di area tertentu, biasanya dilakukan melalui teknik *max pooling*. Proses *pooling* membantu mengurangi jumlah parameter dan meningkatkan ketahanan model terhadap variasi dalam citra.

Pada tahap akhir, *fully connected layer* menghubungkan semua neuron dari lapisan sebelumnya untuk melakukan proses klasifikasi berdasarkan fitur-fitur yang telah diekstraksi. CNN juga menggunakan fungsi aktivasi seperti *ReLU* (*Rectified Linear Unit*) dan *sigmoid*. Fungsi aktivasi *ReLU* bertujuan untuk menambahkan elemen non-linearitas pada model, sehingga dapat menangkap pola dalam citra.

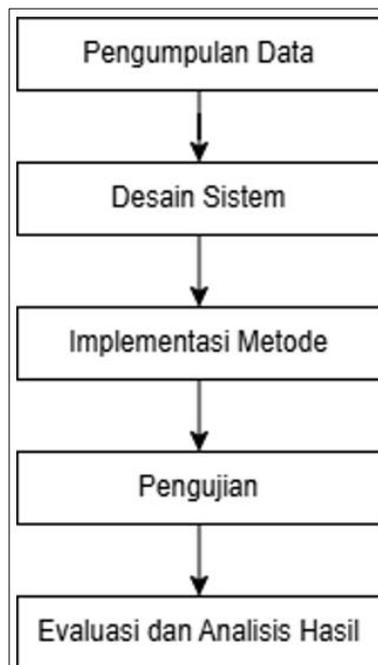
Sedangkan, fungsi *sigmoid* digunakan pada lapisan *output* untuk menghasilkan probabilitas klasifikasi, terutama dalam hal klasifikasi biner. Agar model terhindar dari *overfitting*, sering kali diterapkan teknik regularisasi seperti *dropout* yang secara acak mengabaikan beberapa neuron selama proses pelatihan. Semua lapisan ini bekerja untuk membentuk model yang mampu mengenali dan mengklasifikasikan objek dalam citra dengan akurasi tinggi (Peryanto et al., 2020).

## BAB III

### METODE PENELITIAN

#### 3.1 Desain Penelitian

Desain penelitian ini mencakup sejumlah tahapan yang dilakukan secara berurutan guna mengarahkan peneliti dalam mencapai hasil penelitian sesuai dengan tujuan yang telah ditetapkan. Dimulai dari pengumpulan data, yang dilanjutkan dengan desain sistem untuk merancang proses yang akan diterapkan pada model. Tahap berikutnya adalah implementasi metode, di mana model dilatih dan diuji untuk memperoleh hasil yang optimal. Untuk mengukur kinerja model, dilakukan pengujian dengan berbagai skenario. Tahap terakhir yaitu evaluasi dan analisis hasil untuk menilai efektivitas model dan mengambil kesimpulan dari kinerja yang diperoleh. Tahapan desain penelitian Gambar 3.1.

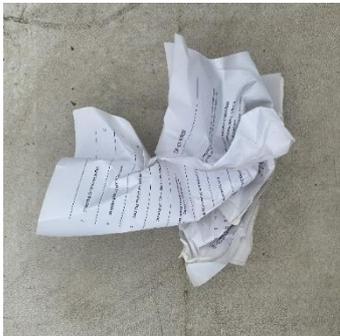


Gambar 3.1 Desain Penelitian

### 3.2 Pengumpulan Data

Dalam penelitian ini, untuk melakukan klasifikasi sampah berbasis citra digunakan data sekunder yang diperoleh melalui website resmi Kaggle. Dataset yang digunakan yaitu “*Real Waste Image Classification*” dengan resolusi gambar  $524 \times 524$  piksel dan “*Trash Images*” dengan resolusi gambar  $384 \times 512$  piksel. Kedua dataset ini berisi citra sampah yang mencakup kategori sampah *paper* dan *plastic*. Jumlah data yang digunakan dalam penelitian ini sebanyak 2000 citra, dengan masing-masing kategori sebanyak 1000 citra berformat JPG. Contoh sampel citra sampah Tabel 3.1.

Tabel 3.1 Contoh Sampel Citra Sampah *Paper* dan *Plastic*

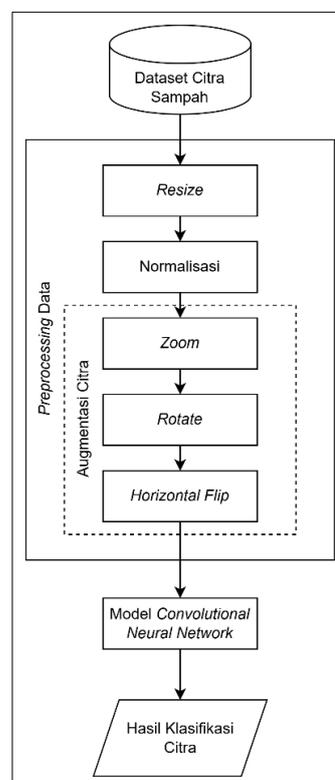
No	Citra Sampah <i>Paper</i>	Citra Sampah <i>Plastic</i>
1		
2		

No	Citra Sampah <i>Paper</i>	Citra Sampah <i>Plastic</i>
3		

### 3.3 Desain Sistem

Desain sistem dirancang untuk memberikan gambaran umum tentang bagaimana proses klasifikasi citra sampah dilakukan menggunakan metode CNN.

Desain sistem penelitian Gambar 3.2.



Gambar 3.2 Desain Sistem

### 3.3.1 *Input Citra*

Pada penelitian ini, citra sampah *paper* dan *plastic* yang telah diperoleh akan digunakan sebagai *input* sistem klasifikasi. Dataset citra sampah *paper* dan *plastic* berformat JPG dengan total citra 2000, terdiri dari 1000 citra *paper* dan 1000 citra *plastic*.

### 3.3.2 *Preprocessing Data*

*Preprocessing* data adalah tahap untuk mempersiapkan citra sebelum dimasukkan ke dalam model. Tujuannya untuk meningkatkan efisiensi dan akurasi model dengan memastikan data yang diolah konsisten dan lebih mudah dipelajari oleh model.

#### 3.3.2.1 *Resize*

*Resize* adalah proses mengubah dimensi citra ke ukuran tertentu agar semua citra dalam dataset memiliki ukuran yang konsisten. Dalam penelitian ini, citra diubah ukurannya menjadi  $128 \times 128$  piksel, di mana setiap piksel baru merupakan pencampuran dari empat piksel terdekat pada citra asli. Pemilihan ukuran ini didasarkan pada temuan dari penelitian sebelumnya yang dilakukan oleh Mishkin et al., (2016) yang menunjukkan bahwa ukuran ini cukup untuk mengenali fitur penting tanpa mempengaruhi kinerja model. Selain itu, pemilihan ukuran ini juga membantu pelatihan model dilakukan lebih cepat dengan tetap mempertahankan akurasi yang baik, sehingga menjadi pilihan yang efisien antara waktu pelatihan dan kualitas hasil.

### 3.3.2.2 Normalisasi

Normalisasi bertujuan untuk mengubah nilai pixel citra dari rentang 0-255 menjadi 0-1. Proses ini membantu mempercepat konvergensi model selama pelatihan, sehingga memudahkan proses optimisasi fungsi *loss*. Rumus normalisasi Persamaan 3.1.

$$\text{nilai\_normalisasi} = \frac{\text{nilai\_pixel}}{255} \quad (3.1)$$

Keterangan:

nilai\_pixel : nilai pixel asli dari citra, berkisar antara 0-255

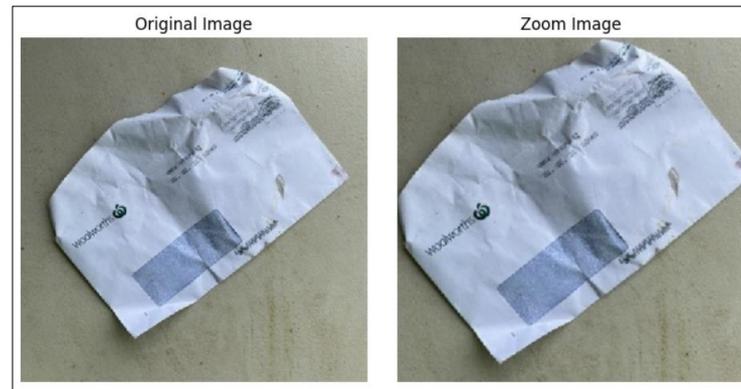
nilai\_normalisasi : hasil normalisasi, berkisar antara 0-1

### 3.3.2.3 Augmentasi Citra

Augmentasi citra adalah teknik yang digunakan untuk meningkatkan keragaman dataset dengan menerapkan transformasi acak pada citra latih. Hal ini bertujuan untuk membantu model menjadi lebih *robust* dan mampu mengenali objek dalam berbagai kondisi, sehingga dapat mengurangi *overfitting* dan meningkatkan akurasi model saat menghadapi data baru.

#### 1. *Zoom*

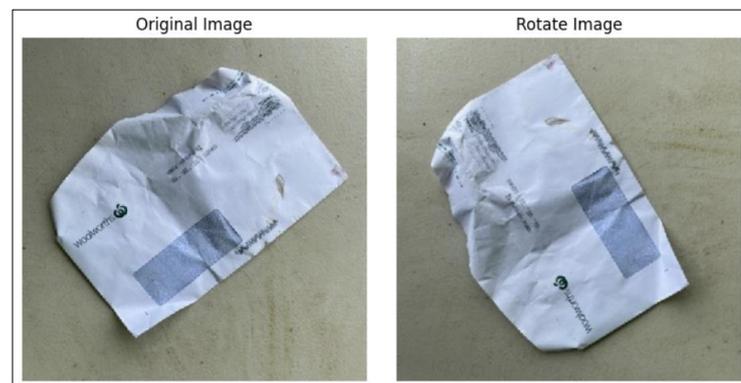
Digunakan untuk memperbesar atau memperkecil citra sehingga model dapat mengenali objek dalam berbagai ukuran pada rentang skala tertentu. Dalam proses ini, perubahan skala dilakukan dalam rentang 20% dari ukuran aslinya, sehingga citra dapat diperbesar atau diperkecil untuk meningkatkan keberagaman data yang membantu model menjadi lebih adaptif terhadap variasi ukuran objek dalam gambar. *Output* dari teknik *zoom* Gambar 3.3.



Gambar 3.3 Zoom

## 2. Rotate

Digunakan untuk memutar citra dalam rentang sudut tertentu agar model dapat mengenali objek dengan berbagai orientasi. Dalam proses ini, citra diputar dalam rentang  $90^\circ$  dari posisi aslinya, sehingga model lebih mudah mengenali objek meskipun mengalami perubahan sudut. *Output* dari teknik *rotate* Gambar 3.4.

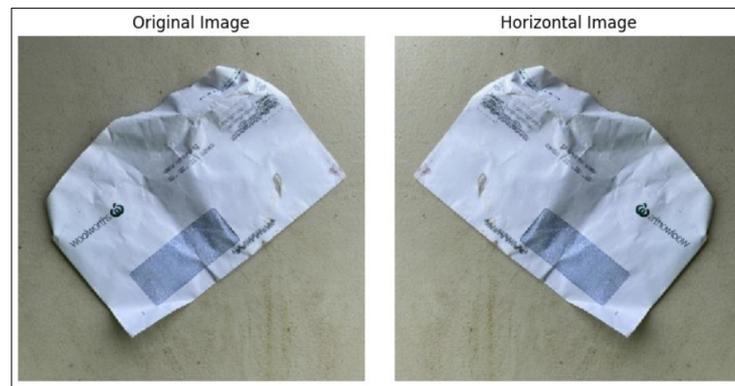


Gambar 3.4 Rotate

## 3. Horizontal Flip

Digunakan untuk membalik citra secara horizontal, sehingga model dapat mengenali objek dalam orientasi terbalik dan meningkatkan variasi data.

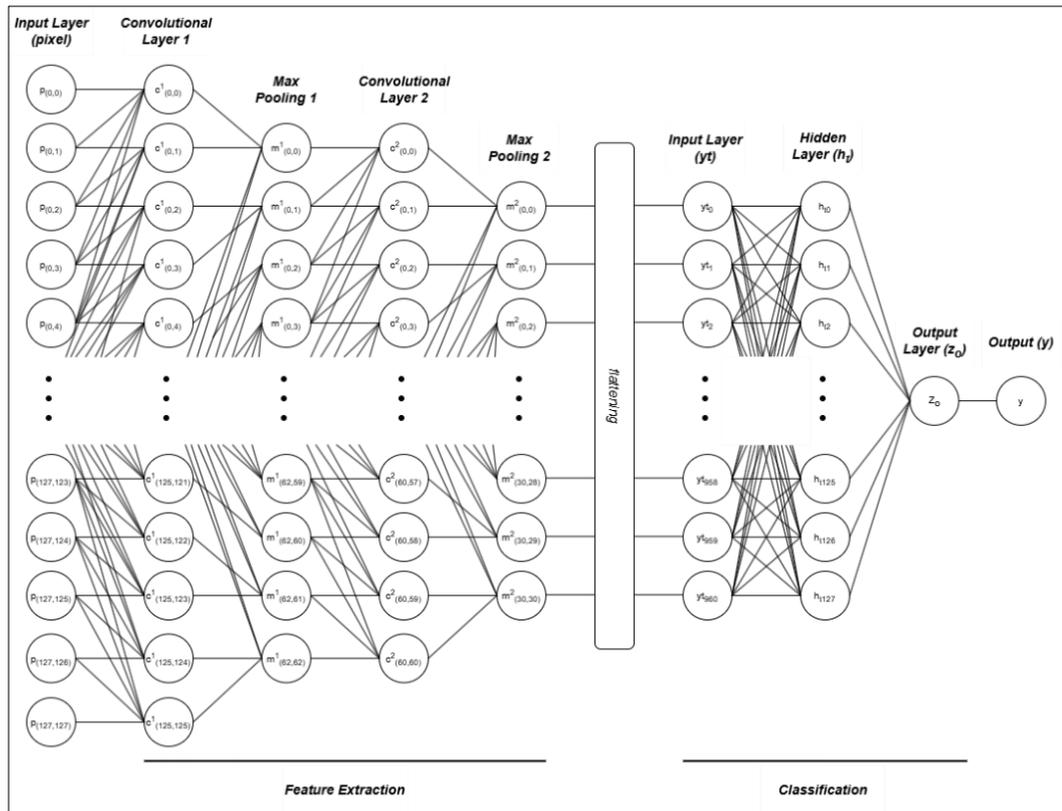
Proses ini dilakukan dengan menukar posisi piksel di sisi kiri maupun kanan gambar. *Output* dari teknik *horizontal flip* Gambar 3.5.



Gambar 3.5 *Horizontal Flip*

### 3.4 Implementasi Metode

Terdapat dua tahapan utama dalam proses ini yaitu ekstraksi fitur dan klasifikasi. Proses ekstraksi fitur berfungsi untuk mengambil informasi penting dari data citra, sementara klasifikasi digunakan untuk menentukan kelas dari citra berdasarkan fitur yang telah diekstraksi. Arsitektur CNN yang digunakan dalam penelitian ini merujuk pada arsitektur yang digunakan oleh Al-Mamun et al., (2023), diantaranya terdapat *input layer*, *2 convolutional layer*, *2 max pooling layer*, *flatten layer*, *hidden layer*, *output layer*, dan *output* Gambar 3.6.



Gambar 3.6 Arsitektur CNN

Data citra yang digunakan dalam penelitian ini terbagi menjadi dua kelas yakni *paper* dan *plastic* dengan resolusi *input* sebesar (128,128,3) di mana angka 3 merepresentasikan tiga *channel* RGB. Proses pembuatan model dimulai dengan memasukkan citra dari kedua kelas tersebut ke dalam *convolutional layer*, di mana citra tersebut melewati *kernel* untuk mengenali fitur-fitur utama seperti garis tepi dan bentuk objek sampah. Setelah itu, citra melewati *max pooling layer* yang berfungsi untuk mengurangi dimensi *output* dari *convolutional layer* serta mengekstraksi fitur-fitur yang relevan. Kemudian, hasil dari *max pooling layer* diteruskan ke *fully connected layer* yang digunakan untuk melakukan klasifikasi terhadap kategori citra sampah yang telah ditentukan. Terakhir, pada lapisan *output* terdapat satu neuron yang memberikan hasil prediksi akhir dalam bentuk nilai

probabilitas, yang menentukan citra tersebut tergolong dalam kategori *paper* atau *plastic*.

### 3.4.1 Convolutional Layer

Pada *convolutional layer*, proses konvolusi dilakukan dengan menggeser *kernel* berukuran  $3 \times 3$  pada *input* citra untuk mengekstraksi fitur-fitur penting. Tujuan dari proses ini adalah mendeteksi pola-pola yang relevan, seperti tepi dalam citra. Setiap langkah konvolusi menghasilkan keluaran berupa *feature map* yang merepresentasikan fitur-fitur yang telah diekstraksi. Perhitungan proses konvolusi Persamaan 3.2.

$$FM_{(a,b)} = \left( \sum_{h=0}^{k_h-1} \sum_{w=0}^{k_w-1} K_{(h,w)} \times X_{(a+h,b+w)} \right) \quad (3.2)$$

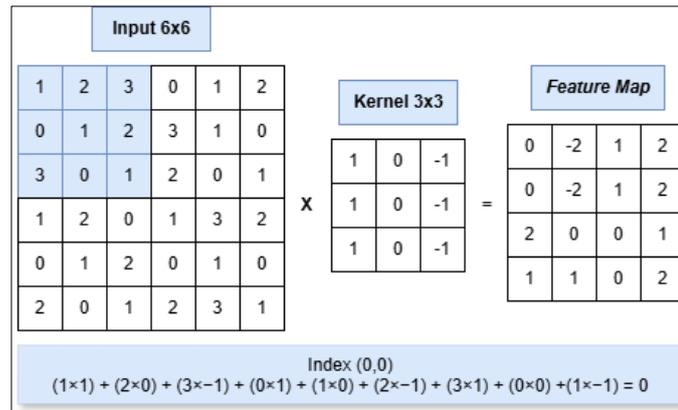
Keterangan:

$FM_{(a,b)}$  : *output* konvolusi pada posisi  $(a, b)$

$K_{(h,w)}$  : *kernel* dengan ukuran  $k_h \times k_w$

$X_{(a+h,b+w)}$  : nilai piksel *input* di posisi  $(a + h, b + w)$  untuk konvolusi

Ukuran *feature map* yang dihasilkan tergantung pada ukuran *kernel* dan *stride*. Pada penelitian ini, *stride* yang digunakan adalah 1 dan *padding* 0, sehingga *kernel* bergerak 1 piksel setiap kali diterapkan pada *input* citra. Berikut contoh perhitungan manual proses *convolutional*, di mana terdapat *input* berukuran  $6 \times 6$  dan *kernel* berukuran  $3 \times 3$  Gambar 3.7.

Gambar 3.7 Perhitungan Proses *Convolutional*

Setelah proses konvolusi selesai, diterapkan fungsi aktivasi ReLU pada setiap nilai dalam *feature map*. Fungsi aktivasi ReLU Persamaan 3.3.

$$f(x) = \max(0, x) \quad (3.3)$$

Keterangan:

$f(x)$  : hasil dari fungsi aktivasi ReLU  
 $x$  : nilai hasil konvolusi pada *feature map*

Fungsi aktivasi ReLU mengubah semua nilai negatif menjadi nol dan mempertahankan nilai positif apa adanya. Penambahan aktivasi ReLU menjadikan *feature map* hanya akan berisi nilai positif, sehingga fitur yang relevan lebih menonjol untuk dilanjutkan ke lapisan berikutnya.

### 3.4.2 Max Pooling Layer

Lapisan *max pooling* digunakan untuk mengurangi ukuran *feature map* yang diperoleh dari lapisan konvolusi sebelumnya. Proses *max pooling* dilakukan dengan mengambil nilai maksimum dari setiap area kecil pada *feature map*. Tujuan utama dari *max pooling layer* adalah untuk menyederhanakan representasi data, mengurangi kompleksitas komputasi, dan mencegah *overfitting* dengan

mempertahankan fitur yang paling signifikan. Operasi *pooling* juga membantu dalam meningkatkan ketahanan model terhadap pergeseran dan distorsi kecil pada *input* citra. Pada penelitian ini, menggunakan *kernel*  $2 \times 2$  dengan *stride* 2 yang berfungsi untuk mengambil nilai maksimum  $2 \times 2$  pada *feature map* dan melangkah 2 piksel berikutnya. Perhitungan proses *max pooling* Persamaan 3.4.

$$P(a, b) = \max \{X(a + h, b + w)\} \quad (3.4)$$

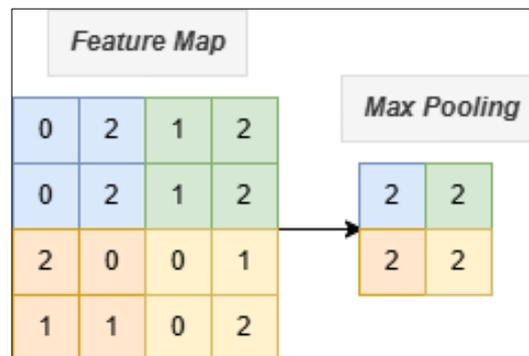
Keterangan:

$P(a, b)$  : *output pooling* di posisi  $(a, b)$

$X(a + h, b + w)$  : nilai piksel *input* dalam area *pooling*

$h$  dan  $w$  : indeks yang menentukan area lokal dalam *pooling window*

Setelah operasi *max pooling* selesai, ukuran *feature map* akan menjadi lebih kecil, namun tetap mempertahankan informasi penting dari fitur citra yang dominan. Contoh penerapan *max pooling* terhadap *feature map* Gambar 3.8.

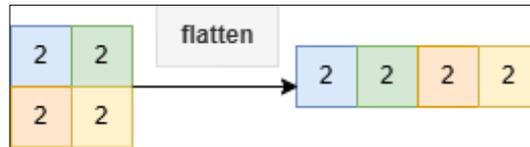


Gambar 3.8 Penerapan Proses *Max Pooling*

### 3.4.3 Fully Connected Layer

*Fully connected layer* berfungsi untuk menghasilkan *output* klasifikasi berdasarkan fitur-fitur yang telah diperoleh dari proses sebelumnya. *Feature map* yang dihasilkan dari proses ekstraksi fitur masih berada dalam bentuk array multidimensional. Sebelum *output* dari lapisan sebelumnya dapat diproses di dalam

*fully connected layer*, *output* tersebut perlu di *flatten* terlebih dahulu menjadi vektor satu dimensi. Contoh hasil dari proses *flatten* Gambar 3.9.



Gambar 3.9 Penerapan Proses *Flatten*

Setelah proses *flatten* selesai, perhitungan untuk *fully connected layer* Persamaan 3.5.

$$y_t = \sum_{s=1}^s w_{t,s} \times h_s + b_1 \quad (3.5)$$

Keterangan:

- $y_t$  : *output* neuron ke-t
- $w_{t,s}$  : bobot dari *input* ke-s
- $h_s$  : *input* dari layer sebelumnya
- $b_1$  : bias dari neuron ke-t

Setelah menghitung  $y_t$ , *output* ini melalui fungsi aktivasi ReLU Persamaan 3.6.

$$h_t = \max(0, y_t) \quad (3.6)$$

Keterangan:

- $h_t$  : *output* setelah aktivasi ReLU

*Output* yang telah diaktivasi dari *hidden layer* diteruskan ke lapisan *output* Persamaan 3.7.

$$z_o = \sum_{t=1}^T w(o, t) \times h_t + b_2 \quad (3.7)$$

Keterangan:

- $z_o$  : *output* sebelum aktivasi di lapisan *output*
- $w(o, t)$  : bobot dari neuron ke-t di *hidden layer* ke neuron *output* ke-o
- $b_2$  : bias pada neuron *output*

Setelah menghitung  $z_o$ , selanjutnya menggunakan aktivasi *sigmoid* untuk menghasilkan *output* akhir Persamaan 3.8.

$$\hat{y} = \sigma(z_o) = \frac{1}{1 + e^{-z_o}} \quad (3.8)$$

Keterangan:

$\hat{y}$  : probabilitas hasil klasifikasi (*output* akhir)

Setelah memperoleh probabilitas hasil klasifikasi melalui fungsi *sigmoid*, langkah berikutnya adalah penentuan kelas prediksi menggunakan metode *threshold*. Dalam hal ini, *output* dari fungsi *sigmoid* dibandingkan dengan nilai ambang batas yang ditetapkan pada 0.5. Penggunaan ambang batas 0.5 merujuk pada penjelasan yang diberikan oleh Brownlee (2021), yang menyatakan bahwa penggunaan *threshold* 0.5 didasarkan pada sifat fungsi *sigmoid*, yang memberikan *output* dalam rentang 0 hingga 1, sehingga nilai 0.5 digunakan sebagai titik tengah yang secara simetris membagi *output* menjadi dua kategori klasifikasi. Nilai lebih besar dari 0.5 menunjukkan kelas positif, sementara nilai di bawah 0.5 menunjukkan kelas negatif. Oleh karena itu, jika probabilitas yang dihasilkan lebih besar dari 0.5, citra tersebut akan diklasifikasikan sebagai kelas *plastic*, ditandai dengan label 1. Sebaliknya, jika probabilitas kurang dari atau sama dengan 0.5, citra akan diklasifikasikan sebagai kelas *paper*, ditandai dengan label 0. Proses ini merupakan tahap akhir dalam implementasi metode, di mana setiap citra akan melalui langkah ini untuk menentukan klasifikasinya, sehingga model dapat memberikan hasil prediksi yang jelas antara dua kelas yaitu *paper* atau *plastic*.

### 3.5 Skenario Pengujian

Penelitian ini menerapkan berbagai skenario pengujian guna menemukan kombinasi parameter yang mampu memberikan performa terbaik pada model untuk melakukan klasifikasi sampah *paper* dan *plastic*.

Skenario pertama adalah penggunaan augmentasi citra pada model. Pada bagian *preprocessing* data, peneliti menggunakan augmentasi citra untuk meningkatkan variasi data latih melalui transformasi seperti *zoom*, *rotate*, dan *horizontal flip*. Namun, dalam skenario ini peneliti melakukan pengujian tanpa augmentasi citra untuk mengetahui pengaruhnya terhadap performa model. Skenario augmentasi citra Tabel 3.2.

Tabel 3.2 Skenario Augmentasi Citra

Skenario	Augmentasi Citra
1	Tanpa augmentasi citra
2	Menggunakan augmentasi citra

Skenario kedua adalah penggunaan metode klasifikasi untuk menentukan hasil akhir prediksi model. Pada bagian implementasi metode, peneliti menggunakan metode *threshold* dengan nilai ambang batas 0.5 untuk menentukan kelas prediksi. Selain itu, peneliti juga mencoba beberapa nilai ambang batas lainnya, yaitu 0.3; 0.4; 0.6; 0.7 untuk melihat pengaruhnya terhadap hasil klasifikasi. Namun, dalam skenario ini peneliti melakukan pengujian menggunakan metode *one-hot encoding* untuk mengetahui pengaruh variasi metode klasifikasi terhadap kinerja model di tahap prediksi akhir, dengan cara membandingkan nilai prediksi tertinggi di *output layer* sebagai penentu kelas prediksi. Skenario metode klasifikasi Tabel 3.3.

Tabel 3.3 Skenario Metode Klasifikasi

Skenario	Metode Klasifikasi
A	<i>Threshold</i> dengan nilai ambang batas 0.3; 0.4; 0.5; 0.6; 0.7
B	<i>One-hot encoding</i>

Dari kombinasi skenario yang telah dirancang, diperoleh beberapa variasi pengujian model untuk mengetahui kombinasi yang memberikan hasil kinerja terbaik. Kombinasi skenario pengujian Tabel 3.4.

Tabel 3.4 Kombinasi Skenario Pengujian

Kombinasi	Kombinasi Skenario Pengujian
1A	Tanpa augmentasi citra dengan metode klasifikasi <i>threshold</i> dengan nilai ambang batas 0.3; 0.4; 0.5; 0.6; 0.7
1B	Tanpa augmentasi citra dengan metode klasifikasi <i>one-hot encoding</i>
2A	Menggunakan augmentasi citra dengan metode klasifikasi <i>threshold</i> dengan nilai ambang batas 0.3; 0.4; 0.5; 0.6; 0.7
2B	Menggunakan augmentasi citra dengan metode klasifikasi <i>one-hot encoding</i>

Sebelum melakukan uji coba setiap skenario, nilai parameter perlu ditetapkan terlebih dahulu. Setiap skenario menggunakan nilai parameter yang sama untuk memastikan konsistensi evaluasi. Dataset dibagi dengan perbandingan 80:20, di mana 80% data digunakan untuk *training* dan 20% untuk *testing*, guna memastikan model memiliki cukup data latih sekaligus menjaga kualitas evaluasi pada data uji. Parameter yang digunakan meliputi *batch size* yaitu jumlah data yang diproses sekaligus dalam satu iterasi pelatihan model, *dropout* yaitu teknik untuk menonaktifkan neuron secara acak untuk mencegah *overfitting*, *learning rate* yaitu besaran pembaruan bobot selama pelatihan model, dan *epoch* yaitu jumlah iterasi pelatihan pada seluruh dataset. Nilai masing-masing parameter Tabel 3.5.

Tabel 3.5 Nilai Parameter

Parameter	Nilai Parameter
<i>Batch Size</i>	32
<i>Dropout</i>	0.5
<i>Learning Rate</i>	0.001
<i>Epoch</i>	25

### 3.6 Evaluasi

Kualitas kinerja sebuah model dapat dievaluasi melalui sejumlah metrik seperti *accuracy*, *precision*, *recall*, dan *F1-score*, yang diperoleh berdasarkan analisis *confusion matrix*. *Confusion matrix* mencakup beberapa nilai penting yaitu *True Positive* (TP) ketika model memprediksi kelas positif dan prediksi tersebut benar, *True Negative* (TN) ketika model memprediksi kelas negatif dan prediksi tersebut benar, *False Positive* (FP) ketika model memprediksi positif namun aktualnya negatif, dan *False Negative* (FN) ketika model memprediksi negatif namun aktualnya positif. Dari nilai ini, tingkat *accuracy* yang menunjukkan seberapa baik model dalam melakukan klasifikasi, dihitung menggunakan Persamaan 3.9.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.9)$$

Selanjutnya, *precision* menunjukkan tingkat ketepatan model dalam memprediksi kelas positif, yaitu persentase prediksi positif yang benar dibandingkan dengan seluruh prediksi positif yang dihasilkan oleh model, dihitung menggunakan Persamaan 3.10.

$$Precision = \frac{TP}{TP + FP} \quad (3.10)$$

Selain itu, *recall* menunjukkan seberapa banyak kejadian positif yang berhasil dideteksi oleh model dibandingkan dengan seluruh kejadian positif yang sebenarnya ada, dihitung menggunakan Persamaan 3.11.

$$Recall = \frac{TP}{TP + FN} \quad (3.11)$$

F1-score memberikan keseimbangan antara *precision* dan *recall*, dihitung menggunakan Persamaan 3.12.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3.12)$$

Untuk memastikan bahwa model memiliki kemampuan generalisasi yang baik dan tidak hanya memberikan hasil optimal pada data tertentu, digunakan teknik validasi silang berupa *k-fold cross validation*. Teknik ini membagi dataset menjadi *k-fold*, kemudian proses pelatihan dan pengujian dilakukan sebanyak *k* kali. Pada setiap putaran, satu *fold* dijadikan sebagai data pengujian, sedangkan sisanya digunakan sebagai data pelatihan. Hasil evaluasi dari tiap iterasi kemudian dihitung rata-ratanya untuk memperoleh estimasi kinerja model yang lebih stabil dan representatif. Penerapan *k-fold cross validation* bertujuan untuk meminimalkan kemungkinan *overfitting* serta memastikan model dapat beradaptasi dengan baik terhadap variasi data yang berbeda.

## BAB IV

### HASIL DAN PEMBAHASAN

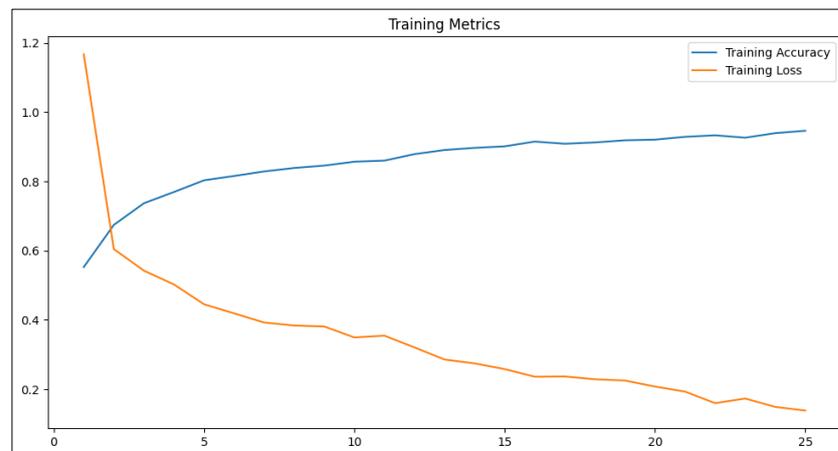
#### 4.1 Hasil Uji Coba

Pada penelitian ini, dilakukan uji coba melalui empat skenario, yaitu skenario uji 1A, 1B, 2A, 2B untuk mengevaluasi performa model CNN dalam mengklasifikasikan sampah. Setiap skenario menggunakan pendekatan yang berbeda, yaitu augmentasi citra dan metode klasifikasi *threshold* atau *one hot encoding*. Hasil yang diperoleh mencakup *training loss*, *training accuracy*, *confusion matrix*, beserta metrik evaluasi yang mencakup *precision*, *recall*, dan *F1-score*.

##### 4.1.1 Skenario Uji 1A

Pengujian pada skenario ini dilakukan menggunakan metode klasifikasi *threshold* tanpa augmentasi citra. Dataset yang digunakan terdiri dari citra sampah *paper* dan *plastic*, dengan total 2000 citra yang terdiri dari 1600 data pelatihan dan 400 data pengujian.

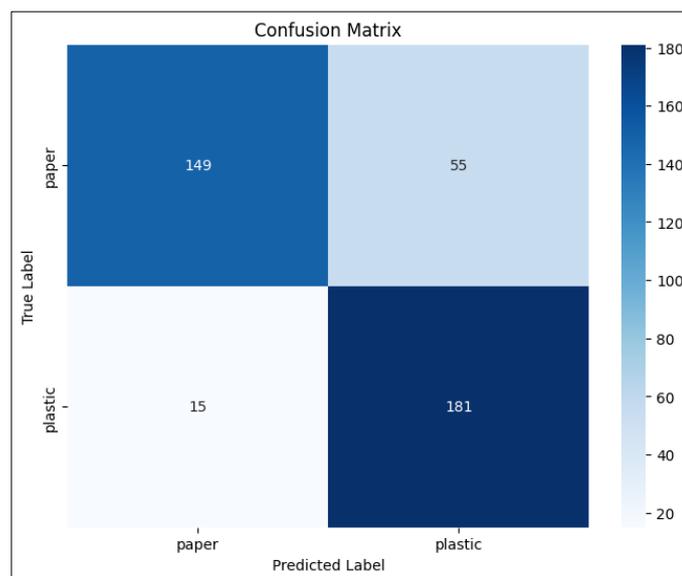
Proses pelatihan model menghasilkan nilai *training loss* sebesar 0,3545 dan *training accuracy* sebesar 85,59%. Model mencapai konvergensi pada *epoch* 15, di mana nilai *loss* dan *accuracy* mulai stabil serta tidak mengalami perubahan signifikan setelahnya. Grafik pelatihan perubahan nilai *loss* dan *accuracy* Gambar 4.1.



Gambar 4.1 Grafik *Training Accuracy* dan *Loss* Skenario Uji 1A

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.3 menunjukkan bahwa model berhasil memprediksi 149 sampel *paper* dengan benar dan 181 sampel *plastic* dengan benar. Selain itu, terdapat 55 sampel *paper* yang salah diprediksi sebagai *plastic* dan 15 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold 0.3*

Gambar 4.2.



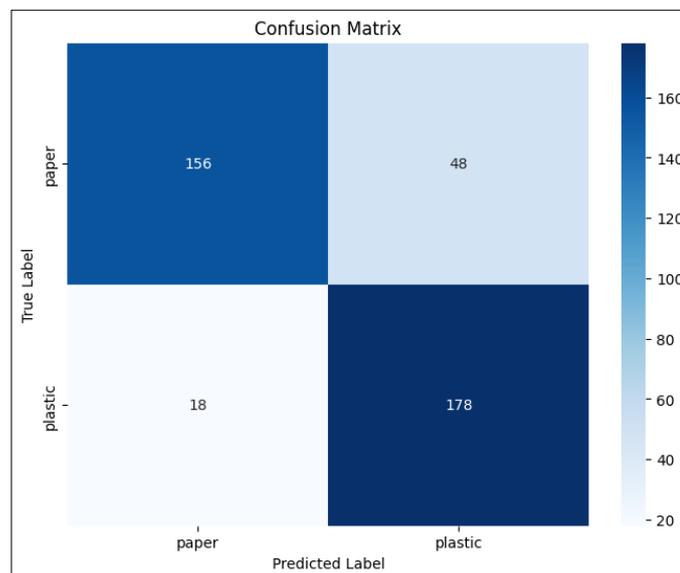
Gambar 4.2 *Confusion Matrix* Skenario Uji 1A *Threshold 0.3*

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.3, model mencapai akurasi *testing* sebesar 82%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.1.

Tabel 4.1 Metrik Evaluasi Skenario Uji 1A *Threshold* 0.3

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	91%	73%	81%
<i>Plastic</i>	77%	92%	84%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.4 menunjukkan bahwa model berhasil memprediksi 156 sampel *paper* dengan benar dan 178 sampel *plastic* dengan benar. Selain itu, terdapat 48 sampel *paper* yang salah diprediksi sebagai *plastic* dan 18 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.4 Gambar 4.3.



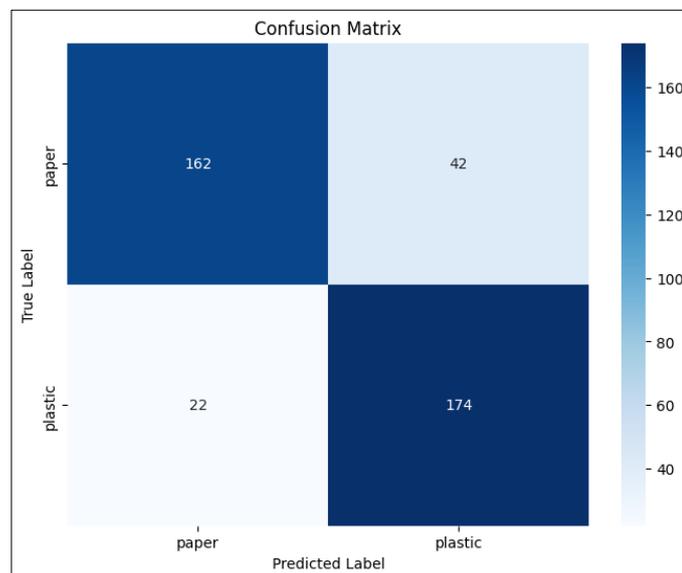
Gambar 4.3 *Confusion Matrix* Skenario Uji 1A *Threshold* 0.4

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.4, model mencapai akurasi *testing* sebesar 83%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.2

Tabel 4.2 Metrik Evaluasi Skenario Uji 1A *Threshold* 0.4

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	90%	76%	83%
<i>Plastic</i>	79%	91%	84%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.5 menunjukkan bahwa model berhasil memprediksi 162 sampel *paper* dengan benar dan 174 sampel *plastic* dengan benar. Selain itu, terdapat 42 sampel *paper* yang salah diprediksi sebagai *plastic* dan 22 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.5 Gambar 4.4.



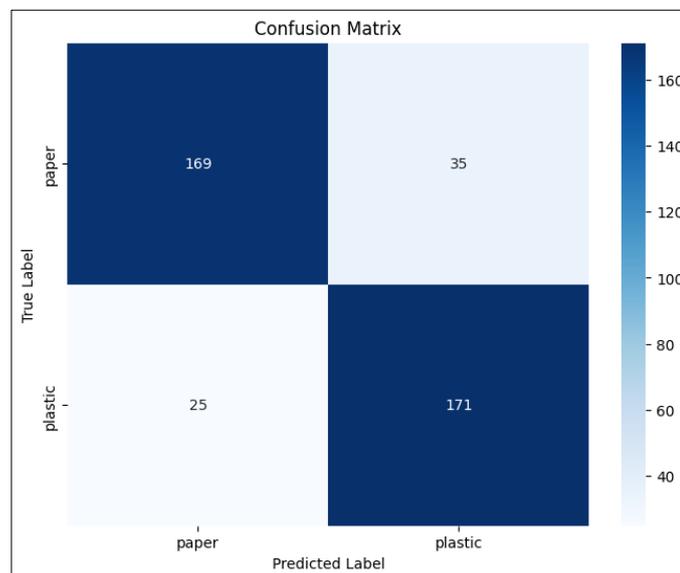
Gambar 4.4 *Confusion Matrix* Skenario Uji 1A *Threshold* 0.5

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.5, model mencapai akurasi *testing* sebesar 84%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.3.

Tabel 4.3 Metrik Evaluasi Skenario Uji 1A *Threshold* 0.5

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	88%	79%	84%
<i>Plastic</i>	81%	89%	84%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.6 menunjukkan bahwa model berhasil memprediksi 169 sampel *paper* dengan benar dan 171 sampel *plastic* dengan benar. Selain itu, terdapat 35 sampel *paper* yang salah diprediksi sebagai *plastic* dan 25 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.6 Gambar 4.5.



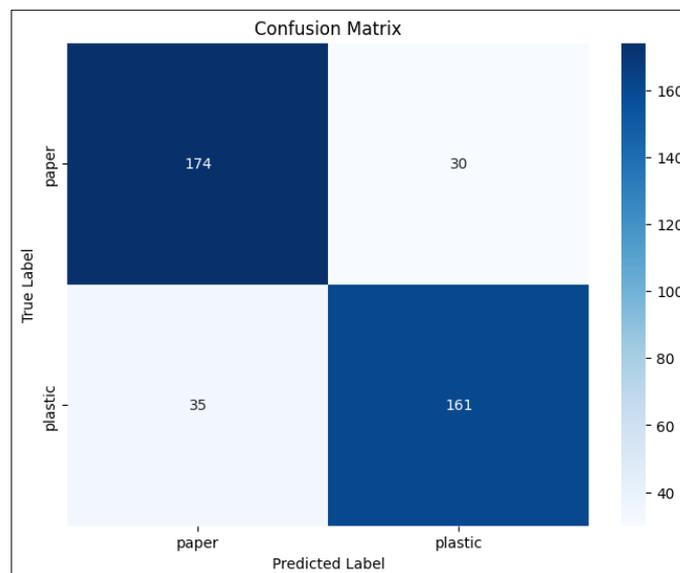
Gambar 4.5 *Confusion Matrix* Skenario Uji 1A *Threshold* 0.6

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.6, model mencapai akurasi *testing* sebesar 85%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.4.

Tabel 4.4 Metrik Evaluasi Skenario Uji 1A *Threshold* 0.6

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	87%	83%	85%
<i>Plastic</i>	83%	87%	85%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.7 menunjukkan bahwa model berhasil memprediksi 174 sampel *paper* dengan benar dan 161 sampel *plastic* dengan benar. Selain itu, terdapat 30 sampel *paper* yang salah diprediksi sebagai *plastic* dan 35 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.7 Gambar 4.6.



Gambar 4.6 *Confusion Matrix* Skenario Uji 1A *Threshold* 0.7

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.7, model mencapai akurasi *testing* sebesar 84%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.5.

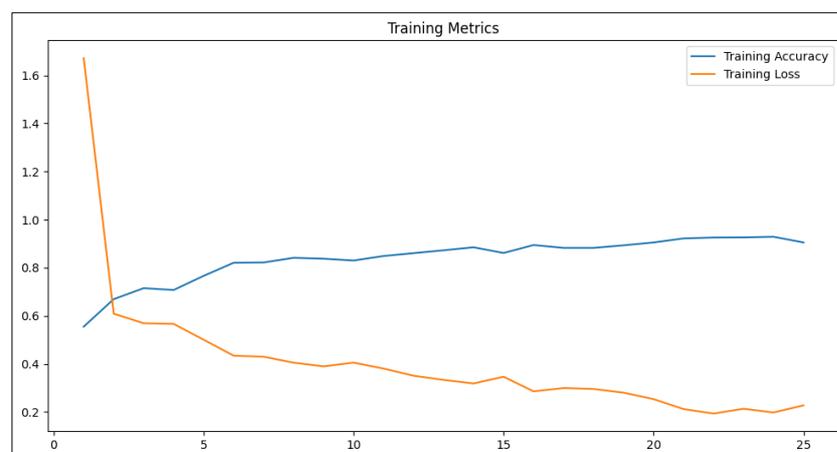
Tabel 4.5 Metrik Evaluasi Skenario Uji 1A *Threshold* 0.7

<b>Kelas</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-score</b>
<i>Paper</i>	83%	85%	84%
<i>Plastic</i>	84%	82%	83%

#### 4.1.2 Skenario Uji 1B

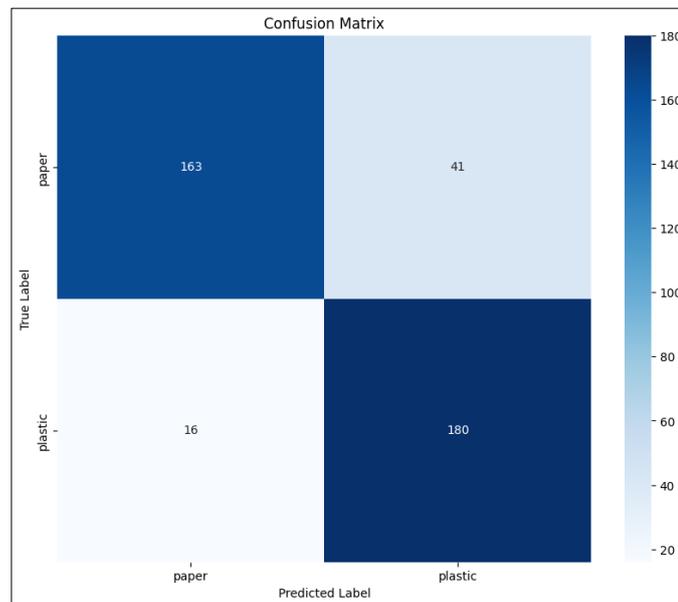
Skenario uji ini dilakukan dengan menggunakan metode klasifikasi *one-hot encoding* tanpa augmentasi citra. Dataset yang digunakan terdiri dari citra sampah *paper* dan *plastic*, dengan total 2000 citra yang dibagi menjadi 1600 data latih dan 400 data uji.

Proses pelatihan model telah dilakukan dan diperoleh nilai *training loss* sebesar 0,3582 dan *training accuracy* sebesar 85,97%. Model mulai menunjukkan konvergensi pada *epoch* 20, ditandai dengan stabilnya nilai *loss* dan *accuracy* tanpa perubahan besar di *epoch* berikutnya. Grafik pelatihan perubahan nilai *loss* dan *accuracy* Gambar 4.7.



Gambar 4.7 Grafik *Training Accuracy* dan *Loss* Skenario Uji 1B

*Confusion matrix* yang dihasilkan dari pengujian menunjukkan bahwa model berhasil memprediksi 163 sampel *paper* dengan benar dan 180 sampel *plastic* dengan benar. Selain itu, terdapat 41 sampel *paper* yang salah diprediksi sebagai *plastic* dan 16 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix* Gambar 4.8.



Gambar 4.8 *Confusion Matrix* Skenario Uji 1B

Pada proses pengujian, model mencapai akurasi *testing* sebesar 86%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.6.

Tabel 4.6 Metrik Evaluasi Skenario Uji 1B

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Paper</i>	91%	80%	85%
<i>Plastic</i>	81%	92%	86%

#### 4.1.3 Skenario Uji 2A

Skenario uji ini dilakukan dengan menggunakan metode klasifikasi *threshold* dan augmentasi citra. Augmentasi citra diterapkan untuk meningkatkan variasi data dan mencegah *overfitting*. Dataset yang digunakan terdiri dari citra sampah *paper*

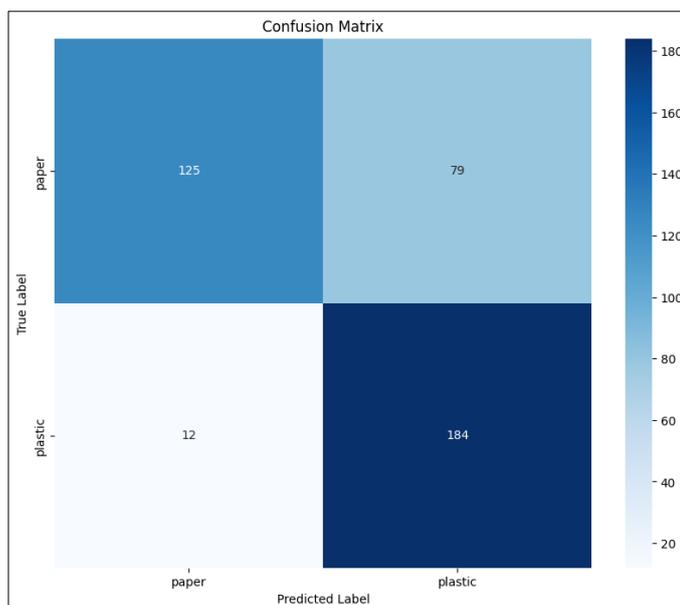
dan *plastic*, dengan total 2000 citra yang terbagi menjadi 1600 untuk pelatihan dan 400 untuk pengujian. Pada proses pelatihan, penerapan augmentasi citra menghasilkan 40.000 variasi citra yang dipelajari oleh model.

Dari proses pelatihan model, diperoleh nilai *training loss* sebesar 0,3694 dan *training accuracy* sebesar 85,54%. Pada *epoch* 20, model telah mencapai konvergensi dengan nilai *loss* dan *accuracy* yang tidak lagi mengalami perubahan signifikan. Grafik pelatihan perubahan nilai *loss* dan *accuracy* Gambar 4.9.



Gambar 4.9 Grafik *Training Accuracy* dan *Loss* Skenario Uji 2A

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.3 menunjukkan bahwa model berhasil memprediksi 125 sampel *paper* dengan benar dan 184 sampel *plastic* dengan benar. Selain itu, terdapat 79 sampel *paper* yang salah diprediksi sebagai *plastic* dan 12 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.3 Gambar 4.10.



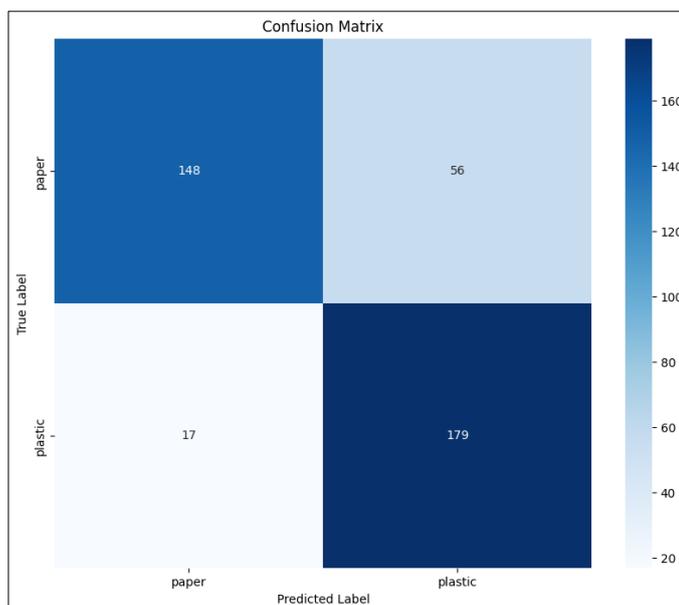
Gambar 4.10 *Confusion Matrix* Skenario Uji 2A *Threshold* 0.3

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.3, model mencapai akurasi testing sebesar 77%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.7.

Tabel 4.7 Metrik Evaluasi Skenario Uji 2A *Threshold* 0.3

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	91%	61%	73%
<i>Plastic</i>	70%	94%	80%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.4 menunjukkan bahwa model berhasil memprediksi 148 sampel *paper* dengan benar dan 179 sampel *plastic* dengan benar. Selain itu, terdapat 56 sampel *paper* yang salah diprediksi sebagai *plastic* dan 17 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.4 Gambar 4.11.



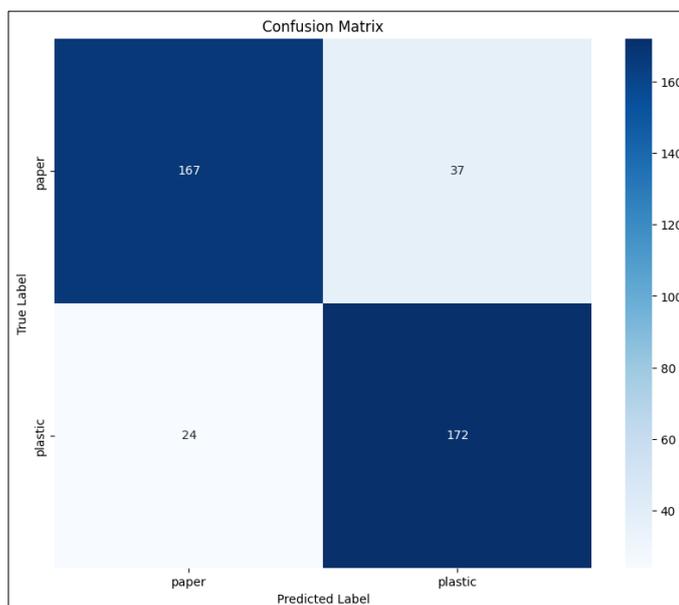
Gambar 4.11 *Confusion Matrix* Skenario Uji 2A *Threshold* 0.4

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.4, model mencapai akurasi testing sebesar 82%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.8.

Tabel 4.8 Metrik Evaluasi Skenario Uji 2A *Threshold* 0.4

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Paper</i>	90%	73%	80%
<i>Plastic</i>	76%	91%	83%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.5 menunjukkan bahwa model berhasil memprediksi 167 sampel *paper* dengan benar dan 172 sampel *plastic* dengan benar. Selain itu, terdapat 37 sampel *paper* yang salah diprediksi sebagai *plastic* dan 24 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.5 Gambar 4.12.

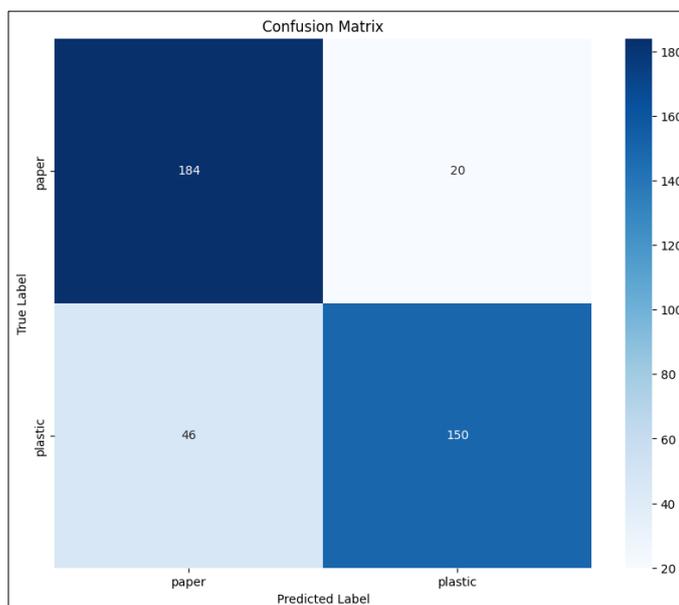
Gambar 4.12 Confusion Matrix Skenario Uji 2A *Threshold* 0.5

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.5, model mencapai akurasi testing sebesar 85%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.9.

Tabel 4.9 Metrik Evaluasi Skenario Uji 2A *Threshold* 0.5

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	87%	82%	85%
<i>Plastic</i>	82%	88%	85%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.6 menunjukkan bahwa model berhasil memprediksi 184 sampel *paper* dengan benar dan 150 sampel *plastic* dengan benar. Selain itu, terdapat 20 sampel *paper* yang salah diprediksi sebagai *plastic* dan 46 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.6 Gambar 4.13.



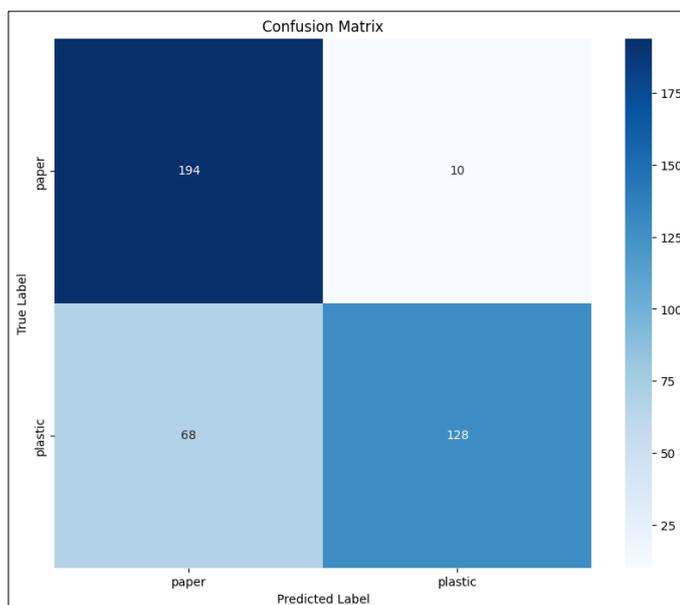
Gambar 4.13 *Confusion Matrix* Skenario Uji 2A *Threshold* 0.6

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.6, model mencapai akurasi testing sebesar 83%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.10.

Tabel 4.10 Metrik Evaluasi Skenario Uji 2A *Threshold* 0.6

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Paper</i>	80%	90%	85%
<i>Plastic</i>	88%	77%	82%

*Confusion matrix* yang dihasilkan dari pengujian menggunakan *threshold* dengan ambang batas 0.7 menunjukkan bahwa model berhasil memprediksi 194 sampel *paper* dengan benar dan 128 sampel *plastic* dengan benar. Selain itu, terdapat 10 sampel *paper* yang salah diprediksi sebagai *plastic* dan 68 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix threshold* 0.7 Gambar 4.14.



Gambar 4.14 *Confusion Matrix* Skenario Uji 2A *Threshold* 0.7

Pada proses pengujian menggunakan *threshold* dengan ambang batas 0.7, model mencapai akurasi testing sebesar 81%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.11.

Tabel 4.11 Metrik Evaluasi Skenario Uji 2A *Threshold* 0.7

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Paper</i>	74%	95%	83%
<i>Plastic</i>	93%	65%	77%

#### 4.1.4 Skenario Uji 2B

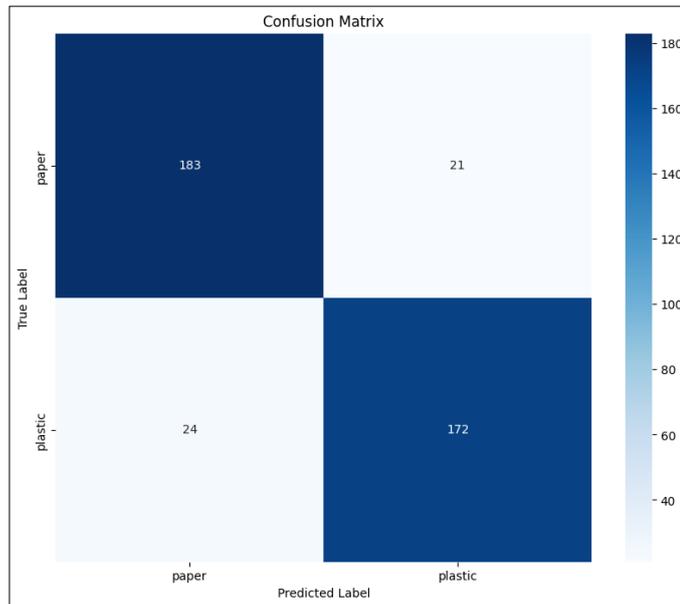
Skenario uji ini dilakukan dengan menggunakan metode klasifikasi *one-hot encoding* dan augmentasi citra. Augmentasi citra diterapkan untuk meningkatkan variasi data dan mencegah *overfitting*. Dataset yang digunakan terdiri dari citra sampah *paper* dan *plastic*, dengan total 2000 citra yang dibagi menjadi 1600 untuk pelatihan dan 400 untuk pengujian. Pada proses pelatihan, penerapan augmentasi citra menghasilkan 40.000 variasi citra yang dipelajari oleh model.

Dari proses pelatihan model, diperoleh nilai *training loss* sebesar 0,3255 dan *training accuracy* sebesar 89,29%. Stabilitas nilai *loss* dan *accuracy* setelah *epoch* 18 menunjukkan bahwa model telah mencapai konvergensi tanpa perubahan signifikan pada *epoch* selanjutnya. Grafik pelatihan perubahan nilai *loss* dan *accuracy* Gambar 4.15.



Gambar 4.15 Grafik *Training Accuracy* dan *Loss* Skenario Uji 2B

*Confusion matrix* yang dihasilkan dari pengujian menunjukkan bahwa model berhasil memprediksi 183 sampel *paper* dengan benar dan 172 sampel *plastic* dengan benar. Selain itu, terdapat 21 sampel *paper* yang salah diprediksi sebagai *plastic* dan 24 sampel *plastic* yang salah diprediksi sebagai *paper*. *Confusion matrix* Gambar 4.16.



Gambar 4.16 *Confusion Matrix* Skenario Uji 2B

Pada proses pengujian, model mencapai akurasi *testing* sebesar 89%. Metrik evaluasi *precision*, *recall*, dan *F1-score* Tabel 4.12.

Tabel 4.12 Metrik Evaluasi Skenario Uji 2B

<b>Kelas</b>	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>
<i>Paper</i>	88%	90%	89%
<i>Plastic</i>	89%	88%	88%

Untuk menunjukkan keberhasilan model dalam mengklasifikasikan data, contoh citra beserta prediksi Tabel 4.13.

Tabel 4.13 Contoh Hasil Klasifikasi Citra

No	Contoh Hasil Klasifikasi Citra
1	 <p>The screenshot shows the 'Trash Classifier' app interface. At the top, it says 'Upload an image...' with a 'Browse Files' button. Below that, it says 'Drag and drop file here' and lists supported file types: 'Limit 20MB per file - JPG, JPEG, PNG'. A file named 'paper.jpg (42.3KB)' is listed. The main image shows a crumpled white paper bag with some text and a logo. Below the image, it says 'Uploaded image'. At the bottom, the prediction is 'Prediction: paper'.</p>
2	 <p>The screenshot shows the 'Trash Classifier' app interface. At the top, it says 'Upload an image...' with a 'Browse Files' button. Below that, it says 'Drag and drop file here' and lists supported file types: 'Limit 20MB per file - JPG, JPEG, PNG'. A file named 'plastic.jpg (34.7KB)' is listed. The main image shows a blue plastic bag. Below the image, it says 'Uploaded image'. At the bottom, the prediction is 'Prediction: plastic'.</p>

## 4.2 Pengujian *K-Fold Cross Validation*

Setelah diperoleh hasil akurasi dari berbagai skenario uji, dilakukan validasi menggunakan teknik *k-fold cross validation* sebanyak 5 *fold* untuk mengukur kemampuan generalisasi model terhadap data yang belum pernah dilatih sebelumnya. Pada pengujian ini, dataset dibagi menjadi 5 *fold*, di mana model dilatih menggunakan 4 *fold* dan diuji pada *fold* yang tersisa. Proses ini diulang 5

kali, dengan 1 *fold* bergiliran menjadi data uji. Hasil pengujian ini menunjukkan akurasi yang diperoleh pada setiap *fold*.

#### 4.2.1 Skenario Uji 1A

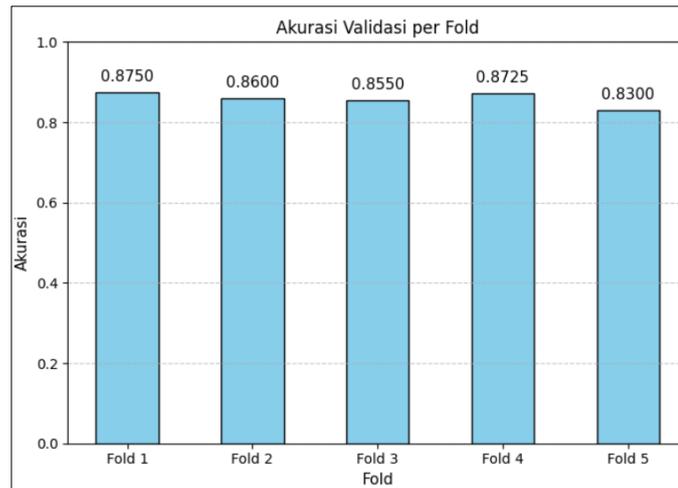
Pada skenario uji 1A, model diuji menggunakan metode klasifikasi *threshold* tanpa augmentasi citra. Pengujian awal menghasilkan akurasi 85%. Untuk mengevaluasi kemampuan model dalam menghadapi variasi data, dilakukan pengujian menggunakan *k-fold cross validation*.

Proses pengujian *k-fold cross validation* dilakukan dengan nilai *k* sebanyak 5. Dataset dibagi menjadi 5 bagian, dan model dijalankan sebanyak 5 kali dengan 1 bagian secara bergiliran dijadikan sebagai data uji dan sisanya sebagai data latih. Hasil akurasi setiap *fold* Tabel 4.14.

Tabel 4.14 Hasil Akurasi Setiap Fold Skenario Uji 1A

<b><i>Fold ke-</i></b>	<b>Akurasi</b>
1	87,50%
2	86,00%
3	85,50%
4	87,25%
5	83,00%
<b>Rata-Rata Akurasi</b>	<b>85,85%</b>

Sebagai visualisasi hasil pengujian menggunakan *k-fold cross validation*, grafik akurasi per *fold* Gambar 4.17.



Gambar 4.17 Grafik Akurasi Setiap Fold Skenario Uji 1A

Berdasarkan hasil pengujian, model menunjukkan hasil akurasi yang konsisten dengan rata-rata 85,85%. Meskipun terdapat sedikit fluktuasi pada akurasi *fold* ke-5 yaitu 83,00%, hasil akurasi lainnya relatif stabil yang menunjukkan bahwa model tidak *overfitting* dan mampu bekerja dengan baik terhadap data baru yang belum pernah dikenali sebelumnya.

#### 4.2.2 Skenario Uji 1B

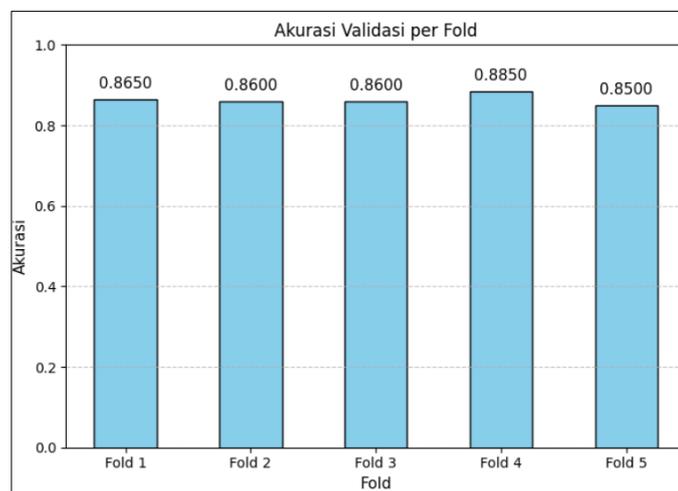
Pada skenario uji 1B, model diuji menggunakan metode klasifikasi *one-hot encoding* tanpa augmentasi citra. Pengujian awal menghasilkan akurasi 86%. Untuk mengevaluasi kemampuan model dalam menghadapi variasi data, dilakukan pengujian dengan pendekatan *k-fold cross validation*.

Evaluasi dengan pendekatan *k-fold cross validation* membagi data ke dalam 5 *fold*. Pengujian dilakukan sebanyak 5 kali, di mana tiap *fold* secara bergantian digunakan sebagai data uji, sedangkan *fold* lainnya digunakan sebagai data latih. Hasil akurasi setiap *fold* Tabel 4.15.

Tabel 4.15 Hasil Akurasi Setiap Fold Skenario Uji 1B

<b>Fold ke-</b>	<b>Akurasi</b>
1	86,50%
2	86,00%
3	86,00%
4	88,50%
5	85,00%
<b>Rata-Rata Akurasi</b>	<b>86,40%</b>

Sebagai visualisasi hasil pengujian dengan pendekatan *k-fold cross validation*, grafik akurasi antar *fold* Gambar 4.18.



Gambar 4.18 Grafik Akurasi Setiap Fold Skenario Uji 1B

Berdasarkan hasil evaluasi, model menunjukkan hasil akurasi yang konsisten dengan rata-rata 86,40%. Meskipun terdapat sedikit fluktuasi pada akurasi *fold* ke-5 yaitu 85,00%, hasil akurasi lainnya relatif stabil yang menunjukkan bahwa model tidak mengalami *overfitting* dan memiliki kemampuan generalisasi yang baik terhadap data yang tidak termasuk dalam proses pelatihan.

### 4.2.3 Skenario Uji 2A

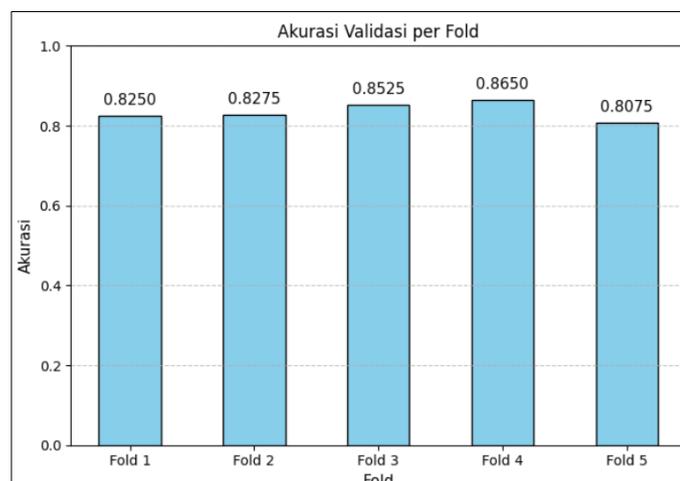
Pada skenario uji 2A, model diuji menggunakan metode klasifikasi *threshold* dan augmentasi citra. Pengujian awal menghasilkan akurasi 85%. Untuk mengevaluasi kemampuan model dalam menghadapi variasi data, dilakukan pengujian dengan menerapkan *k-fold cross validation*.

Pengujian dilakukan dengan teknik *cross validation* dengan  $k$  bernilai 5. Dataset dibagi menjadi 5 bagian dan proses pelatihan serta pengujian dilakukan secara bergantian sebanyak 5 kali. Setiap *fold* berperan sebagai data uji 1 kali, sementara 4 *fold* lainnya menjadi data latih. Hasil akurasi setiap *fold* Tabel 4.16.

Tabel 4.16 Hasil Akurasi Setiap Fold Skenario Uji 2A

<b>Fold ke-</b>	<b>Akurasi</b>
1	82,50%
2	82,75%
3	85,25%
4	86,50%
5	80,75%
<b>Rata-Rata Akurasi</b>	<b>83,55%</b>

Sebagai visualisasi hasil pengujian dengan menerapkan *k-fold cross validation*, grafik akurasi tiap *fold* Gambar 4.19.



Gambar 4.19 Grafik Akurasi Setiap Fold Skenario Uji 2A

Berdasarkan hasil tersebut, model menunjukkan hasil akurasi yang konsisten dengan rata-rata 83,55%. Meskipun terdapat fluktuasi pada akurasi *fold* ke-5 yaitu 80,75%, hasil akurasi lainnya relatif stabil yang menunjukkan bahwa model memiliki performa yang cukup baik dan mampu beradaptasi dengan data baru yang belum pernah dilibatkan dalam pelatihan.

#### 4.2.4 Skenario Uji 2B

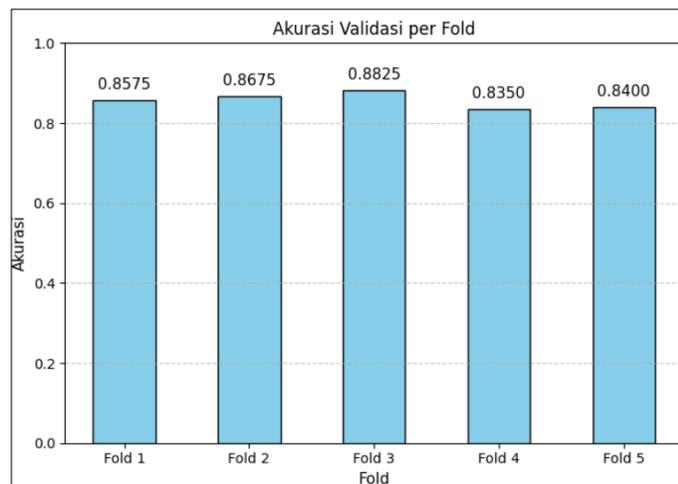
Pada skenario uji 2B, model diuji menggunakan metode klasifikasi *one-hot encoding* dan augmentasi citra. Pengujian awal menghasilkan akurasi 89%. Untuk mengevaluasi kemampuan model dalam menghadapi variasi data, dilakukan pengujian dengan teknik *k-fold cross validation*.

Pengujian dilakukan melalui teknik *k-fold cross validation* dengan nilai *k* sebanyak 5. Dataset dibagi menjadi 5 subset. Pada setiap iterasi, 1 subset dijadikan data uji, sedangkan 4 lainnya digunakan untuk pelatihan. Proses ini diulang hingga seluruh subset digunakan sebagai data uji. Hasil akurasi setiap *fold* Tabel 4.17.

Tabel 4.17 Hasil Akurasi Setiap Fold Skenario Uji 2B

<b><i>Fold</i> ke-</b>	<b>Akurasi</b>
1	85,75%
2	86,75%
3	88,25%
4	83,50%
5	84,00%
<b>Rata-Rata Akurasi</b>	<b>85,65%</b>

Sebagai visualisasi hasil pengujian dengan teknik *k-fold cross validation*, grafik akurasi setiap iterasi Gambar 4.20.



Gambar 4.20 Grafik Akurasi Setiap Fold Skenario Uji 2B

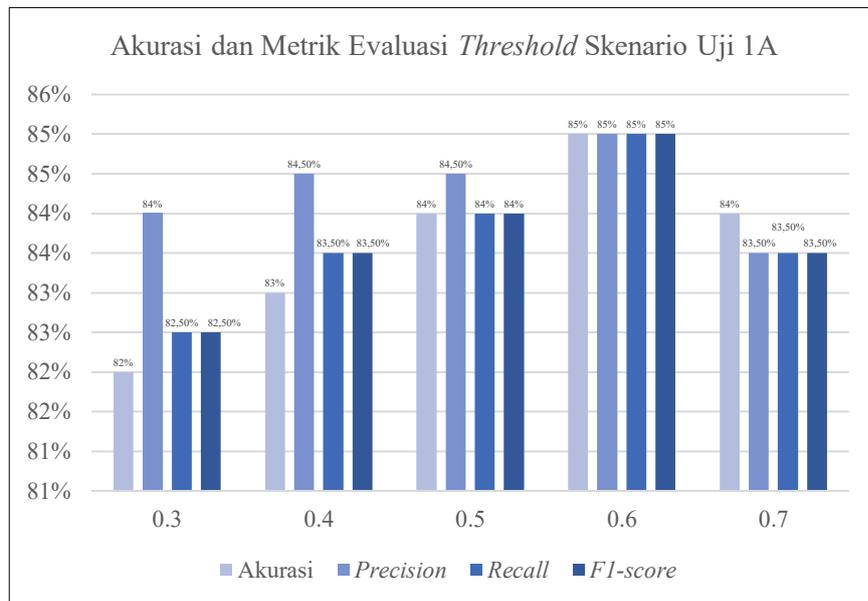
Berdasarkan hasil pengujian tersebut, model menunjukkan hasil akurasi yang konsisten dengan rata-rata 85,65%. Meskipun terdapat fluktuasi pada akurasi *fold* ke-4 yaitu 83,50% dan *fold* ke-5 yaitu 84,00%, hasil akurasi lainnya relatif stabil yang menunjukkan bahwa model memiliki generalisasi yang baik serta mampu mengenali pola baru.

### 4.3 Pembahasan

Dalam penelitian ini, dilakukan berbagai skenario pengujian untuk mengevaluasi efektivitas metode CNN dalam mengklasifikasikan sampah berbasis citra, khususnya pada kategori *paper* dan *plastic*. Analisis dilakukan dengan mengacu pada sejumlah metrik evaluasi seperti akurasi, *precision*, *recall*, dan F1-*score* serta pengaruh dari augmentasi citra dan metode klasifikasi *threshold one-hot encoding* terhadap kinerja model. Pembahasan ini bertujuan untuk mengidentifikasi pola peningkatan atau penurunan performa pada setiap skenario serta menganalisis faktor-faktor yang memengaruhi kemampuan model dalam melakukan klasifikasi dengan lebih akurat dan generalisasi yang lebih baik.

Pada skenario 1A, tanpa augmentasi citra dan menggunakan metode klasifikasi *threshold* dengan ambang batas 0.5, model memperoleh akurasi 84%, rata-rata *precision* mencapai 84,5%, dengan *recall* dan F1-score masing-masing 84%. Meskipun metode klasifikasi *threshold* cukup efektif dalam klasifikasi awal, terdapat keterbatasan dalam menentukan batas probabilitas yang optimal, sehingga model mengalami kesulitan dalam membedakan kelas dengan lebih akurat.

Hasil pengujian lanjutan menunjukkan bahwa pada *threshold* 0.3, model memperoleh akurasi 82%, hasil *precision* 84%, *recall* dan F1-score 82,5%. Pada *threshold* 0.4, terdapat sedikit peningkatan akurasi menjadi 83%, sedangkan *precision* 84,5%, *recall* dan F1-score 83,5%. Selanjutnya pada *threshold* 0.6, model memperoleh akurasi 85%, nilai *precision*, *recall*, dan F1-score yang juga mencapai 85%. Ketika *threshold* dinaikkan ke 0.7, akurasi menurun menjadi 84%, *precision*, *recall*, dan F1-score 83,5%. Peningkatan *threshold* dapat meningkatkan akurasi hingga titik tertentu, namun jika nilainya terlalu tinggi, model menjadi terlalu selektif dalam mengklasifikasikan data, sehingga menurunkan kemampuannya dalam mendeteksi kelas secara umum. Perbandingan akurasi dan rata-rata dari metrik evaluasi pengujian *threshold* skenario 1A Gambar 4.21.



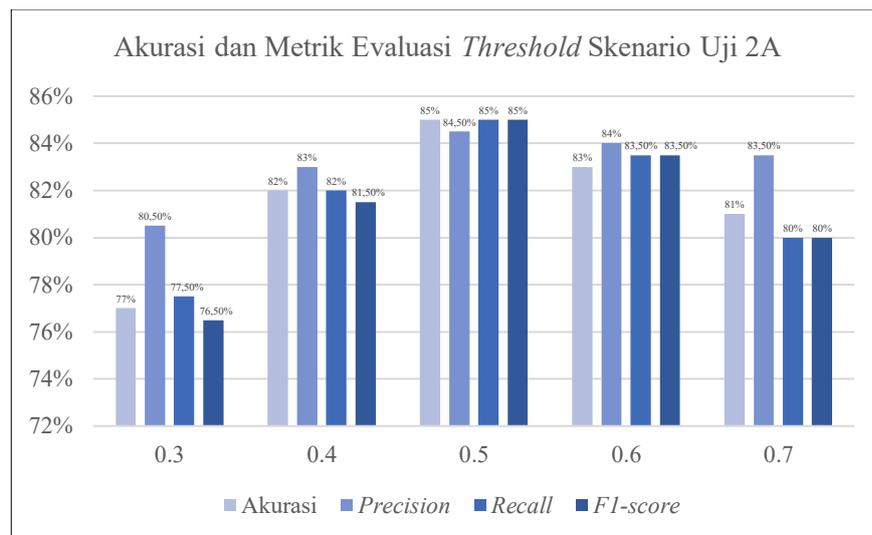
Gambar 4.21 Akurasi dan Metrik Evaluasi *Threshold* Skenario 1A

Ketika metode klasifikasi *one-hot encoding* diterapkan pada skenario 1B, terjadi peningkatan akurasi menjadi 86%, dengan kenaikan rata-rata *precision* dan *recall* 86%, serta *F1-score* 85,5%. Peningkatan ini menunjukkan bahwa metode klasifikasi *one-hot encoding* lebih baik dalam menangkap pola distribusi data dibandingkan metode klasifikasi *threshold*, terutama dalam kondisi tanpa augmentasi citra.

Pada skenario 2A, augmentasi citra diterapkan bersama dengan metode klasifikasi *threshold* dengan ambang batas 0.5, yang menghasilkan akurasi 85%, rata-rata *precision* 84,5%, serta *recall* dan *F1-score* 85%. Akurasi model tidak mengalami kenaikan signifikan dibandingkan skenario 1B, sehingga augmentasi citra pada metode klasifikasi *threshold* tidak cukup untuk meningkatkan akurasi secara keseluruhan.

Pengujian dengan variasi nilai *threshold* juga dilakukan untuk skenario 2A. Pada *threshold* 0.3, model memperoleh akurasi 77%, hasil *precision* 80,5%, *recall*

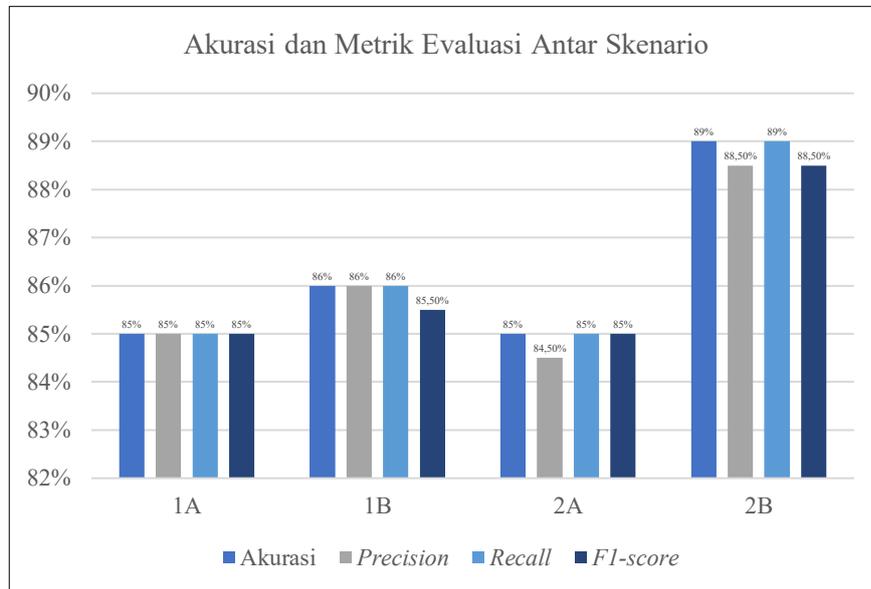
77,5%, dan F1-score 76,5%. Pada *threshold* 0.4, terdapat peningkatan akurasi menjadi 82%, sedangkan *precision* 83%, *recall* 82%, dan F1-score 81,5%. Selanjutnya pada *threshold* 0.6, model memperoleh akurasi 83%, nilai *precision* 84%, *recall* dan F1-score 83,5%. Pada *threshold* 0.7, akurasi kembali menurun menjadi 81%, *precision* 83,5%, *recall* dan F1-score 80%. Peningkatan *threshold* hanya efektif hingga titik tertentu dan nilai *threshold* yang terlalu tinggi justru menurunkan akurasi serta keseimbangan antar metrik evaluasi karena model menjadi terlalu selektif dalam mengklasifikasikan data. Perbandingan akurasi dan rata-rata dari metrik evaluasi pengujian *threshold* skenario 2A Gambar 4.22.



Gambar 4.22 Akurasi dan Metrik Evaluasi *Threshold* Skenario 2A

Skenario 2B yang mengombinasikan augmentasi citra dengan metode klasifikasi *one-hot encoding* memberikan hasil terbaik dengan akurasi 89%, rata-rata *precision* 88,5%, *recall* 89%, dan F1-score 88,5%. Hasil ini menunjukkan bahwa metode klasifikasi *one-hot encoding* lebih optimal dalam mengelola informasi tambahan yang dihasilkan dari augmentasi citra, sehingga mampu meningkatkan generalisasi model terhadap data uji.

Perbandingan akurasi dan rata-rata dari metrik evaluasi antar skenario berdasarkan hasil terbaik masing-masing pengujian *threshold* Gambar 4.23.



Gambar 4.23 Akurasi dan Metrik Evaluasi Antar Skenario

Setelah menganalisis performa model berdasarkan masing-masing skenario, untuk menilai kemampuan model dalam mengenali pola pada data yang bervariasi, digunakan pendekatan *k-fold cross validation* dengan *k*-5. Proses ini bertujuan untuk memastikan bahwa akurasi yang diperoleh tidak hanya tinggi pada pengujian awal, tetapi juga konsisten ketika data dibagi ke dalam skenario pelatihan dan pengujian yang berbeda. Rata-rata akurasi dari masing-masing skenario berdasarkan hasil pengujian *k-fold* Tabel 4.18.

Tabel 4.18 Rata-Rata Akurasi Pengujian *K-Fold* Antar Skenario

Skenario Uji	Rata-Rata Akurasi Pengujian <i>K-Fold</i>
1A	85,85%
1B	86,40%
2A	83,55%
2B	85,65%

Berdasarkan tabel di atas, skenario 1B menghasilkan rata-rata akurasi tertinggi 86,40%, diikuti oleh skenario 2B 85,65%, lalu 1A 85,85%, dan 2A 83,55%. Secara umum, skenario yang menggunakan metode klasifikasi *one-hot encoding* menunjukkan performa yang lebih unggul dibandingkan dengan metode klasifikasi *threshold*, baik tanpa maupun dengan augmentasi citra.

Pengaruh penggunaan augmentasi citra terhadap performa model bervariasi tergantung pada metode klasifikasi yang digunakan. Berdasarkan hasil pengujian, pada skenario 2A, akurasi model sama dengan skenario 1A, yaitu 85%. Namun, *precision* mengalami sedikit penurunan menjadi 84,5%. Setelah dilakukan evaluasi menggunakan *k-fold cross validation*, akurasi skenario 2A menurun menjadi 83,55%, lebih rendah dibandingkan skenario 1A yang mencapai 85,85%. Hal ini menunjukkan bahwa augmentasi citra tidak selalu memberikan dampak positif terhadap performa model, terutama jika dikombinasikan dengan metode klasifikasi *threshold* yang memiliki keterbatasan dalam menangani variasi data. Selanjutnya, pada skenario 2B terjadi peningkatan akurasi 89% dari skenario 1B yaitu 86%. Namun, akurasi tersebut menurun menjadi 85,65% setelah dilakukan evaluasi menggunakan *k-fold cross validation*, yang menunjukkan bahwa meskipun augmentasi citra dapat meningkatkan performa awal, kestabilan model menurun saat dihadapkan pada data yang lebih beragam.

Perbandingan metode klasifikasi menunjukkan bahwa *one-hot encoding* menghasilkan performa yang lebih konsisten dan akurat dibandingkan *threshold*. Skenario 1B memiliki akurasi tertinggi setelah dilakukan evaluasi menggunakan *k-fold cross validation* yaitu 86,40% sekaligus menjadi skenario yang paling stabil di

antara seluruh skenario. Hal ini menunjukkan bahwa *one-hot encoding* mampu merepresentasikan kelas dengan lebih jelas, sehingga model lebih mudah mengenali dan membedakan setiap kelas. Meskipun *threshold* telah diuji pada berbagai nilai ambang batas, akurasi tetap bergantung pada ketepatan pemilihan nilai tersebut. Sementara itu, *one-hot encoding* memberikan pemisahan kelas yang lebih jelas dan mendukung proses pembelajaran model secara lebih efektif. Oleh karena itu, skenario 1B direkomendasikan sebagai pendekatan paling optimal karena mampu mencapai performa tinggi dan stabilitas yang baik, bahkan tanpa bantuan augmentasi citra.

Hasil evaluasi menggunakan *k-fold cross validation* menunjukkan bahwa model CNN cukup *robust* dengan akurasi yang konsisten di setiap *fold*, terutama pada skenario yang menggunakan metode klasifikasi *one-hot encoding*. Skenario 1B menjadi pendekatan terbaik karena menggabungkan kestabilan klasifikasi, performa tinggi, dan konsistensi hasil uji. Hasil ini juga menunjukkan bahwa augmentasi citra tidak selalu meningkatkan performa secara signifikan, terutama jika model dan data sudah cukup kuat dalam mengenali pola fitur secara akurat. Metode klasifikasi *one-hot encoding* efektif dalam memberikan pemetaan kelas yang jelas, sehingga mempermudah model dalam membedakan jenis sampah tanpa menambah kompleksitas data. Oleh karena itu, hasil skenario 1B merupakan pilihan paling seimbang dari segi performa dan kestabilan, serta sangat relevan untuk diterapkan dalam sistem klasifikasi sampah berbasis citra guna mendukung efisiensi pemilahan sampah secara otomatis.

#### 4.4 Integrasi Sains dan Islam

Penelitian ini berfokus pada penerapan CNN dalam mengklasifikasikan sampah berbasis citra untuk mempermudah proses pemilahan sampah, yang diharapkan pengelolaan sampah dapat dilakukan dengan lebih efektif, mendukung proses daur ulang, serta mengurangi pencemaran lingkungan. Penerapan teknologi kecerdasan buatan dalam penelitian ini selaras dengan ajaran Islam yang mengajarkan pentingnya menjaga kelestarian lingkungan dan mencegah kerusakan di muka bumi. Dalam ajaran Islam, kelestarian lingkungan merupakan tanggung jawab setiap individu. Al-Qur'an telah memberikan peringatan mengenai dampak buruk yang timbul akibat kelalaian manusia dalam memelihara keseimbangan alam, sebagaimana termuat dalam surah Ar-Rum ayat 41:

ظَهَرَ الْفَسَادُ فِي الْبَرِّ وَالْبَحْرِ بِمَا كَسَبَتْ أَيْدِي النَّاسِ لِيُذِيقَهُمْ بَعْضَ الَّذِي عَمِلُوا لَعَلَّهُمْ يَرْجِعُونَ

*“Telah nampak kerusakan di darat dan di laut disebabkan karena perbuatan tangan manusia, supaya Allah merasakan kepada mereka sebahagian dari (akibat) perbuatan mereka, agar mereka kembali (ke jalan yang benar).” (QS. Ar-Rum : 41).*

Dalam Tafsir Al-Wajiz, Syaikh Prof. Dr. Wahbah Az-Zuhaili menjelaskan bahwa kerusakan telah nyata terjadi dalam berbagai aspek kehidupan, seperti kekeringan, kebakaran, banjir, wabah penyakit, serta ketidakstabilan dan ancaman dari musuh. Berbagai kerusakan merupakan konsekuensi dari dosa serta tindakan maksiat yang dilakukan oleh manusia. Allah menimpakan sebagian akibat dari perbuatan manusia di dunia agar mereka merasakan dampaknya sebelum mendapat hukuman di akhirat. Tujuannya adalah agar mereka sadar, meninggalkan

kemaksiatan, dan kembali bertaubat atas kesalahan yang telah mereka perbuat (Az-Zuhaili, 1996).

Selain bertujuan untuk mengurangi pencemaran, penelitian ini juga berupaya mengoptimalkan pemanfaatan sumber daya yang ada agar lebih efisien. Dengan adanya klasifikasi otomatis, sampah yang masih dapat didaur ulang bisa lebih mudah dipisahkan dari sampah yang tidak bisa digunakan kembali. Hal ini dapat mendukung prinsip keberlanjutan dalam pengelolaan sampah. Dalam Islam, konsep pemanfaatan sumber daya secara bijak juga ditekankan, di mana manusia diajarkan untuk tidak menyia-nyiakan aungerah dan nikmat yang telah Allah berikan. Prinsip ini ditegaskan dalam surah Luqman ayat 20 yang berbunyi:

أَلَمْ تَرَ أَنَّ اللَّهَ سَخَّرَ لَكُمْ مَّا فِي السَّمٰوٰتِ وَمَا فِي الْأَرْضِ وَأَسْبَغَ عَلَيْكُمْ نِعْمَهُ ظُهْرًا وَبَاطِنًا ۗ وَمِنَ النَّاسِ مَن يُجٰدِلُ فِي اللَّهِ بِغَيْرِ عِلْمٍ وَلَا هُدًى وَلَا كِتٰبٍ مُّنبِئٍ

*“Tidakkah kamu perhatikan sesungguhnya Allah telah menundukkan untuk (kepentingan) mu apa yang di langit dan apa yang di bumi dan menyempurnakan untukmu nikmat-Nya lahir dan batin. Dan di antara manusia ada yang membantah tentang (keesaan) Allah tanpa ilmu pengetahuan atau petunjuk dan tanpa Kitab yang memberi penerangan.” (QS. Luqman : 20).*

Syaikh Abdurrahman bin Nashir As-Sa’di dalam kitabnya yaitu Tafsir Al-Karim Ar-Rahman fi Tafsir Kalam Al-Mannan menjelaskan bahwa Allah mengingatkan manusia akan nikmat-Nya yang melimpah, baik yang tampak maupun tersembunyi. Dia telah menundukkan segala yang ada di langit dan bumi demi kemaslahatan umat manusia, termasuk matahari, bulan, bintang, hewan, tumbuhan, dan sumber daya alam lainnya. Nikmat ini mencakup manfaat duniawi serta petunjuk agama yang membawa kebaikan dan mencegah keburukan. Namun, sebagian manusia ada yang tetap membantah keesaan Allah tanpa ilmu, petunjuk,

atau wahyu yang jelas. Mereka menolak kebenaran yang dibawa para rasul dan lebih memilih mengikuti keyakinan yang tidak berdasar (As-Sa'di, 2002).

Melalui penelitian ini, tidak hanya aspek teknologi yang dikembangkan, tetapi juga nilai-nilai Islam yang mengajarkan pentingnya keseimbangan alam. Dengan memanfaatkan ilmu pengetahuan secara bijak dan bertanggung jawab, manusia dapat berkontribusi dalam upaya pelestarian lingkungan, yang sejalan dengan ajaran Islam mengenai tanggung jawab menjaga alam.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Hasil penelitian ini membuktikan bahwa metode *Convolutional Neural Network* (CNN) efektif dalam melakukan klasifikasi terhadap citra sampah *paper* dan *plastic*. Skenario terbaik diperoleh pada pengujian 1B yang menggabungkan data tanpa augmentasi citra dan metode klasifikasi *one-hot encoding* dengan akurasi 86% rata-rata *precision* 86%, *recall* 86%, dan *F1-score* 88,5%. Berdasarkan hasil pengujian dari empat skenario, model tetap mampu menghasilkan performa yang tinggi meskipun tanpa augmentasi citra dikarenakan data yang digunakan memiliki kualitas dan distribusi kelas yang cukup baik, sehingga model dapat mengenali pola dengan akurat. Selain itu, metode klasifikasi *one-hot encoding* membantu model untuk memahami perbedaan antar kelas dengan lebih jelas dan menangkap pola distribusi data yang kompleks sehingga menghasilkan klasifikasi yang lebih stabil. Kombinasi ini membuktikan bahwa penggunaan metode klasifikasi *one-hot encoding* yang tepat bahkan tanpa augmentasi citra mampu memberikan performa model yang optimal dalam mengklasifikasikan citra sampah.

#### 5.2 Saran

Mengacu pada hasil yang diperoleh serta mempertimbangkan sejumlah keterbatasan dalam penelitian ini, maka beberapa saran yang dapat diberikan untuk pengembangan penelitian selanjutnya sebagai berikut:

1. Menambah variasi augmentasi citra, seperti *shear*, *crop*, dan *vertikal flip* untuk meningkatkan kemampuan model dalam mengenali citra yang bervariasi.
2. Menerapkan arsitektur CNN yang lebih kompleks dengan menambah *convolutional layer* atau arsitektur lain seperti VGG16 dan ResNet guna meningkatkan akurasi dan generalisasi model.

## DAFTAR PUSTAKA

- Al-Mamun, A.-M., Hossain, R., Sharmin, M. M. M., Kabir, E., & Iqbal, M. A. (2023). Garbage classification using convolutional neural networks (CNNs). *Material Science & Engineering International Journal*, 7(3), 140–144. <https://doi.org/10.15406/mseij.2023.07.00217>
- Al-Qarni, A. (2016). *At-Tafsir Al-Muyassar*. Darul Haq.
- Aqilla, A. R., Razak, A., Barlian, E., Syah, N., & Diliarosta, S. (2023). Gudang Jurnal Multidisiplin Ilmu Pengaruh Sampah Plastik Dalam Pencemaran Air. *Gudang Jurnal Multidisiplin Ilmu*, 1(6), 275–280.
- As-Sa'di, A. bin N. (2002). *Taisir Karim ar-Rahman fi Tafsir Kalam al-Mannan*. In *Darussalam* (p. 976). Dar Ibnul Jauzi.
- Az-Zuhaili, P. D. W. (1996). *Tafsiru Ibn 'Athiyah al Muharrir al Wajiz fi Tafsir al Kitab al Karim*. Dar Al-Fikr Beirut.
- Brownlee, J. (2021). *A Gentle Introduction to Threshold-Moving for Imbalanced Classification*. Machine Learning Mastery. <https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/>
- Dacipta, P. N., & Putra, R. E. (2022). Sistem Klasifikasi Limbah Menggunakan Metode Convolutional Neural Network (CNN) Pada Webservice Berbasis Framework Flask. *Journal of Informatics and Computer Science (JINACS)*, 3(04), 394–402. <https://doi.org/10.26740/jinacs.v3n04.p394-402>
- Deslanda, D., Mukti, P., Kaisar, M., Lestari, N. D., & Salma, S. (2023). Sosialisasi Pengolahan Sampah Organik dan Anorganik Menjadi Produk Kreatif. *Madani : Jurnal Ilmiah Multidisiplin*, 1(11), 745–749.
- Hamidah, N., Sinthia, C., & Anshori, M. (2023). Pengaplikasian Komposter Sampah Organik Untuk Pemenuhan Kebutuhan Pupuk Di Desa Palengaan Dajah Kecamatan Palengaan Kabupaten Pamekasan. *Communnity Development Journal*, 04(04), 7980–7991.
- Hutamaputra, W., Krisnabayu, R. Y., Mawarni, M., Yudistira, N., & Bachtiar, F. A. (2022). Perbandingan Convolutional Neural Network VGG16 dan ResNet34 pada Sistem Klasifikasi Sampah Botol. *Jurnal Teknologi Dan Sistem Komputer*, 10(2), 136–142. <https://doi.org/10.14710/jtsiskom.2021.14045>
- Julia Lingga, L., Yuana, M., Aulia Sari, N., Nur Syahida, H., & Sitorus, C. (2024). Sampah di Indonesia: Tantangan dan Solusi Menuju Perubahan Positif. *INNOVATIVE: Journal Of Social Science Research*, 4, 12235–12247.
- Kementerian Lingkungan Hidup dan Kehutanan. (2023). *Capaian Kinerja Pengelolaan Sampah Indonesia*. Kementerian Lingkungan Hidup Dan

Kehutanan. <https://sipsn.menlhk.go.id/sipsn/>

- Kurniawan, R., Wintoro, P. B., Mulyani, Y., & Komarudin, M. (2023). Implementasi Arsitektur Xception Pada Model Machine Learning Klasifikasi Sampah Anorganik. *Jurnal Informatika Dan Teknik Elektro Terapan*, 11(2), 233–236. <https://doi.org/10.23960/jitet.v11i2.3034>
- Lasaiba, M. A. (2024). Strategi Inovatif untuk Pengelolaan Sampah Perkotaan: Integrasi Teknologi dan Partisipasi Masyarakat. *GEOFORUM Jurnal Geografi Dan Pendidikan Geografi*, Vol. 3. No(1), 1–19. <https://doi.org/10.30598/geoforumvol3iss1pp1-18>
- Leonardo, L., Yohannes, Y., & Hartati, E. (2020). Klasifikasi Sampah Daur Ulang Menggunakan Support Vector Machine Dengan Fitur Local Binary Pattern. *Jurnal Algoritme*, 1(1), 78–90. <https://doi.org/10.35957/algoritme.v1i1.440>
- Majida, A. Z., Muzaki, A., Karomah, K., & Awaliyah, M. (2023). Pemanfaatan Sampah Plastik dengan Metode Ecobrick Sebagai Upaya Mengurangi Limbah Plastik. *Profetik: Jurnal Pengabdian Masyarakat*, 1(01), 49–62. <https://doi.org/10.62490/profetik.v1i01.340>
- Mishkin, D., Sergievskiy, N., & Matas, J. (2016). *Systematic evaluation of CNN advances on the ImageNet*. <https://doi.org/10.1016/j.cviu.2017.05.007>
- Nisa, I. Z., Endah, S. N., Sasongko, P. S., Kusumaningrum, R., Khadijah, K., & Rismiyati, R. (2022). Klasifikasi Citra Sampah Menggunakan Support Vector Machine dengan Ekstraksi Fitur Gray Level Co-Occurrence Matrix dan Color Moments. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 9(5), 921–930. <https://doi.org/10.25126/jtiik.2022954868>
- Peryanto, A., Yudhana, A., & Umar, R. (2020). Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation. *Journal of Applied Informatics and Computing*, 4(1), 45–51. <https://doi.org/10.30871/jaic.v4i1.2017>
- Prabhu, D. S. M. (2022). Environmental pollution analysis in developing countries. *International Journal of Statistics and Applied Mathematics*, 7(6), 134–138. <https://doi.org/10.22271/math.2022.v7.i6b.908>
- Suhartono. (2012). *Ilustrasi Pemodelan Pertumbuhan Tanaman Berbasis Lingkungan: Pendekatan Jaringan Saraf Tiruan* (Vol. 75, Issue 8). Kementerian Agama Republik Indonesia.
- Sumartiningtyas, H. K. N. (2020). *Indonesia Hasilkan 64 Juta Ton Sampah, Bisakah Kapasitas Pengelolaan Tercapai Tahun 2025?* Kompas.Com. <https://www.kompas.com/sains/read/2020/12/18/070200023/indonesia-hasilkan-64-juta-ton-sampah-bisakah-kapasitas-pengelolaan?page=all>
- Sunanto, O. D. S., & Utomo, P. H. (2022). Implementasi Deep Learning Dengan Convolutional Neural Network Untuk Klasifikasi Gambar Sampah Organik Dan Anorganik. *Pattimura Proceeding: Conference of Science and*

*Technology*, 1(2), 335–340.  
<https://jurnal.unej.ac.id/index.php/prosiding/article/view/33527>

Sutanty, E., & Kusuma Astuti, D. (2023). *DECODE: Jurnal Pendidikan Teknologi Informasi PENERAPAN MODEL ARSITEKTUR VGG16 UNTUK KLASIFIKASI JENIS SAMPAH*. 3(2), 407–419.

Waluyo, D. (2023). *Masalah Kita Adalah Sampah*. Indonesia.Go.Id. [https://www.indonesia.go.id/kategori/editorial/7714/masalah-kita-adalah-sampah?lang=1#:~:text=Dari total produksi sampah nasional,\(ton\) belum terkelola dengan baik.](https://www.indonesia.go.id/kategori/editorial/7714/masalah-kita-adalah-sampah?lang=1#:~:text=Dari total produksi sampah nasional,(ton) belum terkelola dengan baik.)

World Bank Group. (2022). *Solid Waste Management*. World Bank Group. <https://doi.org/10.1201/9781003189602>

Yan, J. (2024). Application of CNN in computer vision. *Applied and Computational Engineering*, 30(1), 104–110. <https://doi.org/10.54254/2755-2721/30/20230081>

Yujie He, Qinyue Gu, & Maguo Shi. (2020). Trash Classification Using Convolutional Neural Networks Project Category: Computer Vision. *CS230: Deep Learning*.

Zahra, A. N., Novianti, F., Anggraeni, M. P., Normalita, N. R., Sabila, N. S., Ruhmianingsih, N., Azkia, R. H., & Amadea, Z. T. (2024). Analisis Dampak Limbah Galon Plastik di Lingkungan Kos Gang Cempaka Sari, Sekaran, Gunung Pati, Semarang. *Jurnal Analis*, 3(1), 42–56.