

**IMPLEMENTASI ARSITEKTUR RESNET50 PADA  
KLASIFIKASI MOTIF BATIK INDONESIA MENGGUNAKAN  
METODE *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

**LAPORAN TUGAS AKHIR**

Laporan ini Disusun untuk Memenuhi Salah Satu Syarat Memperoleh  
Gelar Sarjana Strata 1 (S1) pada Program Studi Teknik Informatika  
Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang



**DISUSUN OLEH:**

**ARABELA MURI AGISTA**

**NIM 32602100032**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM SULTAN AGUNG  
SEMARANG  
MEI 2025**

***IMPLEMENTATION OF RESNET50 ARCHITECTURE IN  
INDONESIAN BATIK MOTIF CLASSIFICATION USING  
CONVOLUTIONAL NEURAL NETWORK (CNN) METHOD***

***FINAL PROJECT***

*This report is prepared to fulfill one of the requirements to obtain a Bachelor's Degree (S1) in the Informatics Engineering Study Program, Faculty of Industrial Technology, Sultan Agung Islamic University, Semarang*



***MAJORING OF INFORMATICS ENGINEERING  
FACULTY OF INDUSTRIAL TECHNOLOGY  
SULTAN AGUNG ISLAMIC UNIVERSITY  
SEMARANG  
MAY 2025***

LEMBAR PENGESAHAN  
TUGAS AKHIR

IMPLEMENTASI ARSITEKTUR RESNET50 PADA KLASIFIKASI  
MOTIF BATIK INDONESIA MENGGUNAKAN METODE  
CONVOLUTIONAL NEURAL NETWORK (CNN)

ARABELA MURI AGISTA  
NIM 32602100032

Telah dipertahankan di depan tim penguji ujian sarjana tugas akhir  
Program Studi Teknik Informatika  
Universitas Islam Sultan Agung  
Pada tanggal : 2 Juni 2025

TIM PENGUJI TUGAS AKHIR :

Ghufron, ST., M.Kom

NIK. 210622056  
(Ketua Penguji)



02-6-2025

Mustafa, ST., MM., M.Kom

NIK. 210 610 0040  
(Anggota Penguji)



29-5-2025

Dedy Kurniadi, ST., M.Kom

NIK. 210615048  
(Pembimbing)



02-6-2025

Semarang, 2 Juni 2025

Mengetahui,

Kaprodi Teknik Informatika  
Universitas Islam Sultan Agung



Moch. Taufik, ST., MIT

NIK. 210 604 034

## SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini:

Nama : Arabela Muri Agista

NIM : 32602100032

Judul Tugas Akhir : Implementasi Arsitektur ResNet50 pada Klasifikasi Motif Batik Indonesia menggunakan Metode *Convolutional Neural Network* (CNN)

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 2 Juni 2025.

Yang Menyatakan,



Arabela Muri Agista

## PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Arabela Muri Agista

NIM : 32602100032

Program Studi : Teknik Informatika

Fakultas : Teknologi Industri

Alamat Asal : Kabupaten Semarang, Ungaran

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul: Implementasi Arsitektur ResNet50 pada Klasifikasi Motif Batik Indonesia menggunakan Metode *Convolutional Neural Network* (CNN). Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, 2 Juni 2025

Yang menyatakan,



Arabela Muri Agista

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada ALLAH SWT, yang telah memberikan rahmat, taufik serta hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Implementasi Arsitektur Resnet50 pada Klasifikasi Motif Batik Indonesia menggunakan metode *Convolutional Neural Network (CNN)*” ini dengan baik. Dengan penuh rasa hormat, penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT, atas segala rahmat dan karunia-Nya.
2. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, SH., M.Hum yang mengizinkan penulis menimba ilmu di kampus ini.
3. Dekan Fakultas Teknologi Industri Ibu Dr. Ir. Hj. Novi Marlyana, S.T., M.T
4. Dosen pembimbing penulis Bapak Dedy Kurniadi, ST., M.Kom yang telah memberikan arahan, bimbingan, dan saran yang sangat berarti dalam proses penelitian hingga penulisan laporan ini.
5. Dosen-dosen Program Studi Teknik Informatika, Fakultas Teknologi Industri UNISSULA yang telah membekali penulis dengan ilmu dan pengetahuan selama masa studi.
6. Orang tua penulis, Bapak Aris Kasmuri dan Ibu Sunarsih. Terima kasih atas kepercayaan yang telah diberikan kepada penulis, serta pengorbanan, cinta, doa, motivasi, semangat, dan nasihatnya. Dan juga tanpa telah mendukung segala keputusan dalam hidup penulis.
7. Kakak penulis, Niken Fitri Aristya Kurniasih, S.I.P. yang selalu memberikan dukungan moril dan material, memotivasi, dan mendoakan penulis.
8. Teman-teman dan rekan seperjuangan, yang telah memberikan moral, motivasi, serta semangat dalam proses penyelesaian tugas akhir ini.
9. Dan kepada seluruh pihak yang tidak dapat saya sebutkan satu persatu.

Semarang, 5 Mei 2025

Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	<b>i</b>
<b>LEMBAR PENGESAHAN TUGAS AKHIR</b> .....	<b>iii</b>
<b>KATA PENGANTAR</b> .....	<b>vi</b>
<b>DAFTAR ISI</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>ix</b>
<b>DAFTAR GAMBAR</b> .....	<b>x</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xi</b>
<b>ABSTRAK</b> .....	<b>xii</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah.....	3
1.3 Pembatasan Masalah .....	3
1.4 Tujuan.....	4
1.5 Manfaat.....	4
1.6 Sistematika Penulisan.....	5
<b>BAB II TINJAUAN PUSTAKA DAN DASAR TEORI</b> .....	<b>6</b>
2.1 Tinjauan Pustaka.....	6
2.2 Dasar Teori .....	9
2.2.1 Batik dan Klasifikasinya .....	9
2.2.2 <i>Deep Learning</i> .....	9
2.2.3 <i>Convolutional Neural Network (CNN)</i> .....	10
2.2.4 ResNet50 .....	13
<b>BAB III METODE PENELITIAN</b> .....	<b>16</b>
3.1 Pengumpulan Data .....	17
3.2 <i>Preprocessing Data</i> .....	17
3.3 Pelatihan Model.....	18

3.4	Evaluasi Model.....	19
3.5	Implementasi Model.....	23
<b>BAB IV HASIL DAN ANALISIS PENELITIAN.....</b>		<b>24</b>
4.1	HASIL PENELITIAN .....	24
4.1.1	Pengumpulan Data .....	24
4.1.2	<i>Preprocessing Data</i> .....	27
4.1.3	Pelatihan Model .....	29
4.2	ANALISA PENELITIAN .....	33
4.2.1	Evaluasi Model.....	33
4.2.2	Hasil Implementasi Model.....	43
<b>BAB V KESIMPULAN DAN SARAN.....</b>		<b>52</b>
5.1	Kesimpulan.....	52
5.2	Saran.....	53
<b>DAFTAR PUSTAKA</b>		



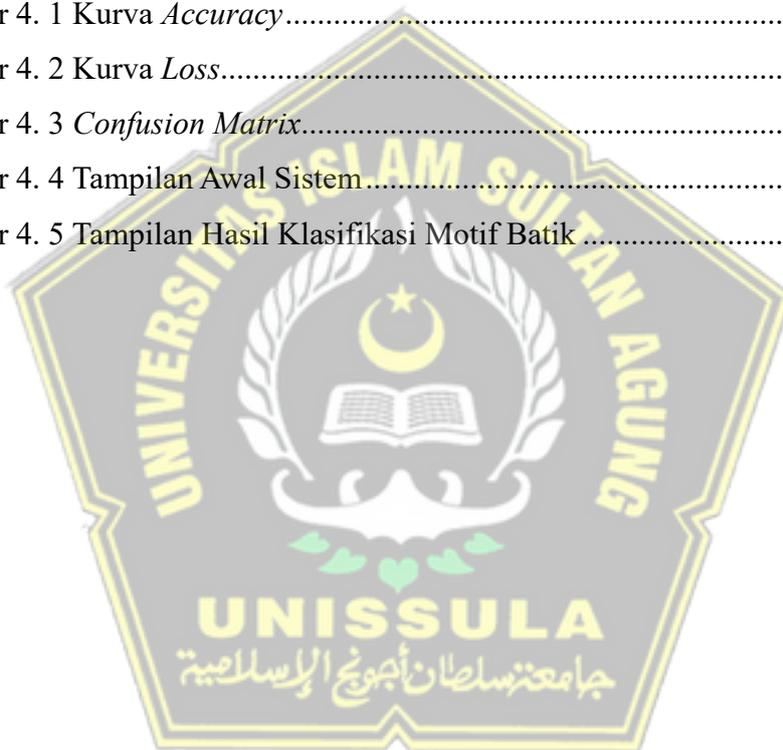
## DAFTAR TABEL

Tabel 4. 1 Sampel Citra Motif Batik .....	26
Tabel 4. 2 <i>Hyperparameter</i> Proses Pelatihan .....	32
Tabel 4. 3 (Lanjutan) .....	33
Tabel 4. 4 Akurasi dan <i>Loss Validation</i> .....	38
Tabel 4. 5 <i>Classification Report</i> .....	39
Tabel 4. 6 Hasil Pengujian Model .....	42
Tabel 4. 7 <i>Hyperparameter Flask</i> .....	45
Tabel 4. 8 (Lanjutan) .....	46
Tabel 4. 9 (Lanjutan) .....	47



## DAFTAR GAMBAR

Gambar 2. 1 Arsitektur CNN (ANHAR & PUTRA, 2023) .....	11
Gambar 2. 2 Operasi Konvolusi.....	12
Gambar 2. 3 Operasi <i>Zero Padding</i> 2 pada Data 3x3 .....	12
Gambar 2. 4 Arsitektur ResNet50 (Agustina, 2024).....	14
Gambar 3. 1 Alur Pembuatan Sistem .....	16
Gambar 3. 2 Alur <i>Preprocessing Data</i> .....	17
Gambar 4. 1 Kurva <i>Accuracy</i> .....	34
Gambar 4. 2 Kurva <i>Loss</i> .....	36
Gambar 4. 3 <i>Confusion Matrix</i> .....	40
Gambar 4. 4 Tampilan Awal Sistem.....	49
Gambar 4. 5 Tampilan Hasil Klasifikasi Motif Batik.....	50



## DAFTAR LAMPIRAN

Lampiran 1. Logbook Bimbingan Pra Sidang.....	58
Lampiran 2. Lembar Revisi Penguji 1 .....	59
Lampiran 3. Lembar Revisi Penguji 2 .....	60



## ABSTRAK

Batik merupakan salah satu bentuk kekayaan budaya Indonesia yang memiliki keragaman corak serta nilai filosofis yang mendalam di setiap wilayah. Meski demikian, proses identifikasi dan pengelompokan motif batik secara manual masih memerlukan keahlian khusus dan memakan waktu yang tidak sedikit. Penelitian ini bertujuan untuk menerapkan arsitektur ResNet50 dalam perancangan model pengenalan motif batik Indonesia dengan menggunakan pendekatan *Convolutional Neural Network* (CNN). Model dilatih untuk membedakan empat kategori motif batik, yaitu Batik Corak Insang, Batik Dayak, Batik Ikat Celup, dan Batik Megamendung. Hasil eksperimen menunjukkan bahwa model mampu mencapai tingkat akurasi sebesar 81%, dengan nilai akurasi validasi sebesar 80,68% dan loss validasi sebesar 65,94%. Di antara semua kelas, model memiliki performa terbaik dalam mengklasifikasikan motif Corak Insang dan motif Dayak. Model ini dikembangkan ke dalam bentuk aplikasi *web* yang memungkinkan pengguna mengunggah gambar batik dan menerima hasil prediksi secara instan. Berdasarkan hasil yang diperoleh, dapat disimpulkan bahwa arsitektur ResNet50 yang dikombinasikan dengan CNN dapat digunakan secara efektif dalam membantu proses identifikasi motif batik secara otomatis, serta berkontribusi dalam upaya digitalisasi dan pelestarian budaya lokal melalui teknologi kecerdasan buatan.

Kata kunci: Batik, Jaringan Syaraf Konvolusional, ResNet50, Pengklasifikasian Citra, Digitalisasi Budaya

## ABSTRACT

*Batik is one of Indonesia's cultural riches, characterized by diverse patterns and deep philosophical values rooted in each region. However, the manual process of identifying and classifying batik motifs still requires specific expertise and is often time-consuming. This study aims to implement the ResNet50 architecture in designing a model for recognizing Indonesian batik motifs using a Convolutional Neural Network (CNN) approach. The model is trained to differentiate four batik motif categories: Batik Corak Insang, Batik Dayak, Batik Ikat Celup, and Batik Megamendung. Experimental results show that the model achieves an accuracy rate of 81%, with a validation accuracy of 80,68% and a validation loss of 66,94%. Among all classes, the model performs best in classifying Corak Insang and Dayak motifs. The trained model is deployed in a web-based application that enables users to upload batik images and receive instant classification results. Based on these outcomes, it can be concluded that the ResNet50 architecture, when integrated with CNN, can be effectively utilized to support automatic batik motif recognition and contribute to the digitalization and preservation of local cultural heritage through artificial intelligence technology.*

*Keywords: Batik, Convolutional Neural Network, ResNet50, Image Classification, Cultural Digitalization*

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dalam buku Ensiklopedia the Heritage of Batik, Identitas Pemersatu Kebanggaan Bangsa oleh (Supriono, 2016). Batik merupakan kekayaan budaya Indonesia yang bersifat adiluhung dan diwariskan oleh nenek moyang secara turun-temurun sejak zaman dahulu. Seni batik telah melintasi waktu mengiringi perjalanan panjang bangsa. Ia juga telah mengarungi pergulatan kebudayaan yang sangat penting. Wajar jika batik kini hadir sebagai salah satu identitas bangsa yang menyatukan sekaligus membanggakan.

Batik telah diakui oleh UNESCO sebagai *Representative List of the Intangible Cultural Heritage of Humanity* pada 2 Oktober 2009. Tanggal tersebut kemudian diperingati sebagai Hari Batik Nasional dengan dikeluarkannya Keputusan Presiden Republik Indonesia Nomor 33 Tahun 2009 Tentang Hari Batik Nasional tanggal 17 November. Salah satu alasan penetapan ini adalah untuk meningkatkan kesadaran masyarakat terhadap upaya perlindungan dan pengembangan batik Indonesia (Yudi Aprianingrum dan Hayati Nufus, 2021).

Seiring perkembangan teknologi digital, muncul kebutuhan akan sistem cerdas yang mampu mengidentifikasi objek secara otomatis dan presisi, termasuk dalam konteks pelestarian budaya visual seperti batik. Digitalisasi motif batik melalui proses klasifikasi citra berbasis kecerdasan buatan menjadi solusi potensial untuk mendokumentasikan ribuan jenis motif yang tersebar di seluruh Nusantara. Proses klasifikasi manual oleh manusia dinilai kurang efisien karena membutuhkan waktu yang lama, tenaga ahli, dan rentan terhadap subjektivitas dalam penilaian motif.

Pada artikel (Alfarizi dkk., 2023), dalam konteks yang lebih luas, *deep learning* adalah cabang dari *machine learning* yang berfokus pada pemanfaatan jaringan syaraf tiruan (JST) dengan arsitektur yang lebih kompleks. Teknologi ini dirancang untuk memungkinkan komputer berpikir dan mengambil keputusan layaknya manusia. Dalam jurnal (Af'idah dkk., 2021) Pada *deep learning*,

komputer dilatih menggunakan kumpulan data berlabel dalam jumlah besar. Data ini kemudian dikonversi ke dalam representasi internal, seperti vektor fitur, berdasarkan nilai piksel gambar. Proses ini memungkinkan komputer untuk melakukan pengenalan pola dan pengklasifikasian yang akurat, baik dalam data gambar, suara, teks, maupun video.

Kemajuan dalam bidang computer vision telah memungkinkan pengembangan sistem berbasis deep learning yang mampu mengenali pola visual dengan akurasi tinggi. CNN, sebagai salah satu metode andalan dalam pengolahan citra, telah banyak diaplikasikan dalam berbagai domain seperti pengenalan wajah, klasifikasi objek, diagnosis medis berbasis citra, dan juga pengenalan karya seni. Dalam penelitian ini, dirancang sebuah sistem yang bertujuan untuk melakukan pengklasifikasian motif citra batik Indonesia dengan memanfaatkan algoritma *Convolutional Neural Network* (CNN). Algoritma ini dipilih karena kemampuannya yang unggul dalam mengenali pola, khususnya pada pengklasifikasian gambar seperti motif batik dijelaskan pada (Pendahuluan, 2008).

Salah satu arsitektur CNN yang telah terbukti unggul dalam banyak tugas klasifikasi citra adalah ResNet50. ResNet50 mampu mengekstraksi informasi visual yang kompleks dari gambar, menjadikannya pilihan ideal untuk tugas klasifikasi motif batik yang memiliki detail dan kompleksitas visual tinggi. Sistem ini dirancang untuk mengidentifikasi dan mengklasifikasikan berbagai jenis motif batik, sehingga dapat membantu dalam pelestarian budaya sekaligus mendukung digitalisasi identitas budaya Indonesia (Peryanto dkk., 2020).

Penggunaan ResNet50 dalam klasifikasi motif batik bukan hanya bertujuan untuk mendemonstrasikan kemampuan teknis dari algoritma deep learning, tetapi juga memberikan kontribusi nyata terhadap upaya konservasi budaya. Dengan tersedianya sistem klasifikasi otomatis, masyarakat luas, pengrajin batik, peneliti, dan lembaga kebudayaan dapat mengakses data motif batik secara lebih praktis dan terstruktur. Hal ini juga dapat dimanfaatkan untuk edukasi, promosi, dan pengembangan industri kreatif berbasis warisan budaya.

Penelitian ini juga bertujuan untuk memperluas wawasan dan memperkaya kajian ilmiah mengenai implementasi CNN dalam pengklasifikasian motif batik.

Diharapkan, penggunaan CNN dalam sistem ini mampu mengenali berbagai motif batik Indonesia secara akurat dan efisien. Dengan memanfaatkan metode deep learning yang terkenal karena kemampuannya dalam pengolahan citra, hasil sistem ini diharapkan dapat memberikan performa yang optimal, dengan tingkat akurasi yang baik dan hasil yang konsisten (Saputra dkk., 2023).

Dengan demikian, integrasi antara teknologi CNN dan digitalisasi motif batik Indonesia melalui arsitektur ResNet50 diharapkan mampu menjadi fondasi awal dalam pembangunan sistem klasifikasi budaya visual yang lebih luas di masa depan. Penelitian ini tidak hanya memiliki nilai teknis dalam bidang kecerdasan buatan, tetapi juga nilai strategis dalam menjaga dan mewariskan identitas budaya bangsa Indonesia secara digital dan berkelanjutan.

## 1.2 Perumusan Masalah

Merujuk pada penjabaran latar belakang yang telah dikemukakan, permasalahan pokok yang diidentifikasi dalam studi ini adalah:

1. Bagaimana performa arsitektur ResNet50 dan algoritma CNN dalam mengklasifikasikan secara akurat berbagai motif batik Indonesia?
2. Bagaimana cara merancang dan mengimplementasikan model ResNet50 dan CNN untuk mengklasifikasikan model batik Indonesia secara akurat?

## 1.3 Pembatasan Masalah

Penelitian “Implementasi Arsitektur Resnet50 pada Klasifikasi Motif Batik Indonesia menggunakan Metode *Convolutional Neural Network* (CNN)” memiliki beberapa batasan, antara lain:

1. Penelitian ini hanya berfokus pada beberapa jenis motif batik yang berasal dari Indonesia, yaitu motif batik Corak Insang, motif batik Dayak, motif batik Ikat Celup, dan motif batik Megamendung.
2. Penelitian ini tidak bertujuan untuk membandingkan hasil klasifikasi yang diperoleh dari model CNN dengan hasil dari metode klasifikasi lain, seperti *Support Vector Machine* (SVM) atau *K-Nearest Neighbor* (KNN).

3. Penelitian ini menggunakan arsitektur ResNet50 sebagai dasar model tanpa melakukan perbandingan dengan arsitektur CNN lainnya, seperti VGG16, InceptionV3, atau MobileNet.

#### **1.4 Tujuan**

Penelitian “Implementasi Arsitektur Resnet50 pada Klasifikasi Motif Batik Indonesia menggunakan Metode *Convolutional Neural Network* (CNN)” memiliki beberapa batasan, antara lain:

1. Merancang, mengembangkan, dan mengimplementasikan model CNN yang dapat mengklasifikasikan motif batik berdasarkan gambar yang diberikan.
2. Menganalisis seberapa akurat model CNN dapat mengenali dan mengklasifikasikan berbagai jenis motif batik.

#### **1.5 Manfaat**

Diharapkan hasil dari penelitian ini dapat memberikan manfaat bagi:

1. Penelitian ini memberikan kontribusi terhadap pengembangan ilmu pengetahuan di bidang kecerdasan buatan, khususnya dalam penerapan arsitektur deep learning seperti ResNet50 untuk klasifikasi citra.
2. Sistem klasifikasi motif batik yang dihasilkan dapat dimanfaatkan oleh pelaku industri kreatif, khususnya di bidang fashion dan kerajinan, untuk mengenali dan mengelompokkan motif batik secara otomatis dan efisien.
3. Dapat membantu dalam upaya digitalisasi dan dokumentasi motif batik Indonesia secara sistematis, sehingga mendukung pelestarian budaya lokal melalui teknologi.

## 1.6 Sistematika Penulisan

Sistematika penulisan yang akan digunakan oleh penulis dalam pembuatan laporan tugas akhir adalah sebagai berikut:

### **BAB I : PENDAHULUAN**

Pada BAB I menjelaskan tentang latar belakang, pemilihan judul, rumusan masalah, batasan masalah, tujuan penelitian, dan sistematika penulisan.

### **BAB II : TINJAUAN PUSTAKA DAN DASAR TEORI**

Pada BAB II memuat dasar teori yang berfungsi sebagai sumber atau alat dalam memahami permasalahan.

### **BAB III : METODE PENELITIAN**

Pada BAB III menjelaskan tentang proses tahapan- tahapan penelitian dimulai dari analisa kebutuhan sistem, kemudian perancangan sistem.

### **BAB IV : HASIL DAN ANALISIS PENELITIAN**

Pada BAB IV menjelaskan hasil dari penelitian yang telah dilakukan, yakni pembuatan sistem yang telah dibuat.

### **BAB V : KESIMPULAN DAN SARAN**

Pada BAB V memuat kesimpulan dari keseluruhan uraian bab-bab sebelumnya dan saran-saran dari hasil yang diperoleh dan diharapkan dapat bermanfaat dalam penelitian selanjutnya.

## BAB II

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1 Tinjauan Pustaka

Untuk mendukung penelitian ini, penulis merujuk pada berbagai penelitian yang ada keterkaitan dengan penelitian yang akan diambil baik secara langsung maupun tidak langsung, penelitian tersebut antara lain:

Berdasarkan hasil pengujian yang telah dilakukan dalam penelitian “Implementasi Algoritma *Convolutional Neural Network* (CNN) Untuk Klasifikasi Batik” sudah berhasil diterapkan. Berdasarkan evaluasi terhadap 200 data uji dengan 20 kategori label, akurasi tertinggi dicapai oleh 'Batik Megamendung' dan 'Batik Celup' masing-masing dengan persentase 80% dan 60%. Akurasi maksimum yang diperoleh selama proses pelatihan model selama 200 epoch mencapai 90%, sedangkan rata-rata tingkat akurasi keseluruhan tercatat sebesar 25,5% (Rizal dkk., 2022).

Penerapan algoritma CNN juga dilakukan untuk mengklasifikasikan gambar mata. Pada penelitian tersebut algoritma CNN menunjukkan hasil yang lebih unggul dibandingkan penelitian terdahulu. Hal ini terbukti dari presentase akurasi yang diperoleh, yakni sebesar 94%. Nilai precision untuk kategori *emaleeyes* mencapai 97%, sedangkan untuk *maleeyes* sebesar 90%. Adapun nilai *recall*-nya masing-masing sebesar 91% untuk *femaleeyes* dan 96% untuk *maleeyes*. Untuk *F1-score*, didapatkan nilai 94% pada *femaleeyes* dan 93% pada *maleeyes* (Firdaus dkk., 2022).

Pemanfaatan algoritma CNN juga dilakukan untuk mengklasifikasikan buah. Pada penelitian ini didapatkan hasil akurasi 96,87%, sedangkan klasifikasi buah menggunakan algoritma SVM mendapatkan akurasi 93,09%. Berdasarkan hasil akurasi yang diperoleh, dapat disimpulkan bahwa model klasifikasi CNN ialah model dengan tingkat ketepatan tertinggi dalam mengklasifikasikan gambar buah, yaitu mencapai 96,87%. Algoritma CNN ini menerapkan proses *subsampling* menggunakan teknik *max pooling* serta pelatihan data dilakukan selama 25 *epoch* (Kurniadi dkk., 2021).

Penggunaan algoritma CNN juga dilakukan dalam mengembangkan sistem identifikasi daun tanaman. Kajian tersebut diperoleh tingkat akurasi tertinggi sebesar 94% dengan nilai kesalahan (loss) minimum sebesar 0,28. Sementara itu, hasil uji (testing) menghasilkan akurasi sebesar 84% dari total 1.650 citra daun herbal, di mana 1.382 gambar berhasil diklasifikasikan ke dalam kategori yang tepat dan 268 gambar teridentifikasi dalam kategori yang keliru. Berdasarkan tahapan yang telah dilakukan, dapat disimpulkan bahwa kinerja algoritma CNN sangat dipengaruhi oleh keseimbangan distribusi data antar kelas dan kuantitas data pelatihan yang mencukupi (Pujiati dan Rochmawati, 2022).

CNN juga dapat diterapkan untuk mengidentifikasi motif kain tenun. Pada kajian tersebut memperlihatkan kinerja yang sangat optimal dalam mendeteksi citra motif kain tenun khas Sumbawa, dengan capaian akurasi tertinggi sebesar 98,14% pada pelatihan selama 20 hingga 25 epoch dalam durasi waktu kurang dari satu jam. Tingkat akurasi pada data pelatihan mencapai 99,71%, disertai nilai kerugian (loss) sebesar 12,99%, yang mengindikasikan bahwa model memiliki kemampuan yang sangat baik dalam mengenali pola pada data pelatihan dan menunjukkan tingkat kesalahan yang relatif rendah (Dianda dkk., 2025).

Metode CNN berhasil diaplikasikan untuk klasifikasi citra batik tanah liat dari Sumatera Barat menggunakan pustaka Keras dan TensorFlow dengan bahasa pemrograman Python. Pada proses pelatihan data latih, diperoleh akurasi sebesar 98,75%, sedangkan pada tahap pengujian data uji, akurasi yang dicapai adalah 62,5%. Tingkat akurasi tersebut tergolong cukup baik dan sudah layak dijadikan sebagai acuan dalam pengembangan aplikasi nyata untuk pengenalan motif batik atau objek lainnya secara umum (Azmi dkk., 2023).

Salah satu kajian terdahulu yang mengimplementasikan metode CNN dan arsitektur ResNet50 yaitu untuk mengklasifikasikan jenis burung. Berdasarkan penelitian yang telah dilakukan, arsitektur model ResNet-50 efektif digunakan dalam proses klasifikasi tersebut. Dengan menerapkan *optimizer* seperti SGD dan Adam, model ini berhasil mencapai tingkat akurasi antara 97% hingga 98%, yang diperoleh melalui evaluasi menggunakan *Confusion Matrix* pada berbagai skenario

yang diuji. Secara keseluruhan, model menunjukkan rata-rata akurasi yang tinggi, yakni sebesar 97% (Alberto dan Hermanto, 2023).

Pemanfaatan arsitektur ResNet50 juga dilakukan untuk mengklasifikasikan tumor otak. Berdasarkan hasil studi yang telah dilaksanakan, penggunaan metode CNN dengan arsitektur ResNet50 pada dataset tumor otak yang terdiri dari empat kategori, *glioma\_tumor*, *meningioma\_tumor*, *pituitary\_tumor*, dan *normal*—dengan pembagian data sebesar 80:20, menghasilkan akurasi tertinggi sebesar 96% yang dicapai pada *epoch* ke-11 (Septipalan dkk., 2024).

Penerapan arsitektu ResNet50 juga dimanfaatkan dalam mengklasifikasikan tumor otak. Model ResNet memiliki keunggulan dalam mengekstraksi pola yang kompleks dari data, namun tingkat akurasinya masih berada di bawah model VGG12 dan InceptionNet. Dengan akurasi mencapai 94,34%, model ini menampilkan kinerja yang cukup baik, meskipun nilai presisi dan recall yang cenderung fluktuatif menjadikan performa prediksinya kurang konsisten dibandingkan kedua model lainnya. Di sisi lain, model AlexNet mencatatkan akurasi paling rendah, yakni sebesar 96,70%, meskipun tetap menunjukkan keunggulan dari segi efisiensi waktu pelatihan. Model ini dinilai sesuai untuk implementasi pada sistem dengan keterbatasan daya komputasi, namun kurang tepat untuk digunakan dalam tugas klasifikasi yang menuntut tingkat akurasi serta konsistensi yang tinggi (Dewi dkk., 2025).

Pada penelitian untuk mengklasifikasikan gambar dagung sapi juga menerapkan arsitektur ResNet50. Hasil penelitian menunjukkan bahwa optimizer SGD merupakan yang terbaik dengan tingkat akurasi mencapai 97,83%, presisi 97%, recall 97%, dan nilai F1 sebesar 97% pada pengaturan batch size 32, learning rate 0,01, dan epoch 50. Selanjutnya, optimizer RMSprop menempati posisi kedua dengan akurasi 88%, presisi 90%, recall 83%, dan F1 score 86% pada batch size 32, learning rate 0,001, dan epoch 50. Sedangkan optimizer Adam berada di peringkat ketiga dengan akurasi 87,83%, presisi 94%, recall 78%, dan F1 score 85% pada batch size 64, learning rate 0,001, dan epoch 50 (Algoritma dkk., 2022).

## 2.2 Dasar Teori

### 2.2.1 Batik dan Klasifikasinya

Batik merupakan salah satu warisan budaya yang sangat berharga bagi Indonesia dan telah diakui oleh UNESCO sebagai warisan budaya non-fisik sejak tahun 2009. Kain batik tidak hanya mengandung nilai filosofis yang mendalam, tetapi juga memiliki nilai estetika yang tinggi serta potensi ekonomi yang besar. Selain itu, batik mencerminkan identitas dan kepribadian budaya bangsa Indonesia secara keseluruhan (Putri dkk., 2024).

Keanekaragaman motif batik yang tersebar di berbagai wilayah Indonesia menuntut adanya proses klasifikasi yang sistematis agar kelestarian warisan budaya tersebut dapat terjaga. Klasifikasi ini penting untuk mempertahankan ragam motif batik sehingga dapat diwariskan dengan tepat kepada generasi berikutnya dan memperkuat posisi batik di tingkat nasional maupun internasional.

Pemahaman tentang berbagai motif batik biasanya hanya dimiliki oleh sebagian kalangan tertentu, dikarenakan motif-motif tersebut seringkali memiliki kemiripan antar daerah sehingga sulit dibedakan secara manual. Untuk mengatasi masalah ini, metode otomatis dalam pengenalan pola dapat diterapkan sebagai solusi. Salah satu pendekatan yang efektif untuk melakukan klasifikasi motif batik adalah dengan memanfaatkan algoritma Convolutional Neural Network (CNN). Teknik pembelajaran mesin ini telah terbukti mampu mengidentifikasi dan mengelompokkan pola batik dengan tingkat akurasi yang tinggi, sehingga dapat membantu pelestarian dan pengembangan studi batik secara modern (Dewi dkk., 2024).

### 2.2.2 Deep Learning

Pembelajaran mendalam (*deep learning*) ialah subbidang dari pembelajaran mesin (*machine learning*) yang memfokuskan diri pada penerapan jaringan saraf tiruan (*artificial neural networks*) dengan struktur berlapis dalam (*deep layers*) untuk mengolah, menganalisis, dan menginterpretasikan data yang rumit. Model ini dirancang untuk meniru mekanisme otak manusia dalam mengidentifikasi pola, mengkaji hubungan, dan membuat keputusan berdasarkan data yang tersedia. *Deep learning* memiliki kemampuan luar biasa dalam menangani data tak

terstruktur, seperti gambar, suara, dan teks, dengan kinerja yang dapat melebihi metode pembelajaran konvensional. Kemajuan teknologi komputasi, adanya big data, dan algoritma yang lebih efisien telah menjadikan *deep learning* sebagai teknologi terdepan dalam revolusi kecerdasan buatan (*artificial intelligence*) (Singh dkk., 2021).

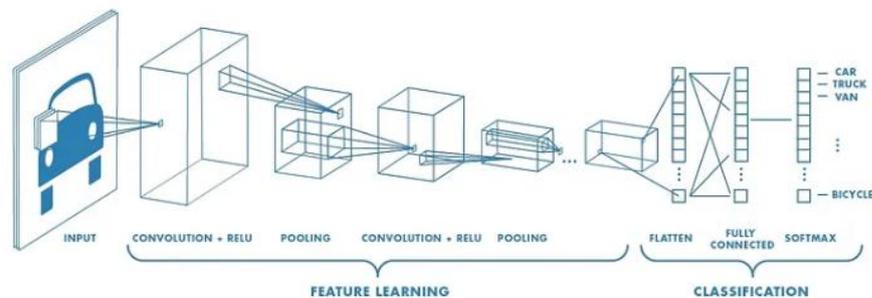
Selain itu, keistimewaan *deep learning* terletak pada kemampuannya untuk mengenali pola yang beragam dari data input yang sangat variatif, sehingga teknologi ini dapat diterapkan di berbagai sektor dan disiplin ilmu. Dalam penerapan pada tugas-tugas seperti pengenalan gambar (*image recognition*) dan pemrosesan bahasa alami (*natural language processing*), *deep learning* biasanya menunjukkan hasil akurasi yang jauh lebih baik dibandingkan dengan teknik pembelajaran mesin konvensional yang mengandalkan fitur yang ditentukan secara manual. Keunggulan ini muncul karena jaringan saraf tiruan dalam *deep learning* dapat secara otomatis mengekstrak dan mempelajari fitur-fitur penting dari data yang diberikan tanpa perlu dilakukan proses pra-pemilihan fitur oleh manusia (Roy, 2024).

Dengan demikian, pembelajaran mendalam menjadi fondasi teknologi yang kuat dan fleksibel, memungkinkan pengembangan aplikasi yang lebih canggih dan adaptif di berbagai bidang seperti pengenalan wajah, diagnostik medis, sistem rekomendasi, serta pengolahan bahasa, yang semuanya mengandalkan kemampuan algoritma untuk belajar secara mandiri dari data dalam skala besar dan kompleks.

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

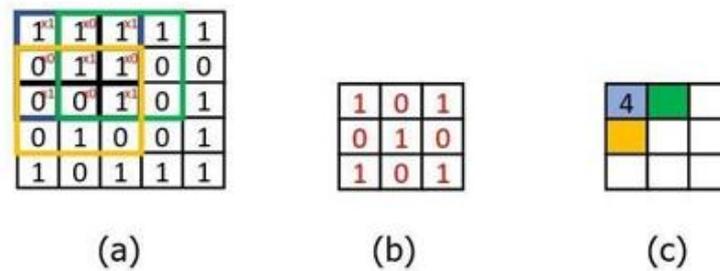
CNN termasuk ke dalam kategori *deep learning* karena memiliki struktur jaringan yang mendalam. *Deep learning* sendiri merupakan bagian dari *machine learning* yang memungkinkan komputer menjalankan tugas-tugas layaknya manusia, seperti belajar melalui proses pelatihan (Wona dkk., 2023). CNN merupakan operasi konvolusi yang menggabungkan sejumlah lapisan pemrosesan, menggunakan beberapa komponen yang bekerja secara paralel dan terinspirasi dari sistem saraf biologis. Pada CNN, setiap neuron direpresentasikan dalam dua dimensi, sehingga metode ini sangat sesuai untuk pengolahan data visual seperti

citra. Desain arsitektur jaringan berbasis CNN ditampilkan pada Gambar 2.1 di bawah ini.

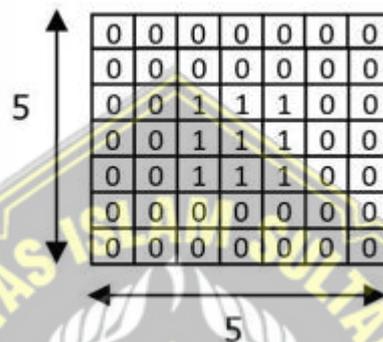


Gambar 2. 1 Arsitektur CNN (ANHAR dan PUTRA, 2023)

Rangkaian CNN mencakup data masukan, tahap pengambilan ciri, tahap pengelompokan, serta hasil keluaran (Seftianto dkk., 2023). Tahap pengambilan ciri pada CNN terdiri atas beberapa lapisan tersembunyi (*hidden layer*), yaitu lapisan konvolusi, fungsi aktivasi (ReLU), dan lapisan peringkasan (*pooling*). CNN beroperasi secara bertingkat, di mana keluaran dari lapisan konvolusi pertama dimanfaatkan sebagai masukan untuk lapisan konvolusi selanjutnya. Sementara pada tahap pengelompokan, digunakan lapisan sepenuhnya terhubung (*fully-connected*) dan fungsi aktivasi (*softmax*) yang menghasilkan keluaran berupa prediksi klasifikasi (Maulana dkk., 2023). Susunan konvolusi memakai filter buat mengekstraksi objek dari citra *input*. Filter ini berisi bobot yang digunakan buat mengetahui kepribadian dari objek semacam tepi, kurva, ataupun warna (Yuliany dkk., 2022). *Padding* merupakan akumulasi dimensi piksel dengan nilai tertentu disekitar informasi input supaya hasil dari bidang *receptive* tidak sangat kecil sehingga tidak banyak data yang lenyap (T. Firmansyah dkk., 2025). Nilai ini umumnya nol sehingga diucap dengan *zero padding*. Hasil dari bidang *receptive* berbentuk informasi tunggal. *Output* dari proses konvolusi ini dijadikan selaku input buat susunan Ilustrasi proses *stride* dan *padding* ditampilkan pada Gambar 2.2 dan Gambar 2.3.



Gambar 2. 2 Operasi Konvolusi

Gambar 2. 3 Operasi *Zero Padding* 2 pada Data 3x3

Pada fase klasifikasi, tahapan proses mencakup tiga langkah utama:

1. *Flatten*, merupakan proses yang mengonversi keluaran dari lapisan konvolusi menjadi vektor fitur satu dimensi yang siap diolah oleh lapisan *fully connected* (Juwita dkk., 2021).
2. Lapisan *fully connected*, memiliki peran serupa dengan *Artificial Neural Network (ANN)* dalam melaksanakan fungsi pengelompokan data.
3. *Softmax*, digunakan untuk menentukan nilai probabilitas dari setiap kategori, di mana kelas dengan nilai probabilitas tertinggi akan dipilih sebagai hasil prediksi.

Dalam pengembangan model *deep learning*, khususnya arsitektur CNN, fenomena *overfitting* sering kali menjadi tantangan utama yang perlu diatasi. *Overfitting* terjadi ketika model terlalu menyesuaikan diri terhadap data pelatihan, sehingga kehilangan kemampuannya untuk melakukan generalisasi terhadap data baru. Untuk meminimalkan risiko ini, digunakan beberapa strategi dalam desain arsitektur jaringan, di antaranya adalah lapisan pooling (*pooling layer*) dan *dropout layer*.

Lapisan *pooling* berfungsi untuk menyederhanakan representasi spasial dari fitur-fitur yang telah diekstraksi oleh lapisan konvolusi sebelumnya. Melalui teknik ini, dimensi dari fitur map diperkecil tanpa menghilangkan informasi yang paling penting, sehingga proses komputasi menjadi lebih efisien dan jumlah parameter yang harus dipelajari oleh model pun berkurang. Selain itu, pooling juga membantu jaringan untuk lebih tahan terhadap pergeseran dan distorsi kecil pada input gambar.

Sementara itu, lapisan dropout digunakan sebagai mekanisme regularisasi yang efektif untuk mengurangi ketergantungan model terhadap neuron-neuron tertentu. Pada tahap pelatihan, dropout bekerja dengan cara menonaktifkan sebagian unit neuron secara acak, berdasarkan nilai probabilitas tertentu ( $p$ ) yang berada pada kisaran 0 hingga 1. Dengan demikian, selama proses training, jaringan akan belajar untuk lebih menyebarkan informasi secara merata dan tidak hanya bergantung pada jalur aktivasi tertentu. Teknik ini secara signifikan mampu meningkatkan performa klasifikasi, karena memperkecil kemungkinan model mengalami overfitting dan membuatnya lebih resilien dalam menghadapi variasi data (Sentosa dkk., 2022).

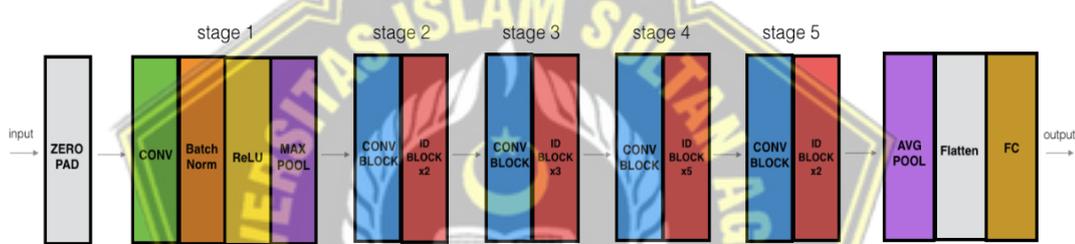
#### 2.2.4 ResNet50

ResNet-50 atau *Residual Network 50* merupakan salah satu varian arsitektur *deep* CNN yang dirancang untuk menyelesaikan permasalahan dalam pengenalan citra (*image recognition*). Model ini termasuk dalam jajaran teknologi kecerdasan buatan (*artificial intelligence*) yang canggih dan telah terbukti handal dalam berbagai tugas klasifikasi visual (Agustina, 2024).

ResNet-50 tersusun atas 50 lapisan jaringan yang terdiri dari berbagai kombinasi konvolusi, normalisasi batch, aktivasi ReLU, serta struktur residual yang menjadi ciri khas utama arsitektur ini. Jaringan ini dilatih pada dataset skala besar, yaitu ImageNet, yang memuat lebih dari satu juta gambar dari ribuan kategori objek yang berbeda. Dengan jumlah data pelatihan yang masif dan desain jaringan yang dalam (*deep architecture*), ResNet-50 mampu mencapai tingkat akurasi tinggi dalam mengenali beragam jenis objek serta fitur visual yang kompleks.

Salah satu inovasi utama dari arsitektur ResNet adalah penggunaan residual block, yaitu blok identitas yang memungkinkan sinyal input untuk dilompati (skip connection) langsung ke lapisan berikutnya. Pendekatan ini efektif dalam mengatasi masalah degradasi performa yang umum terjadi pada jaringan sangat dalam, di mana akurasi justru menurun karena kesulitan dalam propagasi gradien selama pelatihan.

Secara keseluruhan, struktur hierarkis ResNet-50 dirancang untuk mengekstraksi fitur dari tingkat rendah hingga tinggi secara bertahap, sehingga model dapat memahami representasi visual dengan sangat detail. Hal ini menjadikan ResNet-50 sebagai salah satu pilihan populer dalam berbagai penelitian klasifikasi gambar, termasuk dalam pengenalan motif batik Indonesia.



Gambar 2. 4 Arsitektur ResNet50 (Agustina, 2024)

1. Arsitektur ResNet50, dimulai dengan tahap awal berupa *zero padding* yang bertujuan untuk mempertahankan dimensi spasial dari citra masukan. Setelah proses *padding*, citra dikirim ke sebuah layer konvolusi dengan ukuran kernel yang relatif besar, dilanjutkan dengan normalisasi *batch* (*batch normalization*) dan fungsi aktivasi ReLU guna menambahkan non-linearitas ke dalam jaringan. Selanjutnya, dilakukan proses *max pooling* yang bertujuan untuk mereduksi dimensi spasial dan mengekstraksi fitur dominan dari citra.
2. Memasuki *stage 2*, jaringan terdiri dari satu *convolutional block* yang bertugas menyesuaikan dimensi agar sesuai dengan *residual connection*, diikuti oleh dua *identity block* yang mempertahankan dimensi fitur. *Identity block* ini memungkinkan arsitektur untuk mempertahankan informasi penting dari lapisan sebelumnya melalui jalur *shortcut* yang langsung mengalirkan *input* ke

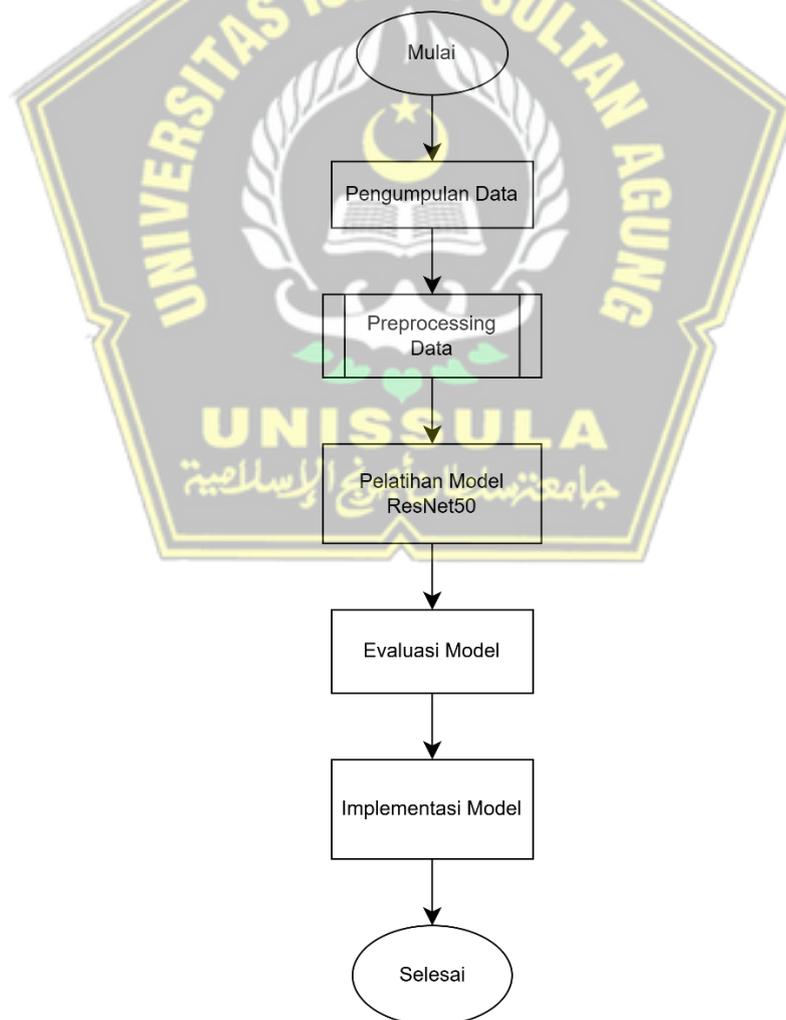
*output*, sehingga mengurangi risiko hilangnya informasi saat jaringan menjadi lebih dalam.

3. Pada *stage 3*, struktur blok residual terdiri atas satu *convolutional block* diikuti oleh tiga *identity block*. Tujuan dari *stage* ini adalah untuk menangkap fitur yang lebih kompleks dengan tetap mempertahankan kestabilan gradien melalui mekanisme *shortcut connection* yang khas dari ResNet.
4. Kemudian, *stage 4* memperdalam arsitektur dengan satu *convolutional block* dan lima *identity block*, yang memungkinkan jaringan untuk mengekstraksi representasi fitur yang semakin abstrak. Penambahan jumlah blok pada tahap ini berperan penting dalam meningkatkan kapasitas representasional jaringan tanpa menambah kompleksitas pelatihan secara signifikan.
5. Pada *stage 5*, arsitektur dilengkapi dengan satu *convolutional block* serta dua *identity block*. Meskipun lebih pendek dari *stage* sebelumnya, tahap ini bertanggung jawab untuk menangkap fitur-fitur tingkat tinggi yang krusial bagi proses klasifikasi akhir.
6. Setelah melewati seluruh tahapan ekstraksi fitur, hasil akhir dari *stage 5* diproses melalui *global average pooling* guna mereduksi dimensi fitur secara menyeluruh. *Output* dari *pooling* kemudian diratakan (*flatten*), dan akhirnya diteruskan ke *fully connected layer* yang berfungsi sebagai layer klasifikasi akhir, menghasilkan *output* akhir dalam bentuk probabilitas terhadap kelas-kelas target.

### BAB III

## METODE PENELITIAN

Pada penelitian ini menggunakan arsitektur ResNet50 dan metode CNN untuk mengklasifikasikan motif batik Indonesia. Dataset yang digunakan ialah citra batik yang didapatkan dari Kaggle. Dataset yang digunakan selanjutnya melalui serangkaian proses, meliputi penyesuaian ukuran citra, pembagian data menjadi beberapa subset, pelatihan model, serta pengujian terhadap model yang telah dilatih. Setelah proses tersebut, dilakukan evaluasi performa model, diikuti dengan implementasi menggunakan arsitektur ResNet50. Adapun langkah-langkah yang dilakukan dalam penelitian ini dijabarkan sebagai berikut:



Gambar 3. 1 Alur Pembuatan Sistem

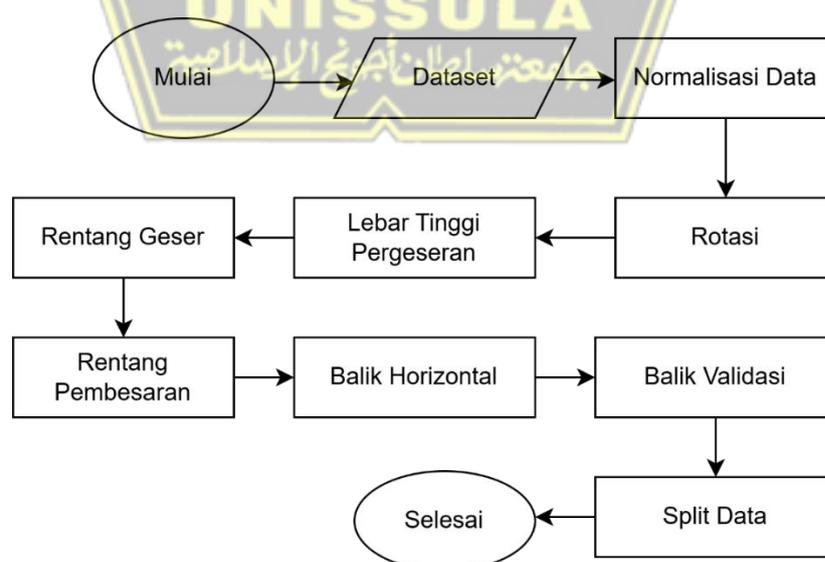
### 3.1 Pengumpulan Data

Pada tahap awal penelitian ini, data yang digunakan untuk pelatihan dan pengujian model diperoleh dari *platform* Kaggle, yaitu sebuah situs berbagi dataset dan kompetisi ilmu data yang umum digunakan dalam penelitian berbasis pembelajaran mesin. Data yang digunakan merupakan data sekunder yang berisi motif batik Indonesia. Dataset tersebut didapatkan dari platform Kaggle. Dataset mencakup variasi warna, pola, ukuran, dan pencahayaan.

Dataset yang dimanfaatkan berjudul "Motif Batik Indonesia", yang berisi kumpulan gambar digital dari berbagai macam motif batik tradisional yang berasal dari berbagai daerah di Indonesia.

### 3.2 Preprocessing Data

*Preprocessing* data atau pra-pemrosesan data merupakan tahapan penting dalam sistem pengolahan citra digital yang bertujuan untuk mempersiapkan data sebelum digunakan dalam proses pelatihan model CNN. Tahapan ini bertujuan untuk meningkatkan kualitas data input dan memperluas variasi citra melalui teknik augmentasi. Dengan demikian, model menjadi lebih tangguh dalam mengenali pola dari motif batik yang memiliki berbagai variasi bentuk, ukuran, dan arah. Sebelum data dilatih, dilakukan proses *preprocessing data* seperti pada Gambar 3.2.



Gambar 3. 2 Alur *Preprocessing Data*

1. Sistem dilakukan *preprocessing data* seperti normalisasi data agar berada pada rentang (0,1),
2. Kemudian dilakukan rotasi citra agar memungkinkan model untuk mempelajari representasi citra dari beragam perspektif visual,
3. Menggeser citra secara acak agar meningkatkan ketahanan model terhadap letak objek di dalam citra,
4. Kemudian distorsi kemiringan pada citra untuk meningkatkan kemampuan adaptasi model terhadap variasi sudut kemiringan,
5. Mengubah skala citra dengan memperbesar atau memperkecil ukurannya,
6. Dilanjutkan dengan pembalikan citra secara horizontal,
7. Sejumlah persen dari keseluruhan data dipisahkan sebagai data validasi,
8. Pada tahap terakhir dilakukan *split data*, dalam model ini, digunakan arsitektur ResNet50, Arsitektur ini dipilih karena kemampuannya dalam menangani gambar dengan tingkat kompleksitas yang tinggi, seperti gambar motif batik Indonesia, dan untuk meminimalkan kesalahan yang dapat muncul akibat kedalaman jaringan yang besar.

### 3.3 Pelatihan Model

Pelatihan model merupakan proses inti dalam pengembangan sistem klasifikasi berbasis CNN dengan arsitektur ResNet50, yang bertujuan untuk melatih jaringan saraf agar mampu mengenali pola dan fitur visual pada gambar motif batik Indonesia. Dengan melakukan pembelajaran yang efektif melalui penyesuaian bobot parameter jaringan, model dapat mengidentifikasi karakteristik unik dari tiap motif batik secara tepat. Pelatihan yang sistematis dan terkontrol juga meminimalkan risiko kesalahan serta memperkuat kemampuan model dalam menghadapi variasi data nyata. Pada tahap ini, model belajar dari data yang telah diberikan label untuk mengoptimalkan bobot parameter secara bertahap sehingga dapat menghasilkan prediksi yang akurat. Tujuan dari pelatihan model yaitu untuk menghasilkan model yang mampu melakukan klasifikasi motif batik dengan tingkat akurasi tinggi, memastikan model dapat menggeneralisasi pola dari data pelatihan

ke data baru, dan meminimalkan kesalahan klasifikasi dengan mengoptimalkan fungsi loss selama pelatihan.

### 3.4 Evaluasi Model

Evaluasi model adalah proses sistematis untuk menilai performa dan efektivitas model klasifikasi setelah proses pelatihan selesai. Tujuannya adalah untuk mengetahui sejauh mana model mampu mengenali dan mengklasifikasikan motif batik secara akurat berdasarkan citra input yang diberikan. Dalam penelitian ini, arsitektur ResNet50 dievaluasi dengan memanfaatkan sejumlah metrik evaluasi yang umum digunakan dalam mengevaluasi performa model, antara lain:

1. Kurva *accuracy* dan *Loss*

Kurva *accuracy* diterapkan pada data pelatihan dan validasi. Sementara itu, kurva *loss* pada pelatihan dan validasi dimanfaatkan untuk memantau penurunan kesalahan model.

2. Akurasi dan *loss validation*

- a. Akurasi Validasi (*Validation Accuracy*)

Akurasi validasi merupakan persentase data uji validasi yang berhasil dikenali dan diklasifikasikan secara tepat oleh model selama proses pelatihan. Metrik ini menggambarkan kemampuan model dalam menggeneralisasi terhadap data yang belum pernah dilatih sebelumnya.

Dalam penelitian ini, akurasi validasi berperan sebagai indikator seberapa efektif arsitektur ResNet50 dalam mengenali serta mengelompokkan berbagai motif batik Indonesia di luar data pelatihan. Persentase akurasi yang tinggi mencerminkan keberhasilan model dalam melakukan klasifikasi secara akurat terhadap beragam jenis motif.

- b. *Loss* Validasi (*Validation Loss*)

*Loss* validasi adalah nilai fungsi kerugian yang dihitung berdasarkan hasil prediksi model terhadap data validasi. Nilai ini menunjukkan sejauh mana kesalahan prediksi model dibandingkan dengan label aktual. *Loss* validasi digunakan untuk mengevaluasi apakah model mengalami *overfitting*, yaitu kondisi di mana model terlalu menyesuaikan diri

terhadap data latih dan gagal bekerja optimal pada data baru. Penurunan nilai *loss* validasi selama pelatihan mengindikasikan bahwa prediksi model semakin mendekati label yang benar. Jika nilai *loss* meningkat meskipun akurasi tetap atau naik, hal ini dapat menandakan bahwa model mulai kehilangan kemampuan generalisasi.

### 3. *Classification report*

Laporan klasifikasi merupakan salah satu metode evaluatif yang menyajikan sejumlah metrik kinerja penting, seperti presisi, *recall*, nilai F1, dan dukungan data (*support*) untuk tiap label kelas dalam tugas klasifikasi. Laporan ini memberikan analisis yang lebih mendalam dibandingkan hanya mengandalkan akurasi secara keseluruhan. Dalam konteks penerapan ResNet50 untuk klasifikasi motif batik Indonesia, laporan klasifikasi digunakan sebagai instrumen untuk mengevaluasi kapabilitas model dalam membedakan motif yang berbeda secara konsisten. Metrik-metrik ini membantu menilai apakah performa model merata di semua kategori, atau terdapat kecenderungan model lebih akurat dalam memprediksi motif tertentu dibandingkan lainnya. Dalam laporan klasifikasi untuk mengevaluasi model menggunakan metrik-metrik berikut.

#### a. *Accuracy*

Akurasi merupakan ukuran yang menunjukkan tingkat kesesuaian total antara prediksi model dan data aktual. Dalam kata lain, akurasi mengindikasikan persentase dari jumlah keseluruhan data uji yang berhasil diklasifikasikan dengan benar oleh model. Persamaan yang digunakan untuk menghitung nilai akurasi:

$$Accuracy = \frac{\text{Number of correct prediction}}{\text{Total of number prediction}} \quad (1)$$

Keterangan :

*Number of Correct Predictions* : Total data uji (testing data)

*Total Number of Predictions* : Total seluruh sampel atau data uji

b. *Precision*

Presisi mengukur seberapa banyak dari seluruh prediksi positif yang dilakukan oleh model benar-benar sesuai dengan kelas aslinya. Nilai presisi dihitung dengan membandingkan jumlah *true positive* terhadap total prediksi yang diklasifikasikan sebagai positif oleh model. Dalam konteks klasifikasi motif batik, nilai presisi yang tinggi pada suatu kelas menunjukkan bahwa motif yang diklasifikasikan ke dalam kelas tersebut sebagian besar memang benar adanya. Presisi sangat berguna dalam kasus di mana kesalahan positif dapat berakibat signifikan, seperti sistem deteksi penyakit atau klasifikasi data sensitif.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Keterangan :

TP : *True Positive*

FP : *False Positive*

c. *Recall*

*Recall* menunjukkan kemampuan model dalam mengenali seluruh citra batik dari suatu jenis motif tertentu. Nilai *recall* mengukur seberapa banyak citra motif yang benar-benar termasuk dalam suatu kelas dan berhasil diklasifikasikan dengan tepat oleh model. Dalam konteks klasifikasi motif batik, *recall* yang tinggi berarti bahwa hampir semua motif dari suatu kelas berhasil dikenali dan dikategorikan dengan benar oleh model. *Recall* menjadi krusial dalam situasi di mana mengabaikan kasus positif sangat merugikan, misalnya pada sistem deteksi cacat atau kelainan.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

Keterangan :

TP : *True Positive*

FN : *False Negative*

d. *F1-Score*

Skor F1 (*F1-score*) mengkombinasikan presisi dan *recall* dalam satu metrik komprehensif dengan menghitung rata-rata harmonis keduanya. Nilai F1 tinggi menunjukkan bahwa model tidak hanya mampu melakukan prediksi yang tepat (*precision*), tetapi juga tidak melewatkan data penting (*recall*). Dalam penelitian ini, skor F1 menjadi metrik yang sangat penting ketika diperlukan keseimbangan antara menangkap semua motif yang benar dan menghindari klasifikasi yang keliru. *F1-score* sangat berguna ketika distribusi data tidak merata dan diperlukan ukuran performa yang lebih adil terhadap semua kelas.

$$F1\ Score = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (4)$$

Keterangan :

TP : *True Positive*

FP : *False Positive*

FN : *False Negative*

4. *Confusion matrix*

Matriks kebingungan merupakan sebuah struktur tabular yang digunakan untuk menganalisis kinerja sistem klasifikasi, khususnya dalam mengukur tingkat kecocokan antara kategori sebenarnya dari suatu data dengan kelas yang diprediksi oleh model. Matriks ini membantu peneliti dalam mengidentifikasi pola kesalahan, serta mengevaluasi akurasi klasifikasi pada tiap label atau kelas secara lebih mendetail. Dalam kasus penelitian klasifikasi motif batik Indonesia menggunakan ResNet50, *confusion matrix* sangat berguna untuk mengetahui apakah model berhasil membedakan berbagai jenis motif secara akurat atau justru sering tertukar satu sama lain. Manfaat *confusion matrix* dalam penelitian ini yaitu untuk mengidentifikasi Kelas yang sering salah klasifikasi, mengukur ketepatan dan kesalahan per kelas, dan menjadi dasar perhitungan metrik lain.

## 5. Pengujian Model

Pengujian model merupakan tahapan krusial dalam proses evaluasi sistem pembelajaran mesin, yang bertujuan untuk mengukur tingkat akurasi dan kapabilitas generalisasi dari model terhadap data yang belum pernah dilihat sebelumnya (data uji). Dalam konteks penelitian ini, model CNN berbasis arsitektur ResNet50 diterapkan untuk mengenali dan mengklasifikasikan berbagai motif batik Indonesia. Pengujian model digunakan untuk mengevaluasi performa model pada data uji secara konkret.

### 3.5 Implementasi Model

Setelah model dievaluasi, model dari arsitektur ResNet50 yang telah dilatih sebelumnya diintegrasikan ke dalam aplikasi web interaktif dengan menggunakan *Flask*, yakni sebuah *micro web framework* berbasis Python yang bersifat ringan namun fleksibel. Tujuan dari tahap ini adalah untuk mempermudah pengguna dalam mengakses dan menggunakan sistem klasifikasi motif batik secara praktis melalui antarmuka berbasis *web*.



## **BAB IV**

### **HASIL DAN ANALISIS PENELITIAN**

#### **4.1 HASIL PENELITIAN**

##### **4.1.1 Pengumpulan Data**

Pada tahap awal penelitian ini, proses pengumpulan data dilakukan dengan mengakses kumpulan citra digital dari motif kain batik khas Indonesia. Dataset yang dimanfaatkan terdiri atas gambar-gambar batik yang telah diklasifikasikan ke dalam empat kelas, yaitu Batik Corak Insang, Batik Dayak, Batik Ikat Celup, dan Batik Megamendung.

Seluruh data diperoleh dari platform sumber terbuka Kaggle, yang menyediakan berbagai set data untuk keperluan riset dan pengembangan model kecerdasan buatan. Dataset ini tersedia dengan nama “Motif Batik Nusantara”, yang memuat gambar-gambar batik dalam format standar dan dapat digunakan secara bebas untuk keperluan ilmiah.

Dataset tersebut dipilih karena memiliki keragaman motif yang representatif terhadap karakteristik batik dari berbagai daerah di Indonesia, serta telah diproses dan dikurasi dengan cukup baik. Ketersediaan label kelas pada setiap citra juga memudahkan dalam proses pelatihan dan pengujian model klasifikasi berbasis deep learning

##### **a. Kelas Batik Corak Insang**

Pada kelas Batik Corak Insang, terdapat sebanyak 120 citra digital yang digunakan sebagai data representatif. Motif ini memiliki keunikan visual yang terinspirasi dari anatomi insang ikan, di mana bentuk dan susunannya menampilkan garis-garis lengkung dan sejajar yang tersusun secara simetris dan berulang, membentuk pola yang menyerupai struktur tenunan atau anyaman. Orientasi pola biasanya tersusun secara horizontal atau diagonal. Karakteristik ini menjadikan Batik Corak Insang mudah dikenali karena tampilan visualnya yang teratur dan elegan. Pola seperti ini juga mencerminkan nilai estetika tinggi yang dipengaruhi oleh elemen alam dan

menjadi identitas visual yang khas pada batik tersebut (H. Firmansyah dkk., 2023).

b. Kelas Batik Dayak

Kelas Batik Dayak terdiri atas 120 gambar batik yang menggambarkan kekayaan warisan budaya masyarakat Dayak di wilayah Kalimantan. Motif-motif yang terdapat dalam batik ini umumnya bersumber dari nilai-nilai tradisi dan simbol-simbol khas suku Dayak, seperti ornamen batang garing (lambang kehidupan), pola rumah panjang (huma betang), ukiran etnik, serta ilustrasi senjata khas seperti mandau. Dari segi warna, batik ini cenderung menggunakan palet warna cerah dan kontras, seperti merah, kuning, biru, dan hijau, yang melambangkan keberanian, kekuatan, dan semangat hidup masyarakatnya. Gaya desain batik Dayak memadukan elemen etnik dengan keindahan visual yang kuat dan ekspresif (Usop dan Usop, 2021).

c. Kelas Batik Ikat Celup

Dalam kategori Batik Ikat Celup, tersedia 100 citra gambar yang memperlihatkan pola-pola yang dihasilkan melalui proses pengikatan kain sebelum pewarnaan. Teknik ini dikenal secara internasional dengan sebutan *tie-dye*, di mana hasil akhirnya sangat bergantung pada teknik pengikatan serta kombinasi warna yang digunakan. Pola-pola yang muncul umumnya berbentuk spiral, garis memanjang, lingkaran, atau bentuk geometris tidak beraturan. Karena metode penciptaannya yang bersifat manual dan unik, motif yang dihasilkan dari batik ikat celup sangat variatif dan jarang ada yang sama persis. Hal ini memberikan nilai artistik yang tinggi dan menjadikan batik ini memiliki keotentikan dan keunikan tersendiri dalam setiap helai kain.

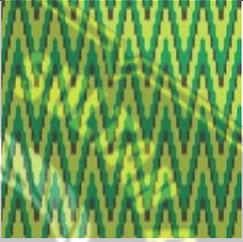
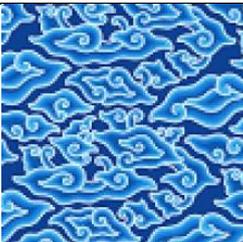
d. Kelas Batik Megamendung

Kelas Batik Megamendung berisi 100 citra visual yang menampilkan pola awan sebagai elemen dominan dalam desainnya. Motif ini berasal dari daerah Cirebon, Jawa Barat, dan sangat dikenal karena bentuk awannya yang bergelombang dengan garis-garis tegas, bertingkat, dan bertumpuk, membentuk kesan mendalam dan dinamis. Warna-warna yang digunakan

biasanya bervariasi dari biru tua, biru muda, hingga merah atau ungu, yang merepresentasikan kesabaran, keteduhan, dan kejernihan pikiran. Dalam filosofi budaya lokal, motif awan ini melambangkan harapan akan ketenangan batin dan pengendalian emosi dalam menghadapi permasalahan hidup (Salsabila dkk., 2022).

Adapun sampel citra dari dataset motif batik yang digunakan terdapat pada Tabel 4.1 berikut.

Tabel 4. 1 Sampel Citra Motif Batik

Kelas Batik	Sampel Citra
Corak Insang	
Dayak	
Ikat Celup	
Megamendung	

#### 4.1.2 *Preprocessing Data*

Tahapan pra-pemrosesan data (*data preprocessing*) merupakan langkah krusial yang dilakukan sebelum data digunakan dalam pelatihan model. Tujuan utama dari proses ini adalah untuk mengoptimalkan kualitas input data, sehingga model CNN, khususnya arsitektur ResNet50, dapat belajar secara efisien dari data visual yang tersedia. Penyesuaian ini penting untuk meningkatkan akurasi model dan mempercepat konvergensi selama proses pelatihan. Adapun tahapan pra-pemrosesan yang diterapkan dalam penelitian ini dijelaskan sebagai berikut:

1. Normalisasi Piksel Gambar

Proses pertama yang dilakukan adalah penyesuaian skala nilai piksel dari setiap citra, di mana rentang awal 0–255 dikonversi menjadi skala 0–1. Transformasi ini bertujuan untuk menstandarkan nilai-nilai input, agar model *deep learning* dapat melakukan komputasi secara lebih stabil dan efisien, mengingat model lebih responsif terhadap *input* dengan nilai numerik yang lebih kecil dan konsisten.

2. Rotasi Acak Citra

Gambar-gambar dalam dataset diputar secara acak dalam rentang sudut  $\pm 20$  derajat. Teknik augmentasi ini diterapkan untuk meningkatkan kemampuan generalisasi model, sehingga model dapat tetap mengenali pola atau motif meskipun terjadi perubahan orientasi posisi.

3. Pergeseran Horizontal (*Horizontal Shifting*)

Citra digeser ke arah kanan atau kiri hingga maksimum 20% dari lebar gambar asli. Tujuannya adalah untuk memperkenalkan variasi posisi objek secara horizontal, sehingga model tidak hanya terpaku pada posisi tetap selama proses pelatihan.

4. Pergeseran Vertikal (*Vertical Shifting*)

Melakukan pergeseran vertikal terhadap gambar (ke atas atau ke bawah) sampai batas 20% dari tinggi gambar. Tujuannya serupa, yaitu memberikan keragaman posisi objek dalam dimensi tinggi.

5. Transformasi *Shearing*

Transformasi shear dilakukan dengan mengubah struktur citra melalui kemiringan sudut tertentu, seolah-olah gambar dimiringkan secara diagonal. Tujuan dari teknik ini adalah untuk mensimulasikan distorsi geometris ringan dan melatih model agar lebih tangguh terhadap bentuk motif yang sedikit berubah dari bentuk aslinya.

6. *Zoom In/Out* Acak

Proses ini melibatkan perbesaran maupun pengecilan citra secara acak hingga  $\pm 20\%$  dari ukuran gambar asli. Teknik ini bertujuan untuk mengenalkan model terhadap objek dalam berbagai ukuran, baik yang tampak lebih dekat maupun lebih jauh, sehingga model tidak bergantung pada skala tertentu dalam mengenali motif.

7. Pembalikan Horizontal (*Horizontal Flipping*)

Gambar-gambar dalam dataset dibalik secara horizontal secara acak. Teknik ini sangat berguna terutama jika arah motif tidak memengaruhi identifikasi kelas, misalnya pada motif batik yang bersifat simetris atau fleksibel arah.

8. Pembagian Dataset (*Data Splitting*)

*Split data* dengan membagi kumpulan data menjadi dua bagian utama, yakni data pelatihan dan data validasi. Adapun tahapan dalam pemisahan data meliputi:

a. Distribusi dataset

Dataset dipisahkan menjadi dua subnet utama, yakni:

- 1) Sebesar 80% dari keseluruhan data, yakni sebanyak 352 gambar, digunakan sebagai data pelatihan untuk membangun model. Data ini berperan dalam membantu model mengenali pola-pola Motif Batik dari empat kategori.
- 2) Data validasi sebanyak 20% atau 88 gambar dari total dataset dimanfaatkan untuk menilai performa model selama proses pembelajaran berlangsung.

- 3) Pembagian dataset dilakukan secara acak (*randomized splitting*) guna memastikan distribusi data dalam setiap kelas tetap seimbang dan mencerminkan representasi yang adil antara subset pelatihan dan validasi.

*b. Batch Size*

*Batch size* mengacu pada jumlah data yang diproses sekaligus dalam satu siklus iterasi. Pengaturan *batch size* disesuaikan dengan kemampuan komputasi perangkat keras (GPU/CPU) yang digunakan. Dalam studi ini, ukuran batch ditetapkan sebesar 32 untuk memfasilitasi pelatihan model secara optimal dan efisien.

#### 4.1.3 Pelatihan Model

CNN adalah jenis jaringan saraf tiruan yang sangat efektif untuk pengolahan dan klasifikasi citra. Dalam penelitian ini, CNN diimplementasikan melalui arsitektur ResNet50 yang merupakan model CNN dalam kategori *deep convolutional network*. CNN bekerja dengan memanfaatkan lapisan-lapisan konvolusi untuk mengekstraksi fitur dari citra masukan, seperti tepi, tekstur, pola, dan bentuk yang kompleks — elemen-elemen yang sangat penting dalam membedakan motif batik. Tahapan proses CNN dalam model ResNet50 antara lain:

1. *Input Citra*

Citra motif batik dengan resolusi yang telah diubah menjadi 224x224 dimasukkan ke dalam model.

2. *Convolutional Layers*

Layer konvolusi di ResNet50 menggunakan filter berukuran kecil untuk mendeteksi fitur visual dari citra. Setiap filter belajar mendeteksi fitur spesifik, seperti pola garis atau kurva khas motif batik.

3. *Batch Normalization & ReLU Activation*

Setelah konvolusi, dilakukan normalisasi untuk mempercepat pelatihan dan diikuti oleh fungsi aktivasi ReLU (Rectified Linear Unit) yang menambahkan non-linearitas agar model mampu belajar pola kompleks.

#### 4. *Residual Block*

Ciri khas dari ResNet50 adalah adanya *residual connections*, yaitu shortcut antar layer yang memungkinkan informasi melewati beberapa lapisan tanpa mengalami transformasi. Hal ini membantu mengatasi masalah *vanishing gradient* pada jaringan yang sangat dalam dan mempercepat konvergensi.

#### 5. *Pooling Layers*

*Layer pooling* digunakan untuk mengurangi dimensi spasial dari fitur dan mempertahankan fitur penting. Dalam model ini, digunakan *GlobalAveragePooling2D* pada akhir model dasar sebelum masuk ke *fully connected layer* untuk merangkum informasi global dari fitur yang telah diekstrak.

#### 6. *Fully Connected Layers (Dense Layers)*

Setelah fitur diekstraksi, CNN melanjutkan ke *dense layers*, yaitu:

- a. Dense(128) dengan aktivasi ReLU
- b. Dropout(0.5) untuk regularisasi
- c. Dense(4) dengan aktivasi softmax untuk mengeluarkan probabilitas dari empat kelas motif batik

#### 7. *Softmax Output*

*Output* dari layer *softmax* menghasilkan vektor probabilitas dengan panjang empat, yang masing-masing menunjukkan kemungkinan citra termasuk dalam satu dari empat kelas batik.

Proses pelatihan model dilakukan dengan memanfaatkan arsitektur ResNet50 sebagai *base model* melalui pendekatan *transfer learning*. Arsitektur ResNet50 digunakan karena telah dilatih pada dataset ImageNet dan memiliki kemampuan ekstraksi fitur visual yang kuat. Pada tahap ini, layer klasifikasi asli pada ResNet50 dihapus dengan mengatur parameter *include\_top=False*, agar model dapat disesuaikan dengan jumlah kelas pada dataset, yaitu empat kelas motif batik. Selanjutnya, seluruh layer pada model dasar dibekukan (*trainable=False*) agar bobotnya tidak diperbarui selama pelatihan. Hal ini bertujuan untuk mempertahankan fitur-fitur umum yang telah dipelajari oleh model dan mencegah overfitting, terutama ketika jumlah data pelatihan relatif terbatas.

Setelah itu, ditambahkan *layer* klasifikasi baru (*custom classifier*) yang terdiri dari *layer GlobalAveragePooling2D* untuk meratakan hasil ekstraksi fitur tanpa menggunakan metode *flatten*, *Dense layer* sebanyak 128 unit dengan fungsi aktivasi ReLU, serta *Dropout* sebesar 0.5 untuk mengurangi risiko *overfitting* dengan menonaktifkan sejumlah neuron secara acak selama proses pelatihan. *Layer output* terdiri dari *Dense layer* dengan 4 unit dan fungsi aktivasi softmax yang digunakan untuk menghasilkan probabilitas prediksi dari masing-masing kelas batik.

Model kemudian dikompilasi menggunakan fungsi *loss categorical\_crossentropy*, karena klasifikasi bersifat multikelas dengan label *one-hot encoded*. Optimizer yang digunakan adalah Adam, yang dikenal mampu melakukan konvergensi secara cepat dan efisien. Untuk meningkatkan performa dan efisiensi pelatihan, digunakan juga mekanisme *EarlyStopping* dengan memantau nilai *val\_loss*. Jika selama 10 epoch berturut-turut tidak ada peningkatan pada nilai *loss* validasi, proses pelatihan akan dihentikan lebih awal dan bobot terbaik akan dikembalikan (*restore best weights*).

Pelatihan model dilakukan menggunakan data latih dan data validasi yang telah dibagi sebelumnya dari dataset citra batik. Proses pelatihan dijalankan hingga maksimum 100 *epoch*, atau berhenti lebih awal jika memenuhi kondisi *early stopping*. Setelah proses pelatihan selesai, model disimpan dalam format *.h5* agar dapat digunakan kembali untuk proses prediksi atau diintegrasikan ke dalam aplikasi klasifikasi batik berbasis *web* atau *mobile*.

Adapun *hyperparameter* yang digunakan pada proses pelatihan, yakni:

Tabel 4. 2 Hyperparameter Proses Pelatihan

Kategori	Hyperparameter	Nilai	Tujuan
<i>Pre-trained</i>	<i>weights</i>	'imagenet'	Menggunakan bobot ( <i>weights</i> ) hasil pelatihan di dataset ImageNet untuk <i>transfer learning</i> .
	<i>Include_top</i>	<i>False</i>	Menghilangkan bagian akhir ( <i>fully connected layer</i> ) dari ResNet50.
	<i>Include_shape</i>	(224, 224, 3)	Menentukan dimensi input gambar.
<i>Transfer Learning</i>	<i>base_model.trainable</i>	<i>False</i>	Membekukan bobot model ResNet50 agar tidak ikut dilatih.
<i>Layer Tambahan</i>	<i>Dense units</i>	128	Menambahkan layer <i>fully connected</i> dengan 128 unit dan fungsi aktivasi ReLU.
	<i>Dropout rate</i>	0.5	Menghindari <i>overfitting</i> dengan mengabaikan 50% neuron secara acak saat pelatihan.
	<i>activation</i>	'relu'	<i>Output layer</i> dengan 4 neuron untuk 4 kelas batik, menggunakan aktivasi <i>softmax</i> .
<i>Optimizer</i>	<i>optimizer</i>	'adam'	Mengatur cara model melakukan <i>update</i> bobot saat pelatihan.

Tabel 4. 3 (Lanjutan)

Kategori	Hyperparameter	Nilai	Tujuan
<i>Data Augmentation</i>	<i>rotation_range</i>	Berbagai nilai	Membuat variasi data pelatihan secara sintetis.
	<i>zoom_range</i>		
<i>Batch &amp; Epoch</i>	<i>batch_size</i>	32	Menentukan jumlah sampel yang diproses sebelum model melakukan <i>update</i> bobot.
	<i>Epochs</i>	100	Jumlah maksimal siklus pelatihan lengkap pada seluruh data.
<i>Early Stopping</i>	<i>Patience</i>	10	Menghentikan pelatihan jika tidak ada peningkatan pada <i>validation loss</i> selama 10 <i>epoch</i> .

## 4.2 ANALISA PENELITIAN

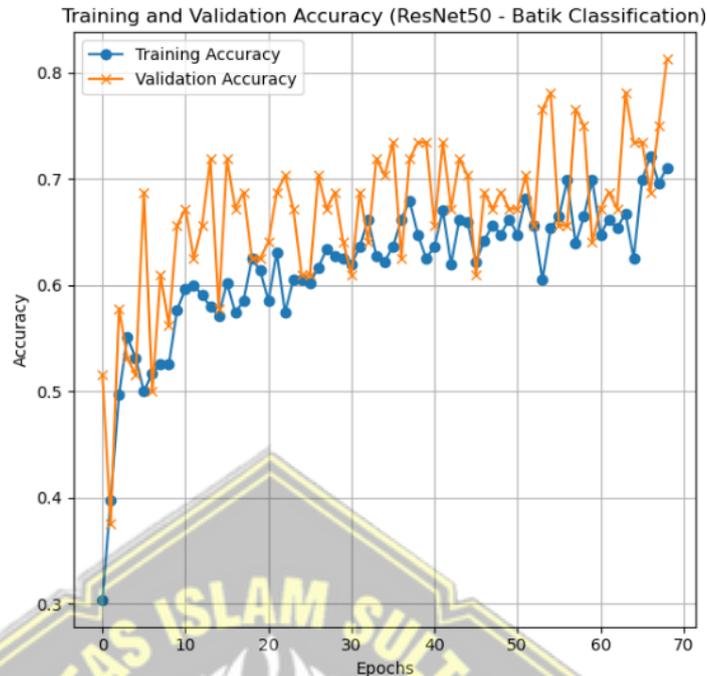
### 4.2.1 Evaluasi Model

Evaluasi terhadap arsitektur model ResNet50 dilakukan dengan memanfaatkan data validasi. Evaluasi ini mencakup aspek kinerja serta efisiensi model. Adapun bentuk evaluasi yang diterapkan pada model ResNet50 adalah sebagai berikut:

#### 1. Kurva *Accuracy* dan *Loss*

##### a. Kurva *Accuracy*

Kurva ini berfungsi untuk mengukur prediksi benar dibandingkan dengan total prediksi. Kurva ini digunakan untuk menilai kemampuan generalisasi model.



Gambar 4.1 Kurva Accuracy

Gambar 4.1 merupakan kurva *accuracy* dari model ResNet50 dalam mengklasifikasikan motif batik Indonesia. Di mana sumbu X merupakan *epochs* atau jumlah iterasi pelatihan, sumbu Y merupakan *accuracy* atau tingkat akurasi antara 0.3 hingga 0.8+. Garis biru yang terdapat pada kurva tersebut merupakan *training accuracy*, dan garis *orange* merupakan *validation accuracy*.

Adapun penjelasan mengenai kurva di atas sebagai berikut.

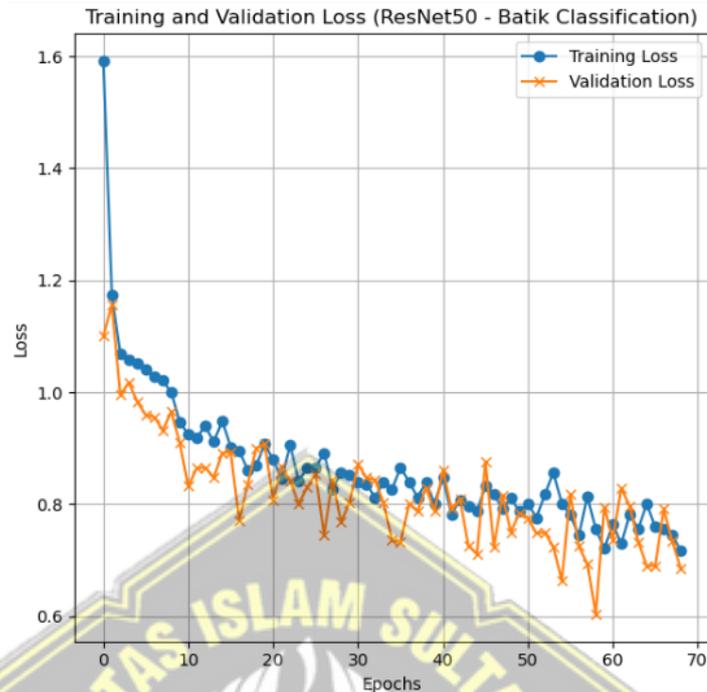
- 1) Awal pelatihan (*Epoch* 0-10)
  - a) Akurasi pelatihan naik tajam dari sekitar 0.32 ke 0.55.
  - b) Akurasi validasi juga meningkat dengan fluktuasi tajam, menunjukkan model mulai belajar dari data.
  - c) Fluktuasi pada *validation accuracy* awal adalah hal yang umum karena model masih dalam proses penyesuaian parameter.

- 2) Pertengahan *epoch* (*Epoch* 10-40)
  - a) Akurasi pelatihan meningkat secara bertahap, namun cenderung lebih stabil.
  - b) *Validation accuracy* cukup berfluktuasi, tetapi secara keseluruhan menunjukkan tren naik.
  - c) Perbedaan antara *training* dan *validation accuracy* tidak terlalu lebar → indikasi bahwa model belum mengalami *overfitting* serius.
- 3) Akhir pelatihan (*Epoch* 40-70)
  - a) *Training accuracy* terus meningkat, mencapai sekitar 0.7.
  - b) *Validation accuracy* menunjukkan fluktuasi signifikan tetapi menyentuh puncak mendekati 0.82.
  - c) Fluktuasi *validation accuracy* yang tinggi bisa disebabkan oleh data validasi yang bervariasi atau model sangat sensitif terhadap *batch*.

Pada hal ini, model ResNet50 menunjukkan performa yang cukup stabil dan efektif dalam mengenali pola pada data batik. *Validation accuracy* yang lebih tinggi dari *training accuracy* menandakan bahwa model tidak hanya menghafal data, tetapi juga belajar dengan baik untuk membuat generalisasi.

b. Kurva *Loss*

Kurva ini berfungsi untuk menunjukkan nilai eror atau kesalahan dalam melakukan prediksi. Kurva *loss* menggambarkan apakah model semakin baik dalam meminimalkan kesalahan seiring bertambahnya *epoch*. Tujuannya yakni untuk memonitor stabilitas dan konfigurasi model. *Loss* yang terus menurun menandakan proses pelatihan berjalan dengan baik.



Gambar 4. 2 Kurva Loss

Gambar 4.2 merupakan grafik kurva *training* dan *validation* loss dari model ResNet50 untuk mengklasifikasikan batik. Di mana sumbu X merupakan *epoch* atau jumlah siklus pelatihan dan sumbu Y merupakan *loss* atau nilai fungsi kerugian. Garis biru yang terdapat pada grafik tersebut merupakan *training loss*, sedangkan garis *orange* merupakan *validation loss*.

Adapun analisis dari kurva tersebut ialah sebagai berikut:

- 1) *Epoch* awal (0-10)
  - a) *Training loss* mulai sangat tinggi (~1.6) dan turun tajam, menandakan bahwa model cepat belajar dari data pelatihan.
  - b) *Validation loss* juga menurun secara signifikan, dari sekitar 1.2 ke 0.9, mengindikasikan bahwa model belajar secara efektif dari data baru.
  - c) Pada fase ini model mengalami *learning* yang cepat (fase konvergensi awal).

## 2) *Epoch* tengah (10-40)

- a) Kedua *loss* (*training* dan *validation*) mengalami penurunan lebih lambat dan stabil.
- b) Kurva *validation loss* kadang berada di bawah kurva *training loss*, ini menunjukkan bahwa generalisasi model cukup baik (tidak terjadi *overfitting*).
- c) Fluktuasi kecil masih wajar dan menunjukkan respons terhadap *batch* data validasi yang berbeda-beda.

## 3) *Epoch* akhir (40-70)

- a) *Training loss* mencapai titik stabil di sekitar 0.7 dan tetap fluktuatif ringan.
- b) *Validation loss* juga menurun bertahap hingga sekitar 0.65 dengan fluktuasi ringan.
- c) Tidak ada indikasi divergensi atau kenaikan tajam → ini menandakan bahwa model tidak *overfit* meskipun pelatihan berlangsung lama.

Pada hal ini, model ResNet50 menunjukkan performa pelatihan yang stabil dan efisien, dengan penurunan *loss* pada *training* dan *validation* yang konsisten. Model memiliki kapasitas generalisasi yang baik terhadap data batik dan tidak menunjukkan gejala *overfitting* yang berarti.

## 2. Akurasi dan *Loss Validation*

*Validation accuracy* merupakan presentase prediksi yang benar dari model saat diuji dengan data validasi yang berguna untuk mengukur kemampuan generalisasi model, yakni seberapa baik model dapat mengenali data baru dan menjadi tolak ukur bahwa model berhadil mengenali pola penting dari data.

*Validation loss* merupakan nilai rata-rata kesalahan prediksi model terhadap validasi yang berfungsi untuk mengukur seberapa baik *output* model mendekati target benar, menyediakan informasi tentang kestabilan dan

kepercayaan model terhadap prediksinya, dan membantu mendeteksi *overfitting*.

Tabel 4. 4 Akurasi dan *Loss Validation*

<i>Validation Loss</i>	0.6594688296318054
<i>Validation Accuracy</i>	0.8068181872367859

Tabel 4.4 merupakan hasil akurasi dan *loss validation*, di mana *validation accuracy* sebesar 0.8068 yang berarti seluruh data validasi yang diberikan kepada model sekitar 80,68% prediksi sudah benar. Angka tersebut menunjukkan bahwa model memiliki tingkat generalisasi yang cukup baik terhadap data yang belum pernah terlihat sebelumnya (data validasi). Untuk klasifikasi citra batik, nilai tersebut termasuk cukup tinggi dan menunjukkan bahwa model mampu mengenali pola-pola visual dari berbagai jenis batik dengan akurasi yang memadai.

Hasil dari *validation loss* sebesar 0.6594, yang lebih rendah menandakan prediksi model semakin mendekati nilai targer yang sebenarnya. Oleh karena itu, model ResNet50 menunjukkan performa validasi yang cukup baik.

### 3. *Classification Report*

*Classification report* menampilkan metrik evaluasi untuk membantu mengidentifikasi kelas mana yang diklasifikasikan dengan baik, kelas mana yang sering salah klasifikasi, dan ketidakseimbangan performa antar kelas.

Tabel 4. 5 *Classification Report*

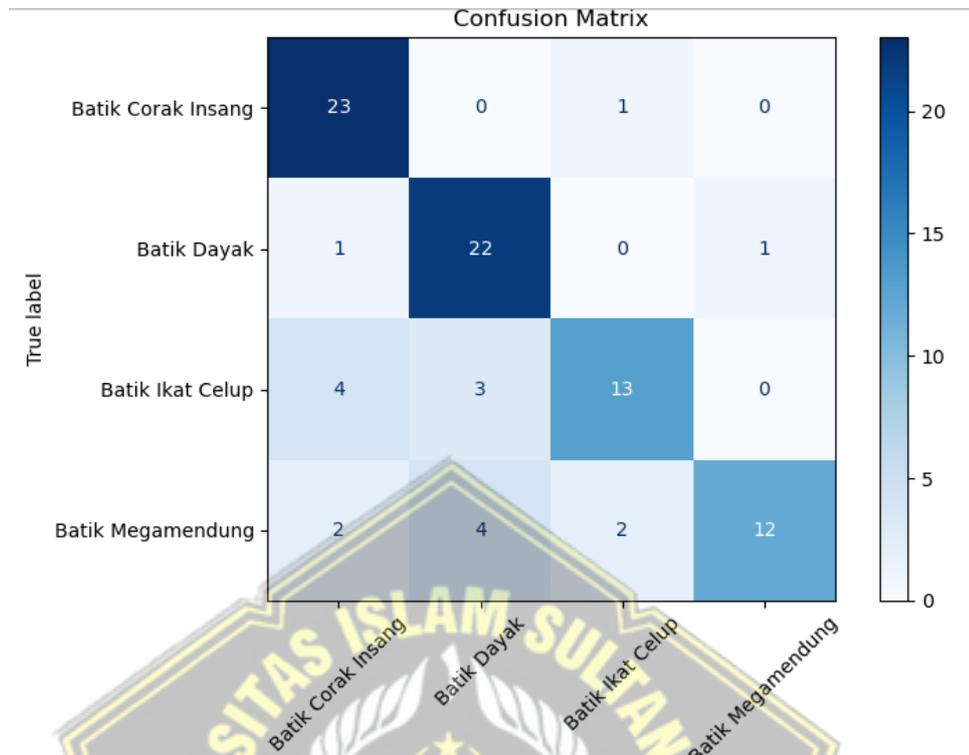
	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>Support</i>
Batik Corak Insang	0.77	0.96	0.85	24
Batik Dayak	0.76	0.92	0.83	24
Batik Ikat Celup	0.87	0.65	0.74	20
Batik Megamendung	0.93	0.65	0.76	20
<i>Accuracy</i>			0.81	88
<i>Macro Avg</i>	0.83	0.79	0.80	88
<i>Weighted Avg</i>	0.82	0.81	0.80	88

Tabel 4.5 merupakan hasil classification report model ResNet50. Pada kelas Batik Corak Insang menunjukkan *precision* 0.77, *recall* 0.96, *f1-score* 0.85 yang menandakan model sangat baik mengenali batik ini, Batik Dayak menunjukkan *precision* 0.76, *recall* 0.92, *f1-score* 0.83 yang menandakan hampir sama dengan Corak Insang, Batik Ikat Celup menunjukkan *precision* 0.84, *recall* 0.65, *f1-score* 0.74 yang menandakan *precision* tertinggi namun *recall* rendah berarti beberapa gambar ikat celup tidak dikenali, Batik Megamendung menunjukkan *precision* 0.93, *recall* 0.65, *f1-score* 0.76.

Akurasi keseluruhan model memiliki akurasi cukup baik yaitu 81%. Model yang paling baik mengenali yakni Batik Corak Insang dan Dayak dengan *recall* tertinggi, namun perlu adanya peningkatan pada *recall* kelas Megamendung dan Ikat Celup agar performa lebih seimbang.

#### 4. *Confusion Matrix*

Pada *confusion matrix* untuk klasifikasi 4 jenis batik menggunakan model ResNet50, menyajikan perbandingan label sebenarnya (*true label*) dengan prediksi model (*predicted label*). Sumbu vertikal (*True Label*) merupakan label sebenarnya dari data, sumbu horizontal (*Predicted Label*) merupakan label yang diprediksi oleh model, nilai diagonal merupakan prediksi yang benar (*True Positive*), dan nilai di luar diagonal merupakan prediksi yang salah (*False Positive* dan *False Negative*).



Gambar 4. 3 Confusion Matrix

Gambar 4.3 merupakan hasil *confusion matrix* dari pengujian model ResNet50 dengan analisa sebagai berikut:

- a. Kelas Batik Corak Insang
  - 1) Dapat diprediksi benar sebagai Corak Insang sebanyak 23 kali
  - 2) Salah diklasifikasikan sebagai Ikat Celup sebanyak 1 kali
  - 3) Total sampel sebanyak 24 dengan akurasi kelas 95.8%
- b. Kelas Batik Dayak
  - 1) Dapat diprediksi benar sebagai Dayak sebanyak 22 kali
  - 2) Salah diklasifikasikan sebagai Corak Insang sebanyak 1 kali
  - 3) Salah diklasifikasikan sebagai Megamendung sebanyak 1 kali
  - 4) Total sampel sebanyak 24 dengan akurasi kelas 91.7%

c. Kelas Batik Ikat Celup

- 1) Dapat diprediksi benar sebagai Ikat Celup sebanyak 13 kali
- 2) Salah diklasifikasikan sebagai Corak Insang sebanyak 4 kali
- 3) Salah diklasifikasikan sebagai Dayak sebanyak 3 kali
- 4) Total sampel sebanyak 20 dengan akurasi kelas 65%

d. Kelas Batik Megamendung

- 1) Dapat diprediksi benar sebagai Megamendung sebanyak 12 kali
- 2) Salah diklasifikasikan sebagai Corak Insang sebanyak 2 kali
- 3) Salah diklasifikasikan sebagai Dayak sebanyak 2 kali
- 4) Total sampel sebanyak 20 dengan akurasi 60%

Analisa di atas menunjukkan bahwa Batik Corak Insang dan Batik Dayak dapat diklasifikasikan sangat baik, dengan kelas Batik Ikat Celup dan Batik Megamendung lebih sulit dibedakan oleh model.

5. Pengujian Model

Pada tahap ini dilakukan proses evaluasi terhadap model klasifikasi gambar batik yang memanfaatkan arsitektur ResNet50, dengan empat kategori kelas, yaitu Batik-Corak Insang, Batik-Dayak, Batik-Ikat Celup, dan Batik-Megamendung.

Setiap gambar dianalisis untuk memperoleh output prediksi kelas batik beserta tingkat keyakinan (*confidence score*) dari model terhadap hasil klasifikasinya. *Output* dari proses evaluasi ini disajikan dalam Tabel X, yang mencakup nama *file* citra, hasil prediksi oleh model, persentase tingkat keyakinan, label kelas sebenarnya, serta status prediksi (tepat atau tidak tepat).

Tabel 4. 6 Hasil Pengujian Model

Citra (ID>Nama)	Hasil Prediksi	Kepercayaan (%)	Data Aktual (Data Asli)	Status
Batik_01.jpg	Batik-Corak Insang	53.81%	Batik-Corak Insang	Benar
Batik_02.jpg	Batik-Corak Insang	99.87%	Batik-Corak Insang	Benar
Batik_03.jpg	Batik-Corak Insang	58.46%	Batik-Corak Insang	Benar
Batik_04.jpg	Batik-Corak Insang	60.45%	Batik-Corak Insang	Benar
Batik_05.jpg	Batik-Corak Insang	81.21%	Batik-Corak Insang	Benar
Batik_06.jpg	Batik-Dayak	86.2%	Batik-Dayak	Benar
Batik_07.jpg	Batik-Ikat Celup	41.21%	Batik-Dayak	Salah
Batik_08.jpg	Batik-Dayak	93.49%	Batik-Dayak	Benar
Batik_09.jpg	Batik-Dayak	90.5%	Batik-Dayak	Benar
Batik_10.jpg	Batik-Dayak	99.06%	Batik-Dayak	Benar
Batik_11.jpg	Batik-Dayak	49.79%	Batik-Ikat Celup	Salah
Batik_12.jpg	Batik-Ikat Celup	68.67%	Batik-Ikat Celup	Benar
Batik_13.jpg	Batik-Corak Insang	57.92%	Batik-Ikat Celup	Salah
Batik_14.jpg	Batik-Ikat Celup	42.43%	Batik-Ikat Celup	Benar
Batik_15.jpg	Batik-Ikat Celup	43.56%	Batik-Ikat Celup	Benar
Batik_16.jpg	Batik-Megamendung	71.49%	Batik-Megamendung	Benar
Batik_17.jpg	Batik-Megamendung	74.79%	Batik-Megamendung	Benar
Batik_18.jpg	Batik-Megamendung	78.28%	Batik-Megamendung	Benar
Batik_19.jpg	Batik-Megamendung	94.71%	Batik-Megamendung	Benar
Batik_20.jpg	Batik-Dayak	88.35%	Batik-Megamendung	Salah

Berdasarkan hasil pengujian, diketahui bahwa model ResNet50 berhasil mengklasifikasikan 16 dari 20 citra uji dengan benar, yang berarti model mencapai akurasi sebesar 80%. Hal ini menunjukkan bahwa model mampu

mengenali sebagian besar pola dan motif batik dengan cukup baik. Adapun 4 citra lainnya diklasifikasikan secara salah, salah satunya adalah citra Batik\_13.jpg yang diprediksi sebagai Batik-Corak Insang, padahal label aktualnya adalah Batik-Ikat Celup. Kesalahan lainnya juga terjadi pada citra Batik\_11.jpg dan Batik\_20.jpg, yang menunjukkan bahwa model masih mengalami kebingungan pada pola-pola yang memiliki kemiripan visual antar kelas.

Selain itu, dapat diamati bahwa tingkat kepercayaan model terhadap prediksi yang benar umumnya berada pada kisaran tinggi (>70%), sedangkan prediksi yang salah cenderung memiliki nilai kepercayaan yang lebih rendah, seperti pada citra Batik\_11.jpg dengan *confidence* 49.79%. Hal ini mengindikasikan bahwa tingkat kepercayaan dapat dijadikan indikator tambahan untuk menilai keandalan prediksi model.

Dengan hasil ini, dapat disimpulkan bahwa arsitektur ResNet50 cukup efektif dalam mengklasifikasikan motif batik ke dalam empat kelas, meskipun masih terdapat ruang untuk perbaikan, terutama dalam membedakan motif yang memiliki ciri visual yang serupa. Evaluasi ini menjadi dasar untuk melakukan perbaikan lebih lanjut, seperti peningkatan jumlah data latih, augmentasi citra, atau penerapan teknik *fine-tuning* untuk meningkatkan akurasi model secara keseluruhan.

#### 4.2.2 Hasil Implementasi Model

Setelah proses pelatihan model selesai dilakukan dan diperoleh model klasifikasi batik dengan performa yang optimal, langkah selanjutnya adalah tahap implementasi. Implementasi bertujuan untuk menerapkan model yang telah dilatih ke dalam lingkungan nyata yang dapat digunakan oleh pengguna akhir. Dalam konteks penelitian ini, model klasifikasi motif batik yang dibangun dengan memanfaatkan arsitektur ResNet50 diintegrasikan ke dalam sebuah aplikasi berbasis web interaktif dengan menggunakan *Flask*, yakni sebuah *micro web framework* berbasis *Python* yang bersifat ringan namun fleksibel. Tujuan dari tahap

ini adalah untuk mempermudah pengguna dalam mengakses dan menggunakan sistem klasifikasi motif batik secara praktis melalui antarmuka berbasis web.

Tujuan dari pengembangan aplikasi ini adalah untuk menyediakan sebuah platform yang memungkinkan proses klasifikasi citra batik dilakukan secara otomatis dan efisien oleh pengguna tanpa perlu keterlibatan langsung dalam aspek teknis dari model machine learning. Model yang telah dilatih sebelumnya disimpan dalam format tertentu (*model serialization*) dan dimuat kembali saat aplikasi dijalankan untuk melakukan prediksi terhadap citra yang diunggah oleh pengguna.

Proses implementasi model klasifikasi motif batik menggunakan arsitektur ResNet50 ke dalam aplikasi web berbasis Flask dimulai dengan menyiapkan model yang telah dilatih. Setelah proses pelatihan dan evaluasi selesai, model disimpan dalam format seperti .h5 agar dapat digunakan kembali di luar lingkungan pelatihan. Selanjutnya, struktur dasar aplikasi Flask dibangun dengan memisahkan beberapa komponen, seperti file utama aplikasi (*app.py*), folder statis untuk menyimpan gambar yang diunggah, folder template untuk file HTML, serta *file* pendukung untuk fungsi *preprocessing* dan prediksi.

Setelah struktur siap, model yang telah disimpan dimuat ke dalam aplikasi *Flask* saat server dijalankan. *Flask* berperan sebagai jembatan antara pengguna dan model, dengan menyediakan antarmuka web yang memungkinkan pengguna untuk mengunggah gambar batik. Ketika gambar diunggah, *Flask* akan menyimpannya ke direktori tertentu dan memproses gambar tersebut sebelum diberikan ke model. Proses ini meliputi perubahan ukuran gambar sesuai dengan kebutuhan model, normalisasi nilai piksel, serta konversi gambar ke dalam format array numerik.

Gambar yang telah diproses kemudian diberikan ke model ResNet50 untuk diklasifikasikan. Model akan memprediksi kelas atau jenis motif batik. Hasil klasifikasi ini kemudian ditampilkan kembali kepada pengguna melalui halaman web, lengkap dengan gambar yang diunggah dan nama motif hasil prediksi. Dengan demikian, proses klasifikasi yang sebelumnya hanya dapat dilakukan melalui kode *Python* kini dapat diakses dengan lebih mudah dan interaktif melalui *browser*.

Aplikasi dirancang dengan memperhatikan prinsip kemudahan penggunaan (*usability*) dan keterbacaan hasil klasifikasi. Antarmuka pengguna (*user interface*)

dibangun secara sederhana namun tetap informatif, menyajikan hasil prediksi berupa nama motif batik, tingkat keyakinan model terhadap hasil tersebut (*confidence score*), serta visualisasi gambar input.

Dengan demikian, implementasi ini tidak hanya menunjukkan validitas model dalam konteks eksperimental, tetapi juga membuktikan potensi penerapannya dalam dunia nyata sebagai alat bantu digital yang mendukung pelestarian dan pengenalan kekayaan budaya batik Indonesia secara lebih luas melalui teknologi kecerdasan buatan.

Adapun *hyperparameter* yang digunakan untuk mengimplementasikan *flask* ke dalam sebuah *website* antara lain:

Tabel 4. 7 *Hyperparameter Flask*

No.	Parameter	Tujuan
1.	<code>from flask import Flask, request, render_template</code>	Untuk membuat <i>web app</i> , menerima input gambar, dan menampilkan hasil ke HTML
2.	<code>from tensorflow.keras.models import load_model</code>	Untuk memuat model CNN yang telah dilatih sebelumnya
3.	<code>from tensorflow.keras.preprocessing.image import img_to_array, load_img</code>	Untuk <i>preprocessing</i> gambar menjadi format input model
4.	<code>import numpy as np</code>	Untuk manipulasi <i>array</i> , normalisasi, dan analisis hasil prediksi
5.	<code>import os</code>	Untuk mengatur <i>path</i> , folder <i>upload</i> , dan penyimpanan <i>file</i>
6.	<code>import matplotlib.pyplot as plt</code>	Untuk membuat grafik <i>confidence</i> hasil prediksi

Tabel 4. 8 (Lanjutan)

No.	Parameter	Tujuan
7.	<code>App = Flask(_name_)</code>	Membuat <i>instance Flask</i> sebagai dasar aplikasi <i>web</i>
8.	<code>UPLOAD_FOLDER = 'static/uploads'</code>	Menentukan folder untuk menyimpan <i>file</i> yang diunggah <i>user</i>
9.	<code>Model = load_model("ResNet50.h5")</code>	Memuat model CNN ResNet50 untuk klasifikasi batik
10.	<code>Class_labels</code> dan <code>class_descriptions</code>	Menyediakan label dan deskripsi budaya untuk setiap kelas yang dikenali model
11.	<code>@app.route("/", methods = ["GET"])</code>	Menampilkan halaman utama ( <i>index.html</i> ) saat <i>user</i> membuka <i>website</i>
12.	<code>@app.route("/predict/", methods = ["POST"])</code>	Menangani unggahan gambar, proses prediksi, dan menampilkan hasil
13.	<code>File = request.files['file']</code>	Mengambil gambar yang diunggah dari <i>form</i> HTML
14.	<code>file.save(filepath)</code>	Menyimpan gambar ke folder <i>static/uploads</i>
15.	<code>load_img(..., target_size=(224, 224))</code>	Resize gambar ke ukuran input model (224x224 untuk ResNet50)
16.	<code>img_to_array(...) / 255.0</code>	Mengubah gambar ke <i>array</i> dan normalisasi nilai piksel ke 0–1

Tabel 4. 9 (Lanjutan)

No.	Parameter	Tujuan
17.	<code>np.expand_dims(...)</code>	Menambahkan dimensi batch agar sesuai dengan input model
18.	<code>model.predict(image_array)</code>	Melakukan prediksi menggunakan model CNN
19.	<code>np.argmax(predictions)</code>	Menentukan kelas dengan nilai <i>confidence</i> tertinggi
20.	<code>matplotlib.pyplot + plt.savefig(...)</code>	Membuat dan menyimpan grafik <i>confidence</i> sebagai visualisasi
21.	<code>render_template(...)</code>	Mengirim data hasil prediksi ke HTML untuk ditampilkan
22.	<code>app.run(debug=True, port=6060)</code>	Menjalankan server <i>Flask</i> di <i>localhost port 6060</i> dengan mode <i>debug</i> aktif

Tabel 4.6 merupakan *hyperparameter* yang digunakan untuk mengimplementasikan *flask*. Untuk membangun aplikasi klasifikasi gambar berbasis web menggunakan Flask, beberapa parameter dan komponen penting perlu diimplementasikan. Pertama, *library Flask* diimpor dengan `from flask import Flask, request, render_template`, yang digunakan untuk membuat struktur aplikasi web, menerima input dari pengguna (seperti gambar), dan menampilkan hasil prediksi ke halaman HTML. Untuk memuat model *deep learning* yang telah dilatih sebelumnya, digunakan `load_model` dari `tensorflow.keras.models`.

Gambar yang diunggah perlu diproses terlebih dahulu, sehingga digunakan `img_to_array` dan `load_img` dari Keras untuk mengubah gambar menjadi *array* numerik. Proses manipulasi *array*, normalisasi, dan pengambilan hasil prediksi dilakukan dengan bantuan *library numpy*. Selain itu, modul `os` digunakan untuk

mengatur direktori, seperti membuat folder *upload* dan menyimpan *file* yang diunggah. Untuk menampilkan hasil prediksi secara visual, digunakan `matplotlib.pyplot` yang memungkinkan pembuatan grafik *confidence* dalam bentuk *bar chart*.

Aplikasi *Flask* sendiri diinisialisasi dengan `app = Flask(__name__)`, yang merupakan titik awal aplikasi. Folder *upload* ditentukan melalui `UPLOAD_FOLDER = 'static/uploads'`, agar gambar yang dikirim oleh user dapat disimpan secara lokal. Model CNN, seperti *ResNet50*, dimuat dengan `model = load_model("ResNet50.h5")`, dan daftar label kelas (`class_labels`) serta deskripsi budayanya (`class_descriptions`) disiapkan agar hasil prediksi bisa ditampilkan secara informatif.

Routing merupakan bagian penting dari *Flask*. Route `@app.route("/", methods=["GET"])` digunakan untuk menampilkan halaman utama (*index.html*) saat *user* pertama kali membuka *website*. Sementara itu, route `@app.route("/predict/", methods=["POST"])` bertugas menangani unggahan gambar, menjalankan proses prediksi, serta mengembalikan hasil ke antarmuka pengguna.

Setelah gambar diambil menggunakan `file = request.files['file']`, gambar tersebut disimpan dengan `file.save(filepath)` ke dalam direktori *upload*. Gambar kemudian diproses dengan `load_img()` untuk *resize* ke ukuran 224x224 piksel, lalu diubah ke *array* dan dinormalisasi dengan `img_to_array(...)/255.0`. Agar dapat diprediksi oleh model, gambar ditambahkan satu dimensi batch menggunakan `np.expand_dims(...)`. Prediksi dilakukan dengan `model.predict(image_array)`, dan hasilnya dianalisis untuk mendapatkan kelas dengan *confidence* tertinggi menggunakan `np.argmax(predictions)`.

Visualisasi *confidence* dari seluruh kelas dilakukan dengan `matplotlib.pyplot` dan hasilnya disimpan menggunakan `plt.savefig(...)`. Semua data hasil prediksi, termasuk nama kelas, *confidence*, deskripsi, dan grafik, dikirim ke HTML menggunakan `render_template(...)`. Terakhir, aplikasi dijalankan dengan `app.run(debug=True, port=6060)`, yang mengaktifkan *server*

*Flask* pada *port* 6060 dalam mode *debug* agar memudahkan pengembangan dan pelacakan *error*.

Gambar berikut merupakan tampilan dari sistem klasifikasi motif Batik Indonesia yang telah diimplementasikan dalam bentuk *website*.



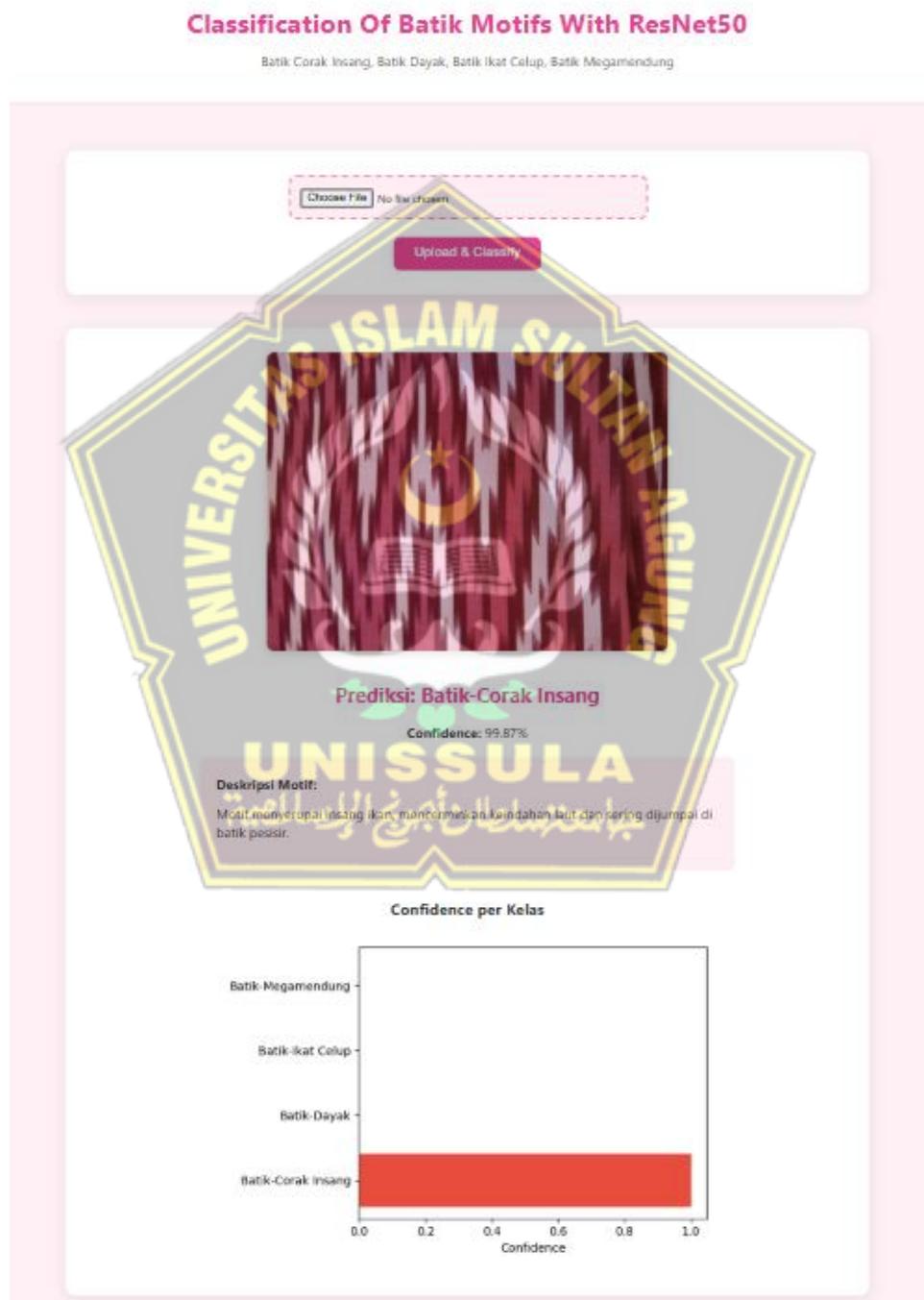
Gambar 4. 4 Tampilan Awal Sistem

Gambar 4.4 merupakan tampilan awal dari hasil implementasi model ResNet50 menggunakan CNN. Tampilan awal ini berfungsi sebagai antarmuka utama bagi pengguna untuk mengunggah citra batik yang ingin diklasifikasikan. Pada bagian atas halaman terdapat judul sistem, yaitu "*Classification of Batik Motifs With ResNet50*", yang menjelaskan fungsi utama aplikasi. Tepat di bawahnya, terdapat daftar nama-nama kelas batik yang dapat dikenali oleh model, yaitu Batik Corak Insang, Batik Dayak, Batik Ikat Celup, dan Batik Megamendung. Penyajian informasi ini dimaksudkan agar pengguna mengetahui cakupan klasifikasi yang tersedia dalam sistem.

Selanjutnya, komponen utama antarmuka terdiri dari dua elemen: tombol unggah *file* (*input type="file"*) dan tombol *Upload & Classify*. Tombol unggah *file* digunakan untuk memilih citra batik dari perangkat pengguna, sementara tombol prediksi akan memproses citra tersebut dan menghasilkan *output* klasifikasi berdasarkan hasil prediksi model.

Antarmuka dirancang dengan gaya visual yang bersih dan minimalis, sehingga memudahkan pengguna umum untuk memahami alur kerja aplikasi tanpa

memerlukan pengetahuan teknis mendalam. Penekanan pada kesederhanaan juga bertujuan untuk meningkatkan pengalaman pengguna (*user experience*), terutama dalam konteks sistem berbasis kecerdasan buatan yang sering kali dianggap kompleks oleh pengguna awam.



Gambar 4. 5 Tampilan Hasil Klasifikasi Motif Batik

Pada Gambar 4.5 merupakan menunjukkan tampilan hasil dari sistem klasifikasi motif batik setelah pengguna mengunggah sebuah citra batik ke dalam aplikasi. Setelah tombol *Upload & Classify* ditekan, sistem akan memproses citra yang diunggah menggunakan model klasifikasi ResNet50 yang telah dilatih sebelumnya, dan kemudian menampilkan hasil prediksi secara langsung pada halaman *web*. aplikasi akan menampilkan hasil prediksi dalam bentuk label kelas batik beserta nilai *confidence* atau tingkat kepercayaan model terhadap hasil prediksi tersebut. Nilai ini ditampilkan dalam persentase, yang menggambarkan seberapa yakin model terhadap keputusan klasifikasinya. Misalnya, jika model memprediksi kelas "Batik-Corak Insang" dengan *confidence* sebesar 99.87%, berarti model sangat yakin bahwa citra tersebut termasuk dalam kelas tersebut.

Untuk memberikan konteks tambahan kepada pengguna, aplikasi juga menyertakan deskripsi umum dari motif batik yang diprediksi. Deskripsi ini berisi penjelasan singkat mengenai ciri khas atau filosofi dari motif batik terkait, misalnya Batik Corak Insang dijelaskan sebagai motif yang menyerupai insang ikan, mencerminkan keindahan laut dan banyak ditemui di batik pesisir.

Sebagai pelengkap visual, aplikasi juga menampilkan sebuah diagram batang (*bar chart*) yang merepresentasikan *confidence* per kelas. Grafik ini menampilkan tingkat kepercayaan model terhadap masing-masing dari keempat kelas batik, sehingga pengguna dapat melihat distribusi probabilitas yang diberikan oleh model dan mengetahui apakah ada kelas lain yang hampir mendekati nilai prediksi utama.

Aplikasi ini dikembangkan menggunakan *framework* Flask pada sisi *backend*, dan memanfaatkan HTML, CSS, serta JavaScript untuk tampilan antarmuka (*UI*). Model ResNet50 yang telah dilatih disimpan dalam format *.h5* dan dipanggil pada saat proses prediksi berlangsung. Dengan adanya implementasi ini, model klasifikasi batik tidak hanya diuji secara akademis, tetapi juga dapat langsung digunakan dalam bentuk aplikasi nyata yang siap pakai.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai “Implementasi Arsitektur ResNet50 pada Klasifikasi Motif Batik Indonesia menggunakan Metode *Convolutional Neural Network (CNN)*”, dapat disimpulkan bahwa penelitian ini berhasil membangun sebuah model klasifikasi citra batik berbasis *deep learning* dengan memanfaatkan arsitektur ResNet50 melalui pendekatan *transfer learning*. Model ini mampu mengklasifikasikan empat jenis motif batik, yaitu Batik Corak Insang, Batik Dayak, Batik Ikat Celup, dan Batik Megamendung. Hasil pelatihan model menunjukkan bahwa akurasi tertinggi yang dicapai pada data uji sebesar 81%, dengan nilai *validation accuracy* sebesar 80,68% dan *validation loss* sebesar 65,94%. Nilai tersebut menunjukkan bahwa model memiliki kinerja yang cukup baik dalam membedakan motif batik, meskipun masih terdapat peluang untuk perbaikan lebih lanjut.

Berdasarkan analisis hasil klasifikasi, diketahui bahwa kelas Batik Corak Insang dan Batik Dayak merupakan kelas yang paling dikenali secara akurat oleh model, dengan tingkat prediksi yang konsisten tinggi. Hal ini mengindikasikan bahwa kedua motif tersebut memiliki ciri visual yang lebih dominan atau khas dibandingkan kelas lainnya. Implementasi model ke dalam bentuk aplikasi *web* memberikan kemudahan bagi pengguna dalam mengklasifikasikan motif batik tanpa perlu memahami konsep teknis dari *machine learning*. Aplikasi ini mampu menampilkan hasil prediksi secara *real-time* dengan visualisasi dan deskripsi motif yang informatif.

## 5.2 Saran

Sebagai tindak lanjut dari penelitian ini, beberapa saran dapat diajukan untuk pengembangan lebih lanjut guna meningkatkan akurasi model klasifikasi citra batik. Pertama, disarankan untuk memperluas jumlah dan keragaman data citra batik yang digunakan, mencakup berbagai daerah dan variasi motif, guna meningkatkan kemampuan generalisasi model serta mengurangi risiko bias terhadap kelas tertentu. Kedua, selain menggunakan arsitektur ResNet50, eksperimen lebih lanjut dapat dilakukan dengan menerapkan arsitektur CNN lain seperti EfficientNet, DenseNet, atau MobileNet untuk mengevaluasi dan membandingkan performa serta efisiensi masing-masing model dalam konteks klasifikasi batik. Ketiga, perlu diterapkan teknik augmentasi data yang lebih bervariasi serta optimasi *hyperparameter* yang lebih intensif, termasuk *fine-tuning* layer-layer tertentu, agar nilai *validation loss* dapat ditekan dan akurasi prediksi dapat ditingkatkan secara signifikan.



## DAFTAR PUSTAKA

- Af'idah, D. I., Dairoh, D., Handayani, S. F., dan Pratiwi, R. W. (2021). Pengaruh Parameter Word2Vec terhadap Performa Deep Learning pada Klasifikasi Sentimen. *Jurnal Informatika: Jurnal Pengembangan IT*, 6(3), 156–161. <https://doi.org/10.30591/jpit.v6i3.3016>
- Agustina, D. A. (2024). Klasifikasi Citra Jenis Kulit Wajah Dengan Algoritma Convolutional Neural Network (Cnn) Resnet-50. *Jurnal Riset Sistem Informasi*, 1(3), 01–07. <https://doi.org/10.69714/13sby24>
- Alberto, J., dan Hermanto, D. (2023). Klasifikasi Jenis Burung Menggunakan Metode CNN ResNet-50 [1]. *Jurnal Teknik Informatika dan Sistem Informasi*, 10(3), 34–36.
- Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., dan Elgar, M. (2023). Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning. *Karya Ilmiah Mahasiswa Bertauhid (KARIMAH TAUHID)*, 2(1), 1–6.
- Algoritma, P., Neural, C., Citra, K., Sapi, D., dan Babi, D. A. N. (2022). *Network Arsitektur Resnet-50 Untuk*.
- ANHAR, A., dan PUTRA, R. A. (2023). Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(2), 466. <https://doi.org/10.26760/elkomika.v11i2.466>
- Azmi, K., Defit, S., dan Sumijan, S. (2023). Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat. *Jurnal Unitek*, 16(1), 28–40. <https://doi.org/10.52072/unitek.v16i1.504>
- Dewi, S., Ramadhani, F., dan Djasmayena, S. (2024). Klasifikasi Jenis Jerawat Berdasarkan Gambar Menggunakan Algoritma CNN (Convolutional Neural Network). *Hello World Jurnal Ilmu Komputer*, 3(2), 68–73. <https://doi.org/10.56211/helloworld.v3i2.518>
- Dewi, S., Saputri, H., Lukman, A., dan Irsan, M. (2025). *Perbandingan Performa*

*Arsitektur CNN Terhadap Klasifikasi Tumor Otak Menggunakan Data MRI.* 6(4). <https://doi.org/10.47065/bits.v6i4.6710>

Dianda, N., Rachman, A. S., dan Yadnya, M. S. (2025). *Identifikasi Citra Motif Kain Tenun Sumbawa ( Kre Alang ) Menggunakan Metode Convolutional Neural Network Arsitektur.* 6(2). <https://doi.org/10.47065/josh.v6i2.6774>

Firdaus, R., Joni Satria, dan Baidarus, B. (2022). Klasifikasi Jenis Kelamin Berdasarkan Gambar Mata Menggunakan Algoritma Convolutional Neural Network (CNN). *Jurnal CoSciTech (Computer Science and Information Technology)*, 3(3), 267–273. <https://doi.org/10.37859/coscitech.v3i3.4360>

Firmansyah, H., Ramadhan, I., Wiyono, H., Putri, A. E., dan Atmaja, T. S. (2023). Perkembangan dan pelestarian tenun Corak Insang khas kota Pontianak. *Satwika : Kajian Ilmu Budaya dan Perubahan Sosial*, 7(1), 12–20. <https://doi.org/10.22219/satwika.v7i1.23933>

Firmansyah, T., Kurniawan, R., dan Hidayat, A. T. (2025). *Klasifikasi Tingkat Kematangan Roasting Biji Kopi Berbasis Deep Learning Klasifikasi Tingkat Kematangan Roasting Biji Kopi Berbasis Deep Learning dengan Arsitektur MobileNet.* January. <https://doi.org/10.47065/josh.v6i2.6811>

Juwita, A. R., Mudzakir, T. Al, Pratama, A. R., Husodo, P., dan Sulaiman, R. (2021). Batik Image Identification with Convolutional Neural Network Method. *Buana Ilmu*, 6(1), 192–208.

Kurniadi, B. W., Prasetyo, H., Ahmad, G. L., Aditya Wibisono, B., dan Sandya Prasvita, D. (2021). Analisis Perbandingan Algoritma SVM dan CNN untuk Klasifikasi Buah. *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA) Jakarta-Indonesia, September*, 1–11.

Maulana, I., Sastypratiwi, H., Muhandi, H., Safriadi, N., dan Sujaini, H. (2023). Implementasi Convolutional Neural Network (CNN) untuk Klasifikasi Motif Batik pada Aplikasi Computer Vision Berbasis Android. *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, 9(3), 384. <https://doi.org/10.26418/jp.v9i3.69496>

Peryanto, A., Yudhana, A., dan 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>

Pujiati, R., dan Rochmawati, N. (2022). Identifikasi Citra Daun Tanaman Herbal Menggunakan Metode Convolutional Neural Network (CNN). *Journal of Informatics and Computer Science (JINACS)*, 3(03), 351–357.  
<https://doi.org/10.26740/jinacs.v3n03.p351-357>

Putri, Y. A., Azhar, Y., dan Minarno, A. E. (2024). Klasifikasi Jenis Batik Menggunakan Algoritma Convolutional Neural Network. *Jurnal Repositor*, 3(2), 83–89. <https://doi.org/10.22219/repositor.v3i2.31050>

Rizal, F., Hasyim, F., Malik, K., dan Yudistira, Y. (2022). Implementasi Algoritma Convolutional Neural Networks (CNN) Untuk Klasifikasi Batik. *COREAI: Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi*, 2(2), 40–47.  
<https://doi.org/10.33650/coreai.v2i2.3365>

Roy, R. R. (2024). Deep Learning. *Networked Artificial Intelligence*, November, 83–98. <https://doi.org/10.1201/9781003499466-10>

Salsabila, A., Kurniawan, A., dan Retnoningsih, S. (2022). Perancangan Media Buku Interaktif Pengenalan Batik Mega Mendung Untuk Anak Sekolah Dasar Kelas 4-6. *Fad*.  
<https://e proceeding.itenas.ac.id/index.php/fad/article/view/1150%0Ahttps://e proceeding.itenas.ac.id/index.php/fad/article/download/1150/1084>

Saputra, D. H., Imran, B., dan Juhartini. (2023). Object Detection Untuk Mendeteksi Citra Buah-Buahan Menggunakan Metode Yolo. *Jurnal Kecerdasan Buatan dan Teknologi Informasi*, 2(2), 70–80.  
<https://doi.org/10.69916/jkbti.v2i2.18>

Seftianto, F., Sukemi, S., dan Nawawi, Z. (2023). Deep Learning Berbasis CNN Untuk Pengenalan Pola Partial Discharge Isolasi Silicone Rubber. *SINTECH (Science and Information Technology) Journal*, 6(2), 68–75.  
<https://doi.org/10.31598/sintechjournal.v6i2.1390>

Sentosa, E., Mulyana, D. I., Cahyana, A. F., dan Pramuditasari, N. G. (2022). Implementasi Image Classification Pada Batik Motif Bali Dengan Data Augmentation dan Convolutional Neural Network. *Jurnal Pendidikan*

*Tambusai*, 6(1), 1451–1463.

- Septipalan, M. L., Hibrizi, M. S., Latifah, N., Lina, R., dan Bimantoro, F. (2024). Klasifikasi Tumor Otak Menggunakan CNN Dengan Arsitektur Resnet50. *Seminar Nasional Teknologi & Sains*, 3(1), 103–108. <https://doi.org/10.29407/stains.v3i1.4357>
- Singh, R., Gehlot, A., Kumar Prajapat, M., dan Singh, B. (2021). Deep Learning. In *Artificial Intelligence in Agriculture*. <https://doi.org/10.1201/9781003245759-5>
- Usop, L. S., dan Usop, T. B. (2021). Peran Kearifan Lokal Masyarakat Dayak dalam Mengembangkan Batik Benang Bintik di Kalimantan Tengah. *Mudra Jurnal Seni Budaya*, 36(3), 405–413. <https://doi.org/10.31091/mudra.v36i3.1502>
- Wona, M. M. A., Asyifa, S. A., Virgianti, R., Hamid, M. N., Handoko, I. M., Septiani, N. W. P., dan Lestari, M. (2023). Klasifikasi Batik Indonesia Menggunakan Convolutional Neural Network (CNN). *Jurnal Rekayasa Teknologi Informasi (JURTI)*, 7(2), 172. <https://doi.org/10.30872/jurti.v7i2.13694>
- Yudi Aprianingrum, A., dan Hayati Nufus, A. (2021). Batik Indonesia, Pelestarian Melalui Museum. *Prosiding Seminar Nasional Industri Kerajinan dan Batik*, 1–14.
- Yuliany, S., Aradea, dan Andi Nur Rachman. (2022). Implementasi Deep Learning pada Sistem Klasifikasi Hama Tanaman Padi Menggunakan Metode Convolutional Neural Network (CNN). *Jurnal Buana Informatika*, 13(1), 54–65. <https://doi.org/10.24002/jbi.v13i1.5022>