

**BUKU TUGAS AKHIR
CAPSTONE DESIGN**



**KLASIFIKASI JENIS BATIK
MENGUNAKAN *DEEP LEARNING*
BERBASIS APLIKASI**

Oleh :

Aulia Chusnyriani Sani Zulkarnaen / 1101194043

I Gusti Ngurah Rejski Ariantara Putra/ 1101190017

Nada Fauzia Reviana / 1101194198

Rahmawati Hidayah / 1101194070

**PRODI S1 TEKNIK TELEKOMUNIKASI
FAKULTAS TEKNIK ELEKTRO
UNIVERSITAS TELKOM
BANDUNG
2023**

LEMBAR PENGESAHAN BUKU CAPSTONE DESIGN

KLASIFIKASI JENIS BATIK MENGGUNAKAN *DEEP LEARNING* BERBASIS APLIKASI

*(BATIK TYPE CLASSIFICATION USING DEEP LEARNING
BASED APPLICATION)*

**Telah disetujui dan disahkan sebagai bagian dari Capstone Design
Program S1 Teknik Telekomunikasi
Fakultas Teknik Elektro
Universitas Telkom
Bandung**

Disusun oleh:

**Aulia Chusnyriani Sani Zulkarnaen / 1101194043
I Gusti Ngurah Rejski Ariantara P / 1101190017
Nada Fauzia Reviana / 1101194198
Rahmawati Hidayah / 1101194070**

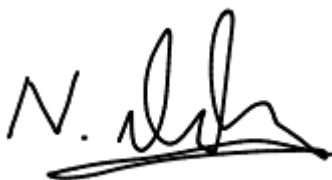
Bandung, 29 Juli 2023

Menyetujui,

Pembimbing 1

Pembimbing 2

Pembimbing 3



Nur Ibrahim S.T., M.T.

NIP. 14870047



R Yunendah Nur
Fu'adah, S.T., M.T.

NIP. 17900087



Nor Kumalasari Caecar
Pratiwi S.T., M. T.

NIP. 20890017

LEMBAR PERNYATAAN ORISINALITAS

Kami, yang bertanda tangan di bawah ini

| | Anggota 1 | Anggota 2 | Anggota 3 | Anggota 4 |
|-----------------|--|--|---|---|
| Nama: | Aulia Chusnyriani Sani Zulkarnaen | I Gusti Ngurah Rejski Ariantara Putra | Nada Fauzia Reviana | Rahmawati Hidayah |
| NIM: | 1101194043 | 1101190017 | 1101194198 | 1101194043 |
| Alamat: | Jl.Sukabirus no.124, ds. Citeureup, Dayeuhkolot, Kab. Bandung. | Jalan Mangga II, RT.1/RW.2, Desa Sukapura, Dayeuhkolot, Bandung | Kost Arga Putri, Jalan Sukabirus RT/RW 03/15 No. 40, Bandung | Jl. PGA No.2, Lengkong, Kec. Bojongsoang, Bandung |
| No. Telepon: | 085707797467 | 081353020531 | 085921518735 | 081295705528 |
| Email : | Auliariiiani7@ gmail.com | gustiari2001@ gmail.com | nadafauzia18 @gmail.com | rahmawatihida yah18@ gmail.com |

Menyatakan bahwa Buku Capstone Design ini merupakan karya orisinal penulis bersama dengan kelompok Capstone Design penulis, dengan judul:

KLASIFIKASI JENIS BATIK MENGGUNAKAN *DEEP LEARNING* BERBASIS APLIKASI

*(BATIK TYPE CLASSIFICATION USING DEEP LEARNING BASED
APPLICATION)*

Atas pernyataan ini, penulis siap menanggung resiko/sanksi yang dijatuhkan kepada penulis apabila dikemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidak aslian karya ini.

Bandung, 29 Juli 2023

(tanda tangan)

(tanda tangan)

Aulia Chusnyriani Sani Zulkarnaen
1101194043

I Gusti Ngurah Rejski Ariantara Putra
1101190017

(tanda tangan)

(tanda tangan)

Nada Fauzia Reviana
1101194198

Rahmawati Hidayah
1101194070

ABSTRAK

Batik adalah kain tradisional dan sebuah warisan turun menurun yang dimiliki oleh bangsa Indonesia. Batik telah diakui oleh dunia melalui penetapan *United Nations Educational, Scientific, and Cultural Organization* (UNESCO) sebagai hak kebudayaan intelektual bangsa Indonesia pada tanggal 2 Oktober 2009. Berdasarkan penelitian dari organisasi IACI motif batik yang tercatat berjumlah 5.849, sehingga diperlukan sebuah sistem yang dapat membantu untuk klasifikasi jenis batik. Pada tugas akhir ini akan dirancang sebuah sistem aplikasi yang dapat mengidentifikasi ke dalam enam kelas jenis batik berbasis aplikasi *mobile* dengan tujuan mudah digunakan dan ramah terhadap pengguna. Pemilihan batik ini didapat dari hasil kuisisioner mengenai kepopuleran dan jenis batik yang tampak familiar di kalangan Mahasiswa Universitas Telkom. Dengan adanya rancangan sistem yang dibuat dapat berkontribusi untuk melestarikan kebudayaan batik yang beragam di Indonesia.

Pada tugas akhir ini akan dirancang sebuah sistem aplikasi bernama BatiQu yang dapat mengidentifikasi jenis batik yang berbeda. Aplikasi BatiQu mampu mengenali dan mengidentifikasi enam jenis batik yaitu batik Ceplok, Kawung, Megamendung, Nitik, Parang dan Tambal. Aplikasi BatiQu menggunakan *Deep Learning* dengan metode CNN menggunakan arsitektur pilihan yang nantinya akan diuji dan digunakan hasil terbaik untuk diimplementasikan pada aplikasi. Arsitektur pilihan untuk metode CNN adalah ResNet152 V2 dan MobileNet V1. Proses *Deep Learning* dimulai dari tahap *pre-processing*, *feature extraction* dan *classification*. Dari proses *Deep Learning* yang sudah dilakukan, selanjutnya akan diimplementasikan ke dalam aplikasi BatiQu untuk pengujian ulang dan melihat seberapa akurat sistem yang telah dibuat. Perancangan aplikasi BatiQu juga diimplementasikan ke dalam bentuk perangkat keras seperti *box*. *Box* BatiQu dirancang berbentuk kubus dengan ukuran Panjang 12 cm, lebar 12 cm dan tinggi 17 cm berbahan kayu.

Dari pengujian yang dilaksanakan, menunjukkan bahwa untuk arsitektur terbaik pada metode CNN adalah arsitektur ResNet152 V2. Hal ini berdasarkan dari hasil pengujian yang dimulai dari tahap *pre-processing*, *feature extraction* hingga *classification*. Dimana hasil yang didapatkan bernilai 89,67% yang menandakan bahwa sistem model dapat berjalan sesuai dengan yang dirancang. Hasil dari pengujian *Deep Learning* kemudian dimasukkan kedalam *mobile application* dan menghasilkan kinerja performa yang baik dan layak untuk digunakan sebagai aplikasi identifikasi jenis batik.

Kata kunci : Batik, Identifikasi, *Deep Learning*, CNN, Aplikasi

ABSTRACT

Batik is a traditional cloth and a heritage owned by the Indonesian people. Batik has been recognized by the world through the determination of the United Nations Educational, Scientific, and Cultural Organization (UNESCO) as the intellectual cultural rights of the Indonesian people on October 2, 2009. Based on research from the IACI organization, there are 5,849 batik motifs recorded, so a system is needed that can help classify batik types. In this final project, an application system that can identify into six classes of batik types based on mobile applications will be designed with the aim of being easy to use and friendly to users. The selection of batik is obtained from the results of a questionnaire regarding the popularity and types of batik that seem familiar among Telkom University students. With the design of the system created, it can contribute to preserving the diverse batik culture in Indonesia.

In this final project, an application system called BatiQu will be designed that can identify different types of batik. BatiQu application is able to recognize and identify six types of batik namely batik Ceplok, Kawung, Megamendung, Nitik, Parang and Tambal. The BatiQu application uses Deep Learning with the CNN method using a selected architecture which will later be tested and the best results used to be implemented in the application. The preferred architectures for the CNN method are ResNet152 V2 and MobileNet V1. The Deep Learning process starts from the pre-processing, feature extraction and classification stages. From the Deep Learning process that has been carried out, it will then be implemented into the BatiQu application for retesting and seeing how accurate the system that has been made. The design of the BatiQu application is also implemented into the form of hardware such as boxes. The BatiQu box is designed in the form of a cube with a size of 12 cm long, 12 cm wide and 17 cm high made of wood.

From the tests carried out, it shows that the best architecture for the CNN method is the ResNet152 V2 architecture. This is based on the test results starting from the pre-processing stage, feature extraction to classification. Where the results obtained are worth 89.67% which indicates that the model system can run as designed. The results of Deep Learning testing are then incorporated into the mobile application and produce good performance and are suitable for use as a batik type identification application.

Keyword : Batik, Identification, Deep Learning, CNN, Application

KATA PENGANTAR

Puja dan puji syukur senantiasa penulis panjatkan kepada Tuhan Yang Maha Esa berkat Rahmat dan Karunia-Nya, penulis mampu menyelesaikan Tugas Akhir *Capstone Design* yang berjudul “Klasifikasi Jenis Batik Menggunakan *Deep Learning* Berbasis Aplikasi” dengan baik. Penelitian ini dibuat dengan tujuan sebagai salah satu persyaratan untuk menyelesaikan Program Studi Prodi S1 Teknik Telekomunikasi Fakultas Teknik Elektro, Universitas Telkom.

Penelitian ini berisikan tentang implementasi untuk identifikasi jenis batik. Identifikasi ini dapat dilakukan dimana saja dan kapan saja. Di Indonesia sendiri terdapat motif batik yang mencapai 5.849 motif berdasarkan penelitian oleh IACI. Dimana dari sekian banyak motif batik yang ada membuat masyarakat terkadang masih merasa kurang mengetahui pada jenis batik. Oleh karena itu, penelitian ini dirancang menggunakan aplikasi yang mudah untuk digunakan dengan biaya serta perawatan yang terjangkau kepada masyarakat. Dengan harapan dan tujuan agar masyarakat lebih mengenali jenis batik dan melestarikan kebudayaan Indonesia.

Penulis menyadari bahwa penelitian Tugas Akhir *Capstone Design* jauh dari kata yang sempurna. Dimana masih terdapat kesalahan dalam penyusunan penulisan Tugas Akhir. Oleh karena itu, penulis meminta maaf yang sedalam-dalamnya atas kesalahan yang dilakukan penulis. Penulis berharap semoga Tugas Akhir ini dapat bermanfaat bagi pembaca dan penulis menerima kiritk serta saran agar dapat dijadikan referensi demi pengembangan ke arah yang lebih baik.

Bandung, 29 Juli 2023

Penulis

UCAPAN TERIMAKASIH

Dalam penyusunan buku dan penelitian ini penulis telah melalui perjalanan panjang serta banyak hambatan yang dihadapi sehingga dapat menyelesaikan penulisan buku Tugas Akhir. Oleh karena itu, dengan penuh kerendahan hati pada kesempatan ini penulis ingin mengucapkan terimakasih sebanyak-banyaknya atas dukungan, bimbingan dan bantuan kepada:

1. Allah SWT dan Ida Sang Hyang Widhi Wasa, karena atas rahmat, berkah dan karunia-Nya yang berlimpah sehingga penulis dapat menyelesaikan tugas akhir *capstone design*.
2. Bapak Nur Ibrahim, S.T., M.T. selaku pembimbing pertama yang telah banyak memberikan bimbingan, arahan, motivasi, kesabaran dan meluangkan waktu untuk membantu penulis dalam menyelesaikan tugas akhir *capstone design*.
3. Ibu R. Yunendah Nur Fu'adah, S.T., M.T. selaku pembimbing kedua yang telah memberikan bimbingan, arahan, dan meluangkan waktu untuk membantu penulis dalam menyelesaikan tugas akhir *capstone design*.
4. Ibu Nor Kumalasari Caecar Pratiwi, S.T., M.T. selaku pembimbing ketiga yang telah memberikan bimbingan, arahan, kesabaran dan meluangkan waktu untuk membantu penulis dalam menyelesaikan tugas akhir *capstone design*.
5. Keluarga dari Aulia Chusnyriani Sani Zulkarnaen, yaitu Bapak Capt. H. Zulkarnaen, M.Mar., Ibu Hj. Suwari Puji Lestari, S.E., dan ade G'rallsyah Jatra Kelana Zulkarnaen yang selalu memberikan dukungan secara moril dan materil serta selalu mendoakan yang tiada tara sehingga penulis dapat menyelesaikan tugas akhir ini.
6. Kedua orang tua beserta adik dari I Gusti Ngurah Rejski Ariantara Putra yaitu Bapak I Gusti Ngurah Antara, S.T., Ibu I Gusti Ayu Aryati, dan I Gusti Ayu Dera Puspitasari dan seluruh keluarga besar dari I Gusti Ngurah Rejski Ariantara Putra yang selalu mendoakan, mendukung, membantu dan menyemangati dalam pembuatan tugas akhir ini.

7. Kedua orang tua penulis Rahmawati Hidayah, Bapak Drs. Gatot Ariyanto dan Ibu Dra. Arifatul Munawaroh dan mba Kumala Hayati, S.I.Kom.,M.Med.Kom. yang sangat *support*, memberikan bimbingan, mendukung dalam kegiatan apapun dan selalu memberikan doa terbaik kepada penulis sehingga dapat menyelesaikan Tugas Akhir hingga selesai.
8. Keluarga dari Nada Fauzia Reviana yakni Bapak Ir., Indradi Vembri Hartanto, Ibu Retno Widayanti, S.H., dan Mas Naufal Abdurrasyid Reviano, S.P. yang telah memberikan kasih sayang, semangat, disertai doa tanpa henti untuk meraih keberhasilan pendidikan serta karir yang penulis jalani.
9. Teman kelas Nikah Aja Yuk, yaitu Qisthi, Rara, Apri, Fia, Indah, Erni, Farah, Roro yang memiliki banyak tingkah laku dengan tawa, canda, suka dan duka serta dukungan yang luar biasa dari awal masa perkuliahan hingga tamat. Kenangan dengan sejuta cerita di masa perkuliahan.
10. Keluarga besar TT-43-01 yang telah menjadi keluarga serta teman perjuangan dalam suka dan duka.
11. Keluarga besar SISKOM 2021/2022 yang telah memberikan support, cerita dan asisten MEN yang telah memberikan warna dan dukungan semasa perkuliahan.
12. Keluarga besar MobileCom yang telah mensupport segala perkuliahan.
13. Keluarga besar DPA-HMTT Parlemen Elite Dirga beserta Bang Rucidi dan Bang Aldi yang telah memberikan penulis pengalaman serta pembelajaran selama satu periode kepengurusan.
14. Cua, Chessa, Nabila, Arum, Ashila, Itta, Via, Amira, selaku sahabat Nada Fauzia Reviana sejak SMA yang masih menemani dan merangkul penulis di saat senang maupun sedih.
15. Teman-teman dari I Gusti Ngurah Rejski Ariantara Putra yaitu Made Teja Krisna Devana, Indah Putri Utami, Yunita, dan Sari yang sudah membantu, memberikan motivasi, dan semangat dalam pembuatan tugas akhir ini.
16. Lelaki yang telah menemani penulis selama tiga tahun masa perkuliahan. Terima kasih, semoga hal baik selalu datang di kehidupanmu.
17. Anak kontrakan Grand Imperial Cikoneng yang telah memberikan tempat untuk mengerjakan tugas akhir.

DAFTAR ISI

| | |
|---|----|
| LEMBAR PENGESAHAN | 1 |
| BUKU CAPSTONE DESIGN | 1 |
| LEMBAR PERNYATAAN ORISINALITAS | 2 |
| ABSTRAK | 4 |
| ABSTRACT..... | 5 |
| KATA PENGANTAR | 6 |
| UCAPAN TERIMA KASIH..... | 7 |
| DAFTAR ISI..... | 9 |
| DAFTAR GAMBAR | 13 |
| DAFTAR TABEL..... | 14 |
| DAFTAR SINGKATAN | 16 |
| BAB 1 USULAN GAGASAN | 17 |
| 1.1 Latar Belakang Masalah..... | 17 |
| 1.2 Informasi Pendukung Masalah..... | 19 |
| 1.3 Analisis Umum | 20 |
| 1.3.1 Aspek Ekonomi | 20 |
| 1.3.2 Aspek Edukatif | 20 |
| 1.4 Kebutuhan yang Harus Dipenuhi | 21 |
| 1.4.1 Perangkat Lunak | 21 |
| 1.4.2 Perangkat Keras | 21 |
| 1.5 Solusi Sistem yang Diusulkan..... | 22 |
| 1.5.1 Karakteristik Produk | 22 |
| 1.5.2 Skenario Penggunaan | 24 |
| 1.6 Tujuan | 25 |

| | | |
|-------------------------------------|---|----|
| 1.7 | Kesimpulan dan Ringkasan CD-1 | 26 |
| BAB 2 DESAIN KONSEP SOLUSI | | 27 |
| 2.1 | Spesifikasi Produk..... | 27 |
| 2.2 | Verifikasi..... | 33 |
| 2.2.1 | Verifikasi Spesifikasi 1 | 33 |
| 2.2.2 | Verifikasi spesifikasi 2..... | 34 |
| 2.3 | Kesimpulan dan Ringkasan CD-2..... | 34 |
| BAB 3 DESAIN RANCANGAN SOLUSI | | 36 |
| 3.1 | Konsep Sistem..... | 36 |
| 3.1.1 | Pilihan Sistem | 36 |
| 3.1.2 | Analisis | 38 |
| 3.1.3 | Sistem yang akan Dikembangkan | 41 |
| 3.2 | Rencana Desain Sistem | 42 |
| 3.3 | Pengujian Komponen (Kalibrasi)..... | 46 |
| 3.4 | Jadwal Pengerjaan | 50 |
| 3.5 | Kesimpulan dan Ringkasan CD-3 | 51 |
| BAB 4 IMPLEMENTASI | | 53 |
| 4.1 | Implementasi Sistem | 53 |
| 4.1.1 | Deep Learning | 53 |
| 4.1.2 | Mobile Application | 61 |
| 4.1.3 | BatiQu Box | 62 |
| 4.2 | Analisis Pengerjaan Implementasi Sistem | 63 |
| 4.3 | Hasil Akhir Sistem | 65 |
| 4.4 | Kesimpulan dan Ringkasan CD-4..... | 67 |
| BAB 5 PENGUJIAN SISTEM | | 69 |
| 5.1 | Skema Pengujian Sistem | 69 |

| | | |
|-------|---|-----|
| 5.1.1 | Pengujian SUS keseluruhan fitur | 69 |
| 5.1.2 | Pengujian Deep Learning | 69 |
| 5.2 | Proses Pengujian | 70 |
| 5.2.1 | <i>Pre-processing</i> | 70 |
| 5.2.2 | <i>Feature Extraction</i> | 71 |
| 5.2.3 | <i>Classification</i> | 73 |
| 5.2.4 | <i>Mobile Application</i> | 74 |
| 5.2.5 | Pengujian pada Fitur Utama Aplikasi | 74 |
| 5.2.6 | Pengujian pada Fitur Tambahan Aplikasi | 76 |
| 5.2.7 | Pengujian pada seluruh sistem Aplikasi | 77 |
| 5.3 | Hasil Pengujian | 78 |
| 5.3.1 | <i>Pre-processing</i> | 78 |
| 5.3.2 | <i>Feature Extraction</i> | 82 |
| 5.3.3 | <i>Classification</i> | 86 |
| 5.3.4 | <i>Mobile Application</i> | 88 |
| 5.3.5 | Pengujian Keseluruhan Fungsi dari Aplikasi BatiQu | 95 |
| 5.4 | Analisis Hasil Pengujian | 98 |
| 5.4.1 | Analisis Pengujian <i>Pre-processing</i> | 98 |
| 5.4.2 | Analisis Pengujian <i>Feature Extraction</i> | 99 |
| 5.4.3 | Analisis Pengujian <i>Classification</i> | 101 |
| 5.4.4 | Analisis Pengujian Halaman Informasi <i>Mobile Application</i> 104 | |
| 5.4.5 | Analisis Pengujian Keseluruhan Fungsi dari <i>Mobile Application</i> 105 | |
| 5.5 | Kesimpulan dan Ringkasan CD-5 | 105 |
| 5.5.1 | Kesimpulan | 105 |

| | | |
|-------|---------------------------|-----|
| 5.5.2 | Saran | 105 |
| | DAFTAR PUSTAKA | 107 |
| | LAMPIRAN CD-1 | 113 |
| | LAMPIRAN CD-2 DAN 3 | 124 |
| | LAMPIRAN CD-4 DAN 5 | 125 |

DAFTAR GAMBAR

| | |
|--|-----|
| Gambar 1.1 Penggunaan Produk A | 24 |
| Gambar 1.2 Skenario Penggunaan Produk B | 25 |
| Gambar 2.1 <i>Box</i> Pengujian Sampel Batik | 28 |
| Gambar 2.2 Mock-up Homepage dan logo aplikasi BatiQu | 29 |
| Gambar 2.3 Tampilan pengambilan gambar dan halaman informasi..... | 30 |
| Gambar 2.4 <i>Page About Us</i> | 31 |
| Gambar 3.1 lapisan Metode Convolutional Neural Network | 42 |
| Gambar 3.2 Flowchart Program CNN | 44 |
| Gambar 3.3 Flowchart Aplikasi BatiQu | 45 |
| Gambar 3.4 (a) Konvolusi arsitektur MobileNet (b) Arsitektur MobileNet V1 dengan model parameter | 47 |
| Gambar 3.5 (a) Arsitektur ResNet125 (b) Model parameter..... | 49 |
| Gambar 4.1 (a)batik tambal (b)batik parang (c)batik nitik (d)batik megamendung (e)batik kawung (f)batik ceplok | 54 |
| Gambar 4.2 <i>Flowchart Pre-processing</i> dengan Augmentasi data..... | 55 |
| Gambar 4.3 <i>Flowchart Feature Extraction</i> | 57 |
| Gambar 4.4 Flowchart alur proses klasifikasi | 58 |
| Gambar 4.5 Implementasi Box secara real | 63 |
| Gambar 4.6 Implementasi Sistem..... | 63 |
| Gambar 4.7 Tampilan kamera dan gallery | 65 |
| Gambar 4.8 Halaman Informasi | 66 |
| Gambar 4.9 Halaman About Us | 67 |
| Gambar 5.1 Skema pengujian Deep Learning..... | 69 |
| Gambar 5.2 halaman pengambilan gambar | 89 |
| Gambar 5.3 Hasil Pengujian About Us | 94 |
| Gambar 5.4 Perhitungan score metode SUS | 95 |
| Gambar 5.5 (a)Grafik visualisasi accuracy (b)Grafik visualisasi loss (c)Hasil Confusion matrix..... | 103 |

DAFTAR TABEL

| | |
|---|-----|
| Tabel 2.1 Spesifikasi Produk | 32 |
| Tabel 2.2 Verifikasi spesifikasi pertama | 33 |
| Tabel 2.3 Verifikasi spesifikasi kedua | 34 |
| Tabel 3.1 (a) Kriteria Ekonomi (b) Kriteria Sistem (c) Kriteria ketersediaan komponen (d) Kriteria Penggunaan/Pengguna (e) Kriteria Performa..... | 39 |
| Tabel 3.2 Jadwal Pengerjaan Capstone Design | 50 |
| Tabel 3.3 Final Jadwal Pengerjaan Capstone Design..... | 51 |
| Tabel 4.1 Tabel Confusion Matrix | 60 |
| Tabel 4.2 Confusion Matrix 6 class..... | 61 |
| Tabel 5.1 Spesifikasi Hardware..... | 69 |
| Tabel 5.2 Pertanyaan Kuesioner..... | 77 |
| Tabel 5.3 (a) Hasil Pengujian Pre-processing MobileNet V1 (b) Hasil Pengujian Pre-processing ResNet152 V2 (c) Hasil Pengujian Pre-processing ResNet101 V2 (d) Hasil Pengujian Pre-processing ResNet50 V2 | 78 |
| Tabel 5.4 (a)Hasil Pengujian Target size MobileNet V1 (b)Hasil Pengujian Target size ResNet152 V2 (c) Hasil Pengujian Target size ResNet101 V2 (d) Hasil Pengujian Target size ResNet50 V2 | 81 |
| Tabel 5.5 Hasil Pengujian Include Top | 82 |
| Tabel 5.6 Hasil Pengujian Pooling Layer..... | 83 |
| Tabel 5.7 Hasil Pengujian Include Top | 83 |
| Tabel 5.8 Hasil Pengujian Pooling Layer..... | 84 |
| Tabel 5.9 Hasil Pengujian Optimizer | 86 |
| Tabel 5.10 Hasil Pengujian Learning Rate..... | 87 |
| Tabel 5.11 Hasil Pengujian Batch Size | 87 |
| Tabel 5.12 Hasil halaman informasi..... | 90 |
| Tabel 5.13 Hasil pengujian model Aplikasi | 93 |
| Tabel 5.14 Pilihan Jawaban kuisisioner | 95 |
| Tabel 5.15 Hasil rekaptulasi SUS..... | 97 |
| Tabel 5.16 Rekap hasil pengujian pre-processing | 98 |
| Tabel 5.17 Rekap Hasil Pengujian Extraction Feature | 100 |
| Tabel 5.18 Rekap hasil pengujian classification..... | 102 |

| | |
|--|-----|
| Tabel 5.19 Hasil performansi Jenis batik | 103 |
| Tabel 5.20 Tabel biaya penelitian Tugas Akhir | 125 |

DAFTAR SINGKATAN

| | |
|-----------|--|
| AI | : <i>Artificial Intelligence</i> |
| Adam | : <i>Adaptive Moment Estimation</i> |
| CNN | : <i>Convolutional Neural Network</i> |
| IACI | : <i>Indonesian Archipelago Cultural Initiative</i> |
| LED | : <i>Light Emitting Diode</i> |
| MobileNet | : <i>Mobile Network</i> |
| DL | : <i>Deep Learning</i> |
| Nadam | : <i>Nesterov-accelerated Adaptive Moment Estimation</i> |
| ResNet | : <i>Residual Network</i> |
| SGD | : <i>Stochastic Gradient Descent</i> |
| SUS | : <i>System Usability Scale</i> |
| UNESCO | : <i>United Nations Educational, Scientific, and Cultural Organization</i> |
| XML | : <i>Extensible Markup Language</i> |
| RMSprop | : <i>Root Mean Square Propagation</i> |

BAB 1

USULAN GAGASAN

1.1 Latar Belakang Masalah

Batik merupakan kain tradisional dan salah satu warisan turun menurun yang dimiliki oleh bangsa Indonesia. Batik berasal dari Bahasa Jawa yaitu kata “amba” (menulis) dan “nitik” (membuat titik atau dot) [1]. Batik saat ini masih terus berkembang dan dilestarikan sehingga masih menjadi identitas budaya bangsa Indonesia. Keberadaan batik telah diakui oleh dunia dan ditetapkan *United Nations Educational, Scientific, and Cultural Organization* (UNESCO) sebagai hak kebudayaan intelektual bangsa Indonesia pada tanggal 2 Oktober 2009 [2]. Dengan diakuihnya batik sebagai kebudayaan bangsa Indonesia, hal ini menjadikan adanya Hari Batik Nasional yang jatuh tiap tanggal 2 Oktober. Di Indonesia sendiri penggunaan batik sangat beragam salah satu contoh kecil dalam penerapan batik saat ini digunakan untuk seragam di beberapa instansi atau pun sekolah yang mewajibkan penggunaan seragam pada hari-hari tertentu. Perkembangan batik pun dapat dijadikan berbagai aksesoris yang menawan dan penuh kreativitas. Berdasarkan penelitian yang dilakukan oleh Bandung FE *Institute* dan Surya *Research International* melalui organisasi *Indonesian Archipelago Cultural Initiative* (IACI), motif batik yang ada di Indonesia mencapai 5.849 motif batik [3]. Dari banyaknya motif batik yang tersebar dari seluruh daerah di Indonesia, perbedaan motif batik pun memiliki esensi dan makna tersendiri yang terkandung dari para leluhur yang mengukir goresan indah pada kain tersebut dan memiliki makna simbolis yang unik.

Dengan banyaknya angka motif batik yang tercatat, hal ini memerlukan suatu sistem yang membantu untuk mengklasifikasikan jenis batik tersebut ke dalam beberapa kelas tertentu. Klasifikasi ini dapat berdasarkan bentuk motifnya seperti geometri, non geometri, dan beberapa motif lainnya [4]. Keberagaman motif batik memperumit identifikasi karakter yang terkandung dalam objek dan basis data yang dikelompokkan. Tahap awal sebelum proses identifikasi adalah ekstraksi ciri dimana objek yang telah diambil untuk gambaran karakteristik objek tersebut dapat

dikenali. Identifikasi pada pengenalan motif citra telah dilakukan oleh beberapa peneliti terdahulu menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur ResNet152 V2. Pengujian sebelumnya pada arsitektur ini menggunakan *hyperparameter* seperti *optimizer* adam serta fungsi *binary cross entropy* untuk mengurangi data *loss* dengan nilai akurasi mencapai 97% [5]. Selanjutnya, terdapat penelitian menggunakan RMSprop sebagai *optimizer* dan *dropout* untuk mengurangi *overfitting* yang memiliki nilai 0.1 dalam arsitektur MobileNet V1. Hasil yang didapatkan dari pengujian ini mencapai akurasi latihan hingga 86% dan akurasi validasi bernilai 77% [6]. Selain itu, penelitian lain didapati bahwa adanya komparasi atau perbandingan antara arsitektur ResNet152 V2 dengan MobileNet V1 yang menyatakan bahwa hasil akurasi dari ResNet152 V2 lebih baik [7]. Pada penelitian tersebut dipaparkan hasil dari arsitektur ResNet152 V2 dengan *optimizer* Adagrad mencapai akurasi sebesar 91.62% dengan nilai loss 0.2518 [7]. Sedangkan, untuk pengujian pada MobileNet V1 dengan *optimizer* Adamax memiliki hasil akurasi 89.52% dan nilai loss sebesar 0.3144 [7].

Pada penelitian ini akan dirancang suatu sistem aplikasi yang dapat mengidentifikasi ke dalam enam kelas berbasis aplikasi *mobile* sehingga dapat digunakan dimana saja dan ramah terhadap pengguna untuk mengaksesnya. Aplikasi ini dirancang untuk mempermudah penggunanya dalam mengklasifikasikan jenis-jenis batik sehingga pengguna pun dapat terbantu dan teredukasi lebih mendalam. Penelitian ini akan menggunakan sampel jenis batik dari batik parang, batik tambal, batik ceplok, batik kawung, batik nitik, serta batik megamendung. Pemilihan dataset dari keenam jenis motif batik ini dilakukan dengan pengambilan kuesioner mengenai kepopuleran dan jenis motif batik yang tampak familier di kalangan mahasiswa Universitas Telkom. Dari hasil kuesioner didapatkan bahwa keenam jenis batik tersebut memiliki tingkat popularitas yang lebih tinggi dibanding jenis batik lain seperti batik asmat, batik tubo, batik gentongan, dan batik simbut. Dari keenam jenis motif batik tersebut telah terkumpul sebanyak 660 sampel batik sebagai dataset. Kemudian dataset tersebut diaugmentasi hingga berjumlah 3.300 sampel batik untuk klasifikasi pengenalan pada komputer. Tujuan dari klasifikasi batik yang akan dirancang adalah membagi citra batik ke dalam kelas-kelas yang sudah disesuaikan dengan pola motifnya.

Dimana dengan adanya klasifikasi ini dapat membantu masyarakat dalam mengidentifikasi jenis batik terutama motif-motif batik yang tersebar di Indonesia. Jika sistem ini diimplementasikan maka pengenalan batik dengan motif dan ciri khas tertentu akan lebih mudah teridentifikasi. Teknologi pada penelitian ini diharapkan dapat berkontribusi untuk melestarikan kebudayaan batik yang sangat beragam.

1.2 Informasi Pendukung Masalah

Sebagai salah satu budaya yang terkenal dari Indonesia, batik tidak luput dari pengakuan oleh negara lain. Hingga saat ini telah terjadi beberapa pengakuan sepihak dari negara lain yang sempat mengancam kelestarian batik. Hal ini disebabkan oleh adanya kemiripan antara kain motif batik asal Indonesia dengan kain bermotif dari negara lain yang menimbulkan kesalahpahaman hingga berujung pada pengakuan atas budaya batik tersebut. Akibat kejadian ini pemerintah mulai serius dalam menguatkan status batik Indonesia dalam skala Nasional dan Internasional dengan mengajukan budaya batik kepada pihak UNESCO dan membuat produk hukum yang dapat menjadi jaminan untuk kelestarian batik.

Pada tanggal 2 Oktober 2009 UNESCO secara resmi mengakui batik sebagai Warisan Kemanusiaan untuk Budaya Lisan dan Nonbendawi (*Masterpieces of the Oral and the Intangible Heritage of Humanity*) [2]. Batik dianggap sebagai simbol dan budaya yang selalu menyertai kehidupan masyarakat Indonesia dari sejak lahir sampai meninggal dunia [8]. Pemerintah juga ikut serta dalam menjaga kelestarian batik dengan mengeluarkan produk hukum. Ada banyak produk hukum yang telah dikeluarkan oleh pemerintah Indonesia yang berkaitan dengan batik di antaranya [9]:

- a. UU Nomor 19 Tahun 2002 tentang Hak Cipta.
- b. Keputusan Presiden Republik Indonesia Nomor 33 Tahun 2009 Tentang Hari Batik Nasional.
- c. Peraturan Menteri Perindustrian 74/M-IND/PER/9/2007 Tentang Penggunaan Batikmark.

1.3 Analisis Umum

Adapun analisis dibuat dengan beberapa aspek-aspek yang akan digunakan sebagai pedoman dalam pembuatan program yang mana pengidentifikasian jenis batik ini sangat bermanfaat kedepannya bagi masyarakat. Berikut aspek-aspek yang digunakan:

1.3.1 Aspek Ekonomi

Batik di Indonesia memiliki banyak macam dan memiliki sejarah yang panjang dari setiap jenis batik yang ada. Maka dari itu pemerintah Indonesia memasukkan batik ke UNESCO pada tanggal 2 Oktober 2009. UNESCO mengakui batik sebagai Warisan Kemanusiaan untuk Budaya Lisan dan Nonbendawi (*Masterpieces of the Oral and Intangible Heritage of Humanity*) sehingga pada tanggal itu disebut sebagai Hari Batik Nasional.

Batik juga termasuk salah satu karya yang dilindungi Hak Cipta dimana diatur dalam Pasal 40 Undang – Undang Hak Cipta tahun 2014 dengan jangka waktu perlindungan hak cipta atas karya seni batik kontemporer berlaku selama 70 (tujuh puluh) tahun [10]. Karya seni batik yang dimaksud dalam Undang – Undang Hak Cipta adalah motif batik kontemporer yang bersifat inovatif, masa kini, dan bukan tradisional. Batik dilindungi karena memiliki nilai seni, baik dalam kaitannya dengan gambar, corak, maupun komposisi warna.

Dengan adanya alat ini, tidak hanya mengetahui jenis batik yang ditampilkan namun juga mengetahui sejarah dari batik tersebut. Sehingga, ketika mengetahui ada klaim sepihak dari negara lain tentang jenis batik ini, maka bisa menunjukkan sejarah yang menunjukkan keberadaan jenis batik ini.

1.3.2 Aspek Edukatif

Jenis batik di Indonesia berjumlah 5.489 corak yang tersebar dari Aceh ke Papua. Dari berbagai macam corak yang tersebar tentu memiliki filosofi yang berbeda dan belum tentu masyarakat dapat menghafal secara rinci sebuah jenis batik hanya dengan sekali melihat jenis motif batik. Maka dari itu penelitian ini merancang sebuah produk berupa aplikasi agar mempermudah masyarakat untuk mengetahui jenis batik yang ada. Dengan adanya aplikasi ini juga dapat

mengedukasi mengenai sejarah lengkap dan filosofi yang ada di dalam batik itu sebagai ilmu tambahan agar mengerti arti dari setiap corak batik yang ada.

1.4 Kebutuhan yang Harus Dipenuhi

Berikut ini merupakan beberapa kebutuhan yang harus dipenuhi dalam penelitian yang akan dirancang:

1.4.1 Perangkat Lunak

- a. Pada *Software Desktop*
 - a) Sistem Operasi *Microsoft Windows 11*
 - b) *Google Colab*
 - c) *Android Studio Electric Flamingo* | 2022.2.1
 - d) Metode CNN (*Convolutional Neural Network*)
 - e) Dataset
- b. Pada *Smartphone*

Software pada *smartphone* yang digunakan adalah sistem operasi *Android* versi 10 *Quince tart*.

1.4.2 Perangkat Keras

- a. Pada *Desktop*
 - a) Laptop *Lenovo Legion Y7000*
 - b) *Processor* : *Intel Core i7-9750H*
 - c) Kartu Grafis : *NVIDIA GeForce GTX 1650*
 - d) RAM : *16 GB DDR4*
 - e) *Storage* : *512 GB SSD M.2 NVME + 1 TB HDD*
- b. Pada *Smartphone* dan *Box*
 - a) *Infinix Zero 5G*
 - b) *Processor* : *MediaTek Dimensity 900*
 - c) RAM : *8 GB*
 - d) Memori internal : *128GB*
 - e) Memori eksternal : *microSDXC (dedicated slot)*
 - f) Ukuran *Box* : *12 x 12 x 17 cm*

1.5 Solusi Sistem yang Diusulkan

Berdasarkan latar belakang masalah yang ada, terdapat dua pilihan solusi sistem yang dapat diterapkan di antaranya menggunakan aplikasi identifikasi jenis batik berbasis *online* dengan menggunakan teknologi AI (*Artificial Intelligence*). Teknologi AI adalah sebuah teknologi yang digunakan untuk meniru kecerdasan yang dimiliki oleh makhluk hidup untuk menyelesaikan sebuah masalah [11]. Teknologi AI banyak sekali digunakan pada aplikasi identifikasi gambar, suara hingga perasaan dari makhluk hidup. Kelebihan dari teknologi ini adalah hasil identifikasi yang akurat dan baik, cara kerja lebih cepat dan hasil identifikasi yang dapat didokumentasikan [12]. Solusi sistem lainnya adalah aplikasi identifikasi jenis batik berbasis *offline*. Kelebihan dari aplikasi ini adalah dapat diakses dimana saja dan kapan saja tanpa membutuhkan internet untuk identifikasi jenis batik. Dari pemaparan dua solusi sistem ini, diperlukan perbandingan untuk mengetahui solusi sistem yang optimal.

1.5.1 Karakteristik Produk

1.5.1.1 Karakteristik Produk A

- Fitur Utama :

Solusi sistem yang akan dirancang adalah aplikasi identifikasi jenis batik dengan menggunakan teknologi AI (*Artificial Intelligence*) dan perlu akses internet. Aplikasi dengan penggunaan teknologi AI ini memiliki beberapa kelebihan seperti hasil identifikasi yang akurat dan baik, cara kerja lebih cepat dan hasil identifikasi yang dapat didokumentasikan [12] sehingga rancangan aplikasi seperti ini sangat umum digunakan. Rancangan aplikasi ini menggunakan kamera *handphone* untuk pengambilan gambar dan menggunakan teknologi AI untuk klasifikasi jenis batik.

- Fitur Dasar :

1. Fitur kamera yang dilengkapi dengan teknologi AI.
2. Halaman informasi yang tertera jelas tingkat akurasi dan dapat menunjukkan tiga jenis motif batik termasuk tiga jenis pembuatan batik.

- Fitur Tambahan :

1. Terdapat *E-Commerce* yang khusus menjual kain batik.

2. Terdapat artikel – artikel khusus tentang kain batik dan beberapa jenis batik.
 3. Terdapat halaman akun.
- Sifat Solusi yang Diharapkan :
 1. Mudah untuk identifikasi jenis batik apa saja karna sudah terbantu dengan teknologi AI.
 2. Mudah untuk di akses karena hanya cukup *install* dan bisa langsung identifikasi jenis batik.
 3. Hasil identifikasi yang di paparkan cukup jelas karena berbentuk persen.
 4. Dipermudah dengan ada nya fitur tambahan yang dapat langsung terhubung dengan *E-Commerce* penjual kain batik.

1.5.1.2 Karakteristik Produk B

- Fitur Utama :

Solusi sistem kedua yang akan dirancang adalah aplikasi identifikasi jenis batik dengan menggunakan teknologi *Deep Learning* dan bisa diakses tanpa perlu menggunakan internet. Rancangan aplikasi ini menggunakan kamera dan galeri sebagai pengambilan gambar kain batik dan menggunakan *Deep Learning* sebagai metode klasifikasi jenis batik. *Deep Learning* yang digunakan adalah model yang telah dilatih menggunakan metode yang cocok pada aplikasi [13].

- Fitur Dasar :
 1. Fitur kamera yang dilengkapi dengan metode *Deep Learning*.
 2. Fitur galeri yang dilengkapi dengan metode *Deep Learning*.
 3. Halaman informasi yang dilengkapi dengan pengetahuan singkat dari jenis batik yang teridentifikasi.

- Fitur Tambahan :

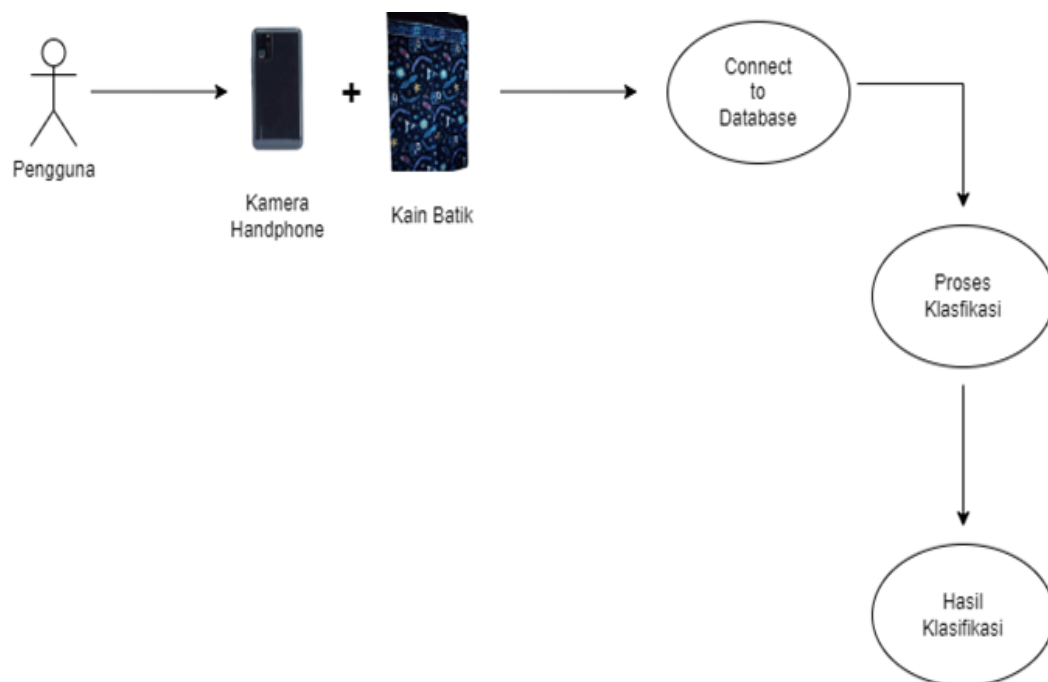
Halaman *About Us* yang berisi definisi aplikasi, fungsi serta data diri dari pembuat aplikasi.

- Sifat Solusi yang Diharapkan :
 1. Mudah untuk diakses karna cukup *install* dan bisa langsung identifikasi jenis batik.
 2. Tidak membutuhkan perawatan yang intensif dan mengefisiensikan waktu dalam pengambilan gambar.

1.5.2 Skenario Penggunaan

Skenario penggunaan menjelaskan mengenai pilihan sistem yang akan digunakan dalam pembuatan aplikasi. Aplikasi ini dirancang agar memudahkan pengguna untuk mengklasifikasikan beberapa motif batik dan mengetahui penjelasan singkat dari batik tersebut. Aplikasi ini dapat menghasilkan keluaran berupa identifikasi motif batik, contohnya apakah batik ini merupakan batik parang, atau batik tambal, batik ceplok, batik kawung, batik nitik, dan batik megamendung.

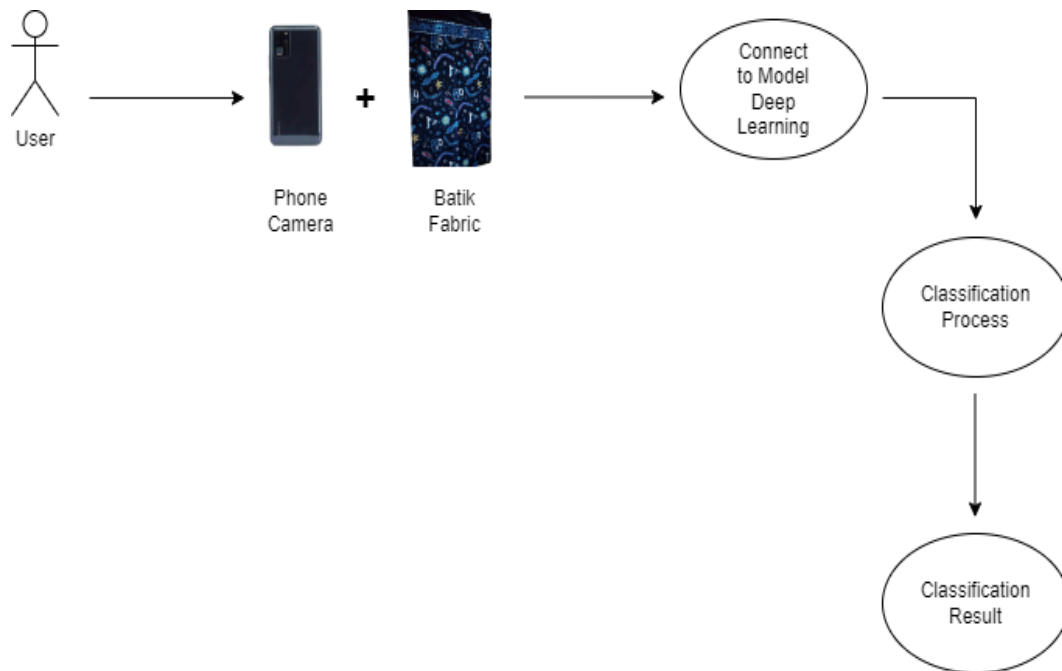
1.5.2.1 Skenario penggunaan Produk A



Gambar 1.1 Penggunaan Produk A

Gambar 1.1 di atas menunjukkan alur identifikasi batik menggunakan solusi sistem produk A. Pengguna dapat identifikasi jenis batik apapun karena produk A dibantu dengan teknologi AI (*Artificial Intelligence*). Alur identifikasi dimulai dari pengguna yang telah memiliki produk A dan kain batik. Selanjutnya, pengguna bersiap mengambil gambar untuk identifikasi jenis batik dengan kain yang sudah dipersiapkan. Setelah gambar diambil, produk A menunggu untuk terhubung ke *database* dan dilanjut dengan proses klasifikasi. Setelah proses klasifikasi selesai dan hasil klasifikasi sudah keluar, maka pengguna bisa melihat halaman informasi dari jenis batik.

1.5.2.2 Skenario penggunaan Produk B



Gambar 1.2 Skenario Penggunaan Produk B

Gambar 1.2 di atas menunjukkan alur identifikasi batik menggunakan produk B. Pengguna dapat melakukan identifikasi jenis batik dimanapun dan kapan pun tanpa perlu menggunakan akses internet. Alur identifikasi dimulai dari pengguna yang telah memiliki produk B dan kain batik. Selanjutnya, pengguna bersiap mengambil gambar untuk identifikasi jenis batik dengan kain yang sudah disiapkan. Setelah gambar diambil, produk B menunggu untuk terhubung pada metode *Deep Learning* yang telah dilatih sebelumnya bersamaan dengan dilakukannya proses klasifikasi. Setelah proses klasifikasi selesai dan hasil klasifikasi telah muncul, maka pengguna dapat melihat halaman hasil klasifikasi dari jenis batik.

1.6 Tujuan

Tujuan dari pengerjaan tugas akhir ini adalah :

1. Penulisan dokumen Capstone Design ini ditujukan untuk memenuhi kewajiban pada kelas Proposal Tugas Akhir.

2. Melakukan analisis hasil sehingga dapat diidentifikasi berdasarkan motif dan jenis Batik Indonesia serta melakukan analisis pada *accuracy* dan *loss* berdasarkan parameter metode yang digunakan dalam *Deep Learning*.
3. Membantu masyarakat agar lebih banyak mengenal jenis ragam batik Indonesia.
4. Merancang program dengan memberikan informasi mengenai jenis batik berdasarkan motif-motif dengan menggunakan metode *Deep Learning*.

1.7 Kesimpulan dan Ringkasan CD-1

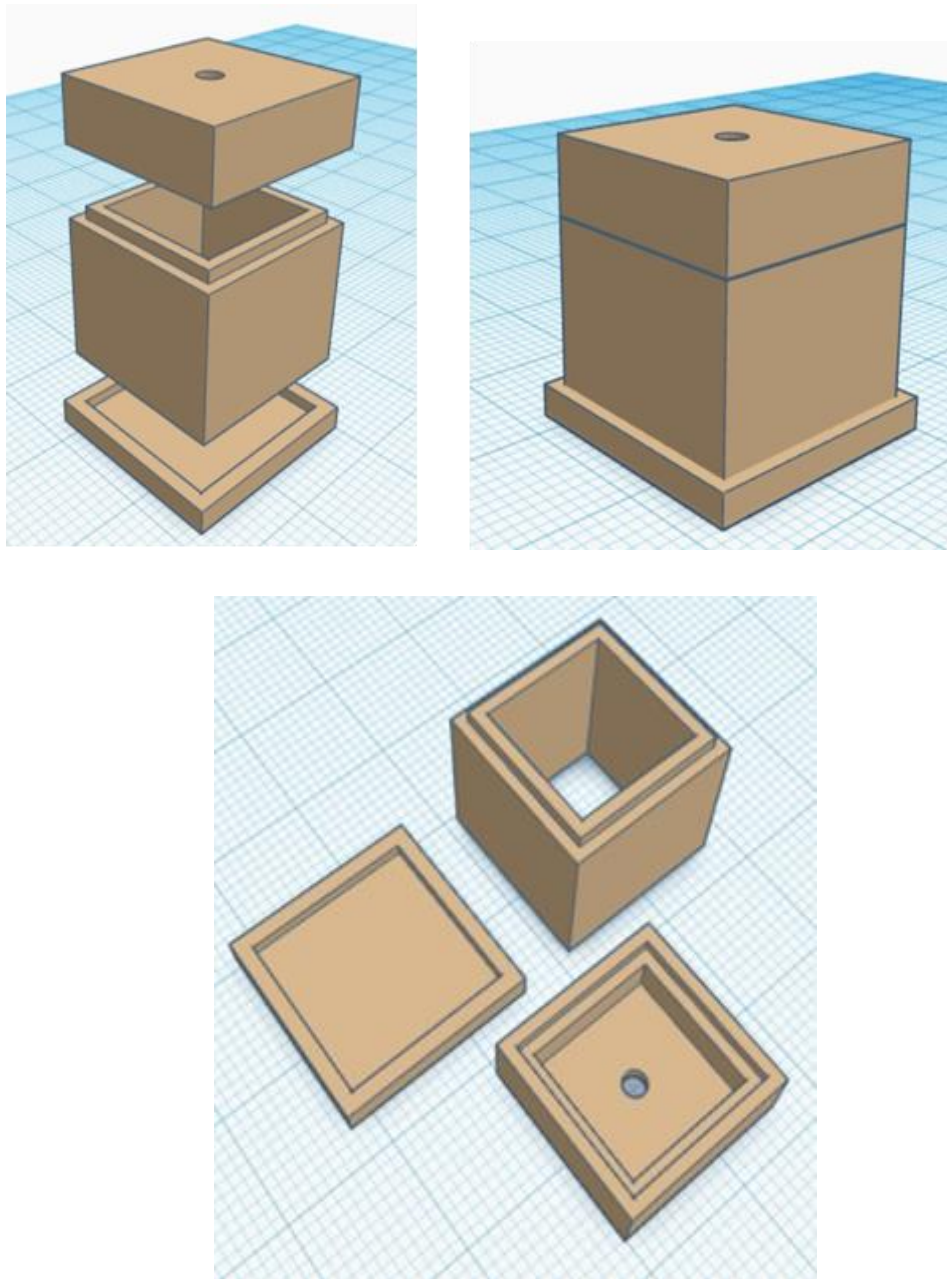
Batik merupakan budaya Indonesia yang telah diakui dunia dan memiliki jenis motif yang beragam. Dengan banyaknya angka motif batik, hal ini memerlukan suatu sistem yang membantu dalam identifikasi jenis batik. Pada penelitian ini akan dirancang suatu sistem aplikasi yang mampu mengidentifikasi jenis batik ke dalam enam kelas. Enam kelas tersebut didapatkan dari hasil kuesioner mengenai kepopuleran dari beberapa jenis batik yang ditujukan pada mahasiswa Universitas Telkom. Hasil kuesioner menunjukkan bahwa terdapat enam batik yang populer di kalangan mahasiswa yaitu batik parang, batik tambal, batik ceplok, batik kawung, batik nitik, serta batik megamendung. Enam jenis motif batik tersebut akan menjadi solusi sistem yang dirancang. Terdapat dua produk untuk solusi sistem yaitu produk A dan produk B. Dimana pada kedua produk tersebut memiliki beberapa fitur utama, fitur dasar, fitur tambahan serta solusi yang diharapkan. Dalam penjelasan fitur-fitur yang ada terdapat penjelasan mengenai skema atau skenario dari penggunaan kedua produk tersebut. Penelitian ini bertujuan untuk menganalisis hasil klasifikasi guna mengidentifikasi enam jenis batik di Indonesia. Hasil analisis tersebut diharapkan dapat membantu meningkatkan pemahaman masyarakat terhadap ragam batik Indonesia. Selain itu, penelitian ini menganalisis hasil *accuracy* dan *loss* berdasarkan parameter metode yang digunakan dalam *Deep Learning*.

BAB 2

DESAIN KONSEP SOLUSI

2.1 Spesifikasi Produk

Pada Tugas Akhir ini, aplikasi BatiQu merupakan perangkat lunak yang akan digunakan untuk mengklasifikasikan jenis batik. Produk ini dirancang berbasis aplikasi *mobile* dengan fitur yang ramah digunakan bagi penggunaannya dan dapat diunduh lewat *Google Playstore*. Pengoperasian dalam aplikasi ini cukup mudah karena hanya memerlukan empat tombol dan bertujuan agar mempermudah masyarakat dalam mengidentifikasi jenis batik. Dari penjelasan di atas, aplikasi BatiQu memiliki fitur berupa penjelasan singkat mengenai jenis-jenis batik yang akan diuji dan dapat ditampilkan secara langsung. Hal ini merupakan inovasi yang baru untuk mengedukasi para pengguna terhadap keragaman batik di Indonesia. Aplikasi BatiQu dapat mengenali enam jenis batik yaitu batik ceplok, batik kawung, batik megamendung, batik nitik, batik parang, dan batik tambal. Perancangan perangkat lunak untuk aplikasi ini juga akan diimplementasikan ke dalam bentuk perangkat keras. Fungsi dari perangkat keras ini akan membantu proses klasifikasi batik terhadap kelas-kelasnya serta menjaga pencahayaan dan posisi pengambilan sampel agar tidak berubah. Bentuk yang akan dirancang dari perangkat keras ini berupa *box*. *Box* BatiQu tersebut dirancang berbentuk kubus dengan panjang 12 cm, lebar 12 cm, dan tinggi 17 cm menggunakan bahan kayu. Dibuatnya *box* dengan ukuran tersebut bertujuan untuk memberi jarak antara kamera *handphone* dan kain batik sebagai bahan uji. Pada bagian atas kotak diberikan ruang untuk memasang LED (*Light Emitting Diode*) sebagai penerangan sampel dan celah untuk menaruh kamera *handphone* saat proses pengambilan gambar. Gambar 2.1 merupakan *box* BatiQu yang akan digunakan dalam pengambilan sampel uji:



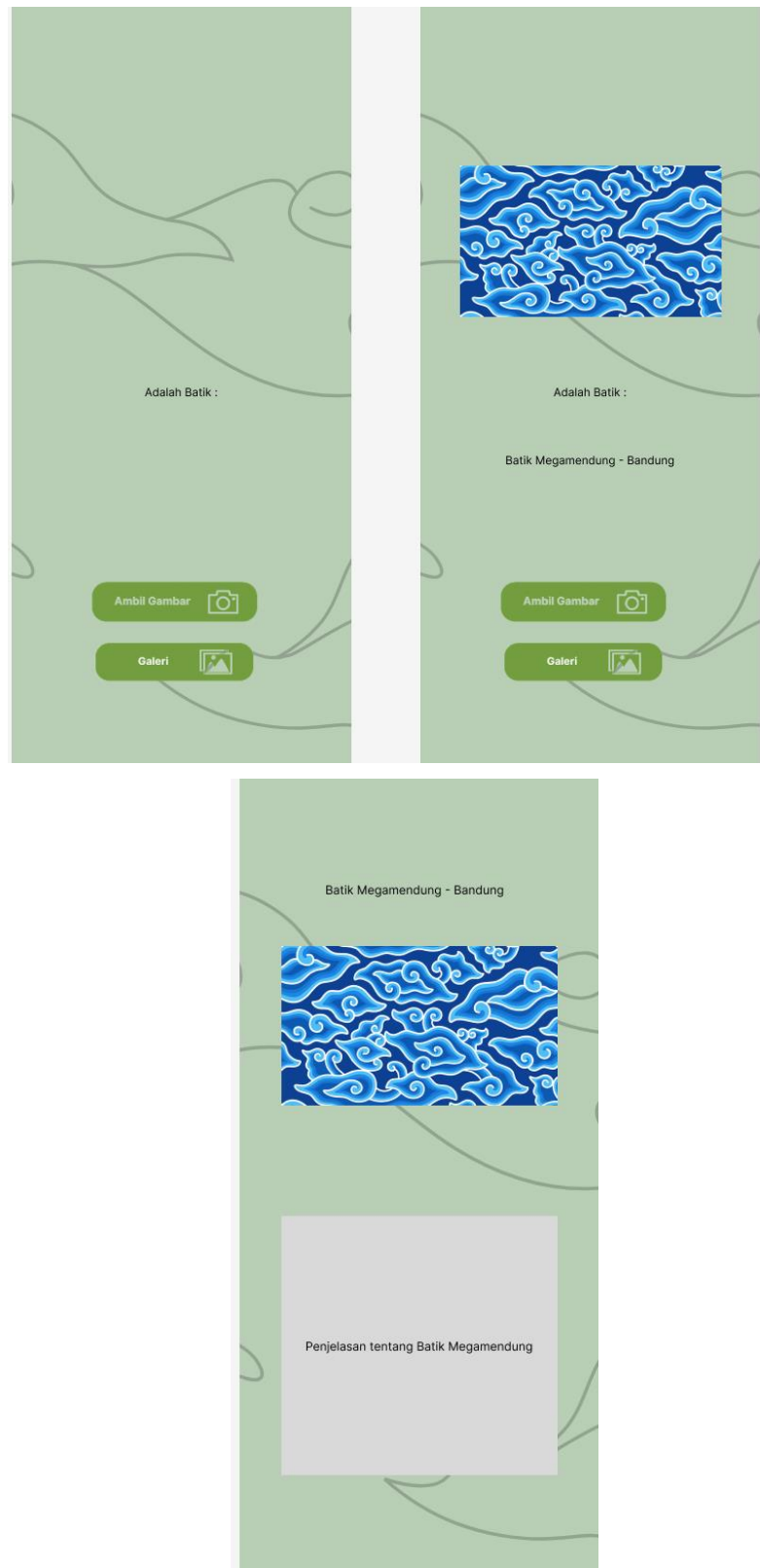
Gambar 2.1 *Box* Pengujian Sampel Batik

Pada penjelasan di bawah ini merupakan tampilan aplikasi BatiQu dengan dua tombol pada halaman utama yang tertera pada gambar 2.2:



Gambar 2.2 *Mock-up Homepage* dan logo aplikasi BatiQu

Tombol yang terlihat dari gambar di atas memiliki beragam fungsi seperti contoh pada tombol pertama yang digunakan untuk memulai prediksi jenis batik. Tombol terakhir atau “*About Us*” berfungsi untuk menampilkan informasi singkat mengenai aplikasi BatiQu dan pembuat aplikasi. Dari tampilan halaman utama yang terlihat, pengguna bisa memilih antara kedua tombol tersebut. Jika pengguna memilih tombol pertama, maka hasil yang terlihat selanjutnya akan seperti pada gambar 2.3:



Gambar 2.3 Tampilan pengambilan gambar dan halaman informasi

Pada halaman ini, pengguna bisa memulai uji klasifikasi jenis batik. Terdapat dua tombol untuk memulai pengambilan gambar jenis batik yaitu dari kamera dan

gallery. Hasil dari sampel uji akan terlihat pada halaman informasi yang dibuat sesuai dengan kelasnya seperti gambar di atas. Halaman informasi berisikan penjelasan singkat mengenai jenis batik yang terdeteksi seperti asal usul, keunikan dari pola batik dan bagaimana cara pakai dari batik tersebut.

Selanjutnya pengguna bisa tekan tombol “*back*” pada *handphone* untuk mengulangi proses dari awal dengan memotret jenis batik lain atau memilih gambar batik yang sudah ada dari *gallery*. Pengguna juga bisa kembali ke halaman utama setelah melakukan uji jenis batik. Jika pengguna memilih tombol kedua pada halaman utama atau tombol “*About Us*” maka tampilan selanjutnya akan ditampilkan pada gambar 2.4:



Gambar 2.4 Page *About Us*

Pada gambar 2.4 menunjukkan Halaman “*About Us*” berisikan informasi singkat dari aplikasi BatiQu seperti kegunaan aplikasi, alasan dibuatnya aplikasi, dan data diri dari pembuat aplikasi BatiQu. Harapan besar dengan adanya aplikasi

ini akan membantu masyarakat dalam mengidentifikasi beberapa jenis motif batik yang ada.

Tabel 2.1 Spesifikasi Produk

| No | Hal | Rincian |
|----|--------------------------|---|
| 1 | Menu “Kamera” | Menu ini berfungsi untuk mengambil foto kain batik yang akan diuji. |
| 2 | Menu “ <i>Gallery</i> ” | Menu ini berfungsi untuk memilih gambar batik dari galeri <i>handphone</i> pengguna yang akan diuji. |
| 3 | Menu “ <i>About Us</i> ” | Menu ini berfungsi untuk menampilkan penjelasan singkat mengenai aplikasi yang mencakup informasi pembuat aplikasi dan tujuan aplikasi. |
| 4 | Tombol berlogo kamera | Tombol ini berfungsi untuk mengambil foto kain batik yang akan diuji. |
| 5 | Halaman informasi batik | Halaman ini akan ditampilkan setelah foto kain batik berhasil diklasifikasi. Fungsi dari halaman ini adalah untuk menampilkan informasi singkat mengenai kain batik yang diuji. |
| 6 | BatiQu <i>Box</i> | Kotak ini berbahan kayu dengan ukuran 12 cm x 12 cm x 17 cm dan memiliki tiga bagian. Kotak ini berfungsi untuk menjaga jarak dan kondisi pencahayaan tetap stabil dalam pengambilan foto kain batik. |

| | | |
|---|------------|---|
| 7 | Kain batik | Pengujian kain batik akan menggunakan enam sampel batik yaitu batik ceplok, batik parang, batik kawung, batik nitik, batik megamendung, dan batik tambal. Dikarenakan terbatasnya kain batik yang dijumpai, maka pengujian dibantu dengan menggunakan motif batik yang dicetak pada kertas. |
|---|------------|---|

Tabel 2.1 menjelaskan mengenai spesifikasi produk yang akan dirancang pada penelitian ini dengan beberapa hal yang diperhatikan. Seperti contohnya untuk hal yang diperhatikan yaitu kain batik, *box* dan beberapa fitur yang ada dalam aplikasi.

2.2 Verifikasi

Dengan adanya verifikasi produk dalam spesifikasi, dapat meningkatkan kinerja serta efektivitas yang mampu diukur agar mengetahui apakah sistem dapat berjalan dengan baik. Berikut merupakan verifikasi apa saja dalam sistem yang akan dirancang.

2.2.1 Verifikasi Spesifikasi 1

Tabel 2.2 Verifikasi spesifikasi pertama

| Hal | Akurasi dari Sistem Klasifikasi |
|--------------------|---|
| Rincian | Sistem dapat mengklasifikasikan jenis batik dengan tingkat akurasi dalam rentang 90% dari data uji yang dimasukkan. |
| Metode Pengukuran | Pengujian dilakukan dengan menjalankan sistem klasifikasi. |
| Prosedur Pengujian | Sistem klasifikasi dijalankan dengan melakukan pengujian pada data uji yang telah disediakan dari masing-masing jenis batik, lalu diamati tingkat akurasinya pada setiap pengujian. |

Tabel 2.2 menjelaskan mengenai verifikasi dengan rincian, metode pengukuran serta prosedur pengujian. Verifikasi ini digunakan untuk mengetahui tingkat akurasi sistem klasifikasi yang akan dirancang pada penelitian ini.

2.2.2 Verifikasi spesifikasi 2

Tabel 2.3 Verifikasi spesifikasi kedua

| Hal | Data Loss dari Sistem Klasifikasi |
|--------------------|---|
| Rincian | Sistem dapat mengklasifikasikan jenis batik dengan data <i>loss</i> mendekati $< 0,1$. |
| Metode Pengujian | Pengujian dilakukan dengan menjalankan sistem klasifikasi. |
| Prosedur Pengujian | Sistem klasifikasi dijalankan dengan melakukan pengujian pada data uji yang telah disediakan dari masing-masing jenis batik, lalu diamati nilai data <i>loss</i> pada setiap pengujian. |

Tabel 2.3 menunjukkan verifikasi kedua dalam spesifikasi sistem. Meskipun terlihat mirip dengan verifikasi pertama, fokus pada tahapan pengujian ini adalah data *loss* yang akan dirancang dalam sistem klasifikasi.

2.3 Kesimpulan dan Ringkasan CD-2

Produk dari klasifikasi batik ini akan dikembangkan dengan pembuatan aplikasi *mobile* bernama BatiQu. Aplikasi BatiQu dirancang dengan fitur yang ramah dan mudah dipahami oleh penggunanya agar dapat mengidentifikasi keenam jenis batik. Selain itu, pada aplikasi ini dapat memberikan edukasi dengan memberikan halaman penjelasan singkat mengenai asal usul, ciri khas dari pola batik dan penggunaan batik tersebut dalam kehidupan sehari-hari. Inovasi ini juga akan diimplementasikan bersama perangkat keras yang dapat membantu proses klasifikasi batik agar menjaga pencahayaan dan posisi pengambilan sampel. Perangkat keras berbahan kayu ini diberi nama BatiQu *Box* dengan panjang 12 cm, lebar 12 cm, dan tinggi 17 cm. Spesifikasi produk ini akan diverifikasi kembali kemampuan serta keefektifannya untuk mengetahui apakah sistem berjalan dengan

baik. Hal pertama adalah sistem dapat dimaksimalkan akurasi hingga 90% saat mengklasifikasikan jenis batik. Selanjutnya, sistem klasifikasi mampu mendapatkan data loss terendah mendekati $< 0,1$.

BAB 3

DESAIN RANCANGAN SOLUSI

3.1 Konsep Sistem

Metode perancangan sistem merupakan hal yang penting untuk diperhatikan. Dalam membangun suatu sistem terdapat berbagai unsur-unsur dan komponen yang berbeda. Perbedaan ini bahkan sangat beragam mulai dari fungsi sistemnya dan operasi untuk menjalankan sistem tersebut. Hal ini membantu dalam menentukan sistem terbaik yang akan diimplementasikan ke dalam rancangan penelitian. Untuk mengklasifikasikan batik ini maka dibutuhkan beberapa sistem untuk memilih sistem yang efektif agar *output* yang dihasilkan tercapai sesuai tujuan dari penelitian ini.

3.1.1 Pilihan Sistem

3.1.1.1 Produk A

a. Fitur Utama

Solusi system yang akan dirancang adalah aplikasi identifikasi jenis batik dengan menggunakan teknologi AI (*Artificial Intelligence*) dan perlu akses internet. Aplikasi dengan penggunaan teknologi AI ini memiliki beberapa kelebihan seperti hasil identifikasi yang akurat dan baik, cara kerja lebih cepat dan hasil identifikasi yang dapat di dokumentasikan sehingga rancangan aplikasi seperti ini sangat umum digunakan [12]. Rancangan aplikasi ini menggunakan kamera handphone untuk pengambilan gambar dan menggunakan teknologi AI untuk klasifikasi jenis batik.

b. Fitur Dasar

1. Fitur kamera yang dilengkapi dengan teknologi AI (*Artificial Intelligence*).
2. Halaman Informasi yang tertera jelas tingkat akurasi nya dan dapat menunjukkan tiga jenis motif batik termasuk tiga jenis pembuatan batik.

c. Fitur Tambahan

1. Terdapat *E-Commerce* yang khusus menjual kain batik
2. Terdapat artikel – artikel khusus tentang kain batik dan beberapa jenis batik.
3. Terdapat halaman akun.

d. Sifat Solusi yang Diharapkan

1. Mudah untuk identifikasi jenis batik apa saja karna sudah terbantu dengan teknologi AI (*Artificial Intelligence*).
2. Mudah untuk di akses karena hanya cukup install dan bisa langsung identifikasi jenis batik
3. Hasil identifikasi yang di paparkan cukup jelas karena berbentuk persen.
4. Dipermudah dengan ada nya fitur tambahan yang dapat langsung terhubung dengan *E-Commerce* penjual kain batik.

3.1.1.2 Produk B

a. Fitur Utama

Solusi system kedua yang akan dirancang adalah aplikasi identifikasi jenis batik dengan menggunakan teknologi *Deep Learning* dan bisa diakses tanpa perlu menggunakan internet. Rancangan aplikasi ini menggunakan kamera dan galeri sebagai pengambilan gambar kain batik dan menggunakan *Deep Learning* sebagai metode klasifikasi jenis batik. *Deep Learning* yang digunakan adalah metode yang telah dilatih menggunakan metode yang cocok pada aplikasi [13].

b. Fitur Dasar :

1. Fitur kamera yang dilengkapi dengan metode *Deep Learning*.
2. Fitur galeri yang dilengkapi dengan metode *Deep Learning*.
3. Halaman Informasi yang dilengkapi dengan pengetahuan singkat dari jenis batik yang teridentifikasi.

c. Fitur Tambahan :

Halaman *About Us* yang berisi definisi aplikasi, fungsi serta data diri dari pembuat aplikasi.

d. Sifat Solusi yang Diharapkan

1. Mudah untuk diakses karna cukup install dan bisa langsung identifikasi jenis batik.
2. Tidak membutuhkan perawatan yang intensif dan mengefisiensikan waktu dalam pengambilan gambar.

3.1.2 Analisis

3.1.2.1 Kriteria Analisis

Dari beberapa solusi sistem yang ada, hanya akan dipilih satu sistem untuk dikembangkan berdasarkan kriteria yang telah ditentukan. Kriteria ini mencakup beberapa parameter, di antaranya:

a. **Ekonomi**

Dari segi ekonomi, sistem yang akan dipilih adalah sistem yang memerlukan biaya produksi dan perawatan yang cukup murah sehingga dapat dengan mudah dijangkau serta digunakan oleh masyarakat umum. Biaya produksi dan perawatan yang murah juga akan memudahkan realisasi dari sistem yang akan dibuat.

b. **Fleksibilitas**

Pada kriteria ini, sistem yang akan dipilih adalah yang mudah dimodifikasi, sehingga dengan cepat diperbarui atau diperbaiki apabila ada kerusakan atau kekurangan.

c. **Ketersediaan komponen**

Sistem yang akan dipilih adalah sistem yang komponennya mudah dicari atau jika tidak menggunakan komponen itu lebih baik lagi. Ketersediaan komponen ini juga dapat mempengaruhi harga komponen yang nantinya berdampak pada biaya produksi.

d. **Penggunaan/Pengguna**

Sistem yang akan dipilih berdasarkan kriteria penggunaan adalah sistem yang mudah untuk digunakan. Kemudahan dalam penggunaan sistem ini dapat dilihat dari tahapan penggunaan alat/sistem yang sedikit, serta tampilan yang sederhana dan mudah dimengerti.

e. **Performa**

Sistem yang akan dipilih adalah sistem dengan performa terbaik. Performa ini dilihat dari kemampuan sistem dalam memproses program klasifikasi batik pada aplikasi. Berdasarkan jenis aplikasinya, performa sistem dapat dipengaruhi oleh

faktor yang berbeda-beda. Salah satu contohnya adalah performa pada aplikasi yang bersifat *online* dapat dipengaruhi oleh kecepatan dan kestabilan koneksi internet ke *server*, sedangkan performa pada aplikasi yang bersifat *offline* dipengaruhi oleh performa mesin dari *handphone* yang digunakan.

3.1.2.2 Analisis Konsep

Dari konsep yang telah diusulkan didapat kelebihan dan kekurangan dari masing-masing sistem seperti:

Tabel 3.1 (a) Kriteria Ekonomi (b) Kriteria Sistem (c) Kriteria ketersediaan komponen (d) Kriteria Penggunaan/Pengguna (e) Kriteria Performa

| Kriteria Ekonomi | |
|---------------------------|--|
| 1. Solusi sistem Produk A | Aplikasi ini membutuhkan biaya produksi dan perawatan cukup mahal dikarenakan cara kerja aplikasi ini sangat membutuhkan <i>server</i> dan internet yang baik. Perawatan dari aplikasi ini juga membutuhkan biaya tambahan untuk <i>maintenance</i> agar aplikasi tetap berjalan dengan baik. |
| 2. Solusi sistem Produk B | Aplikasi ini membutuhkan biaya produksi dan perawatan yang cukup murah karena cara kerja dari aplikasi ini tidak perlu akses internet. Perawatan dari aplikasi ini pun tidak membutuhkan biaya tambahan karena untuk <i>maintenance</i> dari aplikasi ini cukup dari metode <i>Deep Learning</i> yang dilatih ulang atau terdapat <i>update data</i> . |

(a)

| Kriteria Fleksibilitas | |
|---------------------------|---|
| 1. Solusi Sistem Produk A | Sistem yang akan dipilih adalah sistem mudah dimodifikasi, sehingga sistem dapat dengan cepat diperbarui atau diperbaiki apabila ada kerusakan atau kekurangan. |
| 2. Solusi Sistem Produk B | |

(b)

| Kriteria Ketersediaan Komponen | |
|--------------------------------|--|
| 1. Solusi sistem Produk A | Aplikasi A membutuhkan <i>server</i> terbaik namun juga memiliki harga terjangkau. Dari kebutuhan ini dapat mempengaruhi biaya produksi. |
| 2. Solusi sistem Produk B | Aplikasi B tidak memerlukan komponen tambahan selain <i>handphone</i> dari masing-masing pengguna. |

(c)

| Kriteria Penggunaan/Pengguna | |
|-----------------------------------|--|
| 1. Solusi Sistem pertama Produk A | Kedua produk ini mudah untuk digunakan. Kemudahan dalam penggunaan kedua produk ini dapat dilihat dari tahapan penggunaan alat/sistem yang sedikit, serta tampilan sistem yang sederhana dan mudah dimengerti. |
| 2. Solusi Sistem kedua Produk B | |

(d)

| Kriteria Performa | |
|---------------------------|--|
| 1. Solusi sistem Produk A | Performa aplikasi ini sangat tergantung dari koneksi internet yang stabil dan ketersediaan dari <i>server</i> yang ada di dalamnya. |
| 2. Solusi sistem Produk B | Performa aplikasi ini tidak bergantung pada koneksi internet dan <i>server</i> yang ada di dalamnya. Performa dari aplikasi ini dilihat dari akurasi metode <i>Deep Learning</i> yang telah dilatih. |

(e)

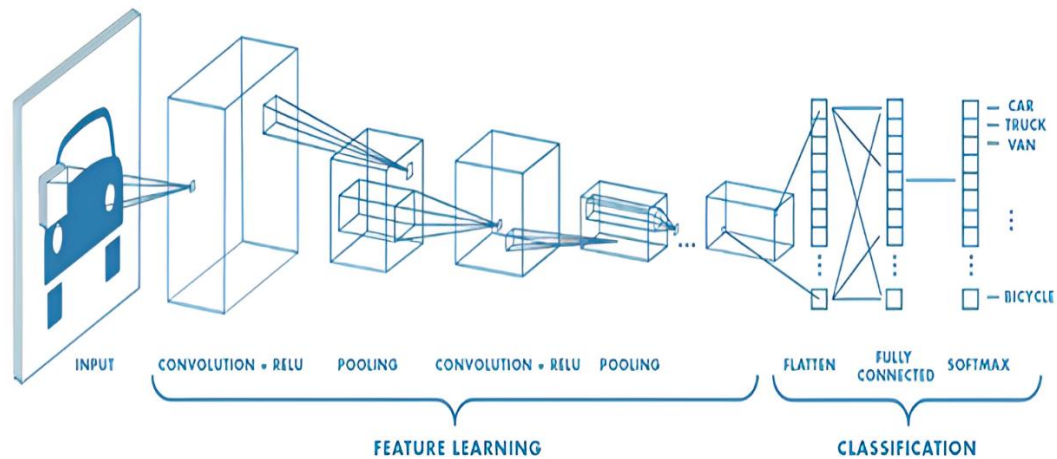
Pada tabel 3.1 berisikan konsep kriteria yang akan dibangun pada perancangan penelitian. Setiap konsep kriteria sudah dipilih berdasarkan kegunaan dari perancangan aplikasi. Dengan adanya kriteria ini, maka mempermudah dalam melakukan konsep perancangan.

3.1.3 Sistem yang akan Dikembangkan

Berdasarkan dari pilihan sistem yang sudah dipaparkan di atas, penelitian ini akan mengembangkan produk B karena memiliki beberapa keunggulan dibandingkan dengan produk A. Contohnya pada produk B karena berbasis *offline* jadi tidak memerlukan *server* untuk menyimpan dataset maupun program dari aplikasinya. Pembuatan dalam aplikasi *offline* juga mempermudah dalam penggunaannya dikarenakan aplikasi dapat digunakan kapan pun dan dimanapun. Perawatan dari aplikasi *offline* yang dirancang juga cukup mudah untuk dilakukan. Pada solusi sistem produk A kurang optimal karena memiliki kekurangan pada aplikasi yang berbasis *online*. Pada produk B tampilan aplikasi memiliki tampilan yang sederhana dan mudah dimengerti sehingga pengguna atau masyarakat lebih mudah untuk menggunakan. Performa dari sistem produk B tidak bergantung pada koneksi internet, namun dari akurasi metode *machine learning* yang telah dilatih. Sehingga, secara keseluruhan perbandingan yang telah dipaparkan memperkuat alasan untuk menggunakan solusi sistem produk B dengan *Deep Learning* yang sudah dilatih.

3.2 Rencana Desain Sistem

Solusi sistem berbasis *android* ini secara garis besar dibagi menjadi dua skenario yaitu skenario pada kode program klasifikasi menggunakan CNN dan skenario pada aplikasi BatiQu. Skenario pertama yaitu pada kode program klasifikasi dengan menggunakan CNN, sebelum memasuki lebih dalam mengenai skenario dan tahap menyiapkan dataset alangkah baiknya mengetahui terlebih dahulu mengenai struktur lapisan pada CNN itu sendiri. Metode *Convolutional Neural Network* (CNN) memiliki beberapa tahapan pembangunan yang dijelaskan pada gambar 3.1:



Gambar 3.1 lapisan Metode *Convolutional Neural Network* [14]

a. *Convolutional layer*:

Pada tahapan *layer* ini masuk ke dalam proses utama yang mendasari metode CNN. *Layer* yang digunakan untuk mengekstraksi fitur/kernel dari *input* citra. *Layer* ini terdiri dari *filter-filter* yang melakukan operasi konvolusi dengan tujuan sebagai ekstraksi fitur sehingga dapat mempelajari representasi *input layer*.

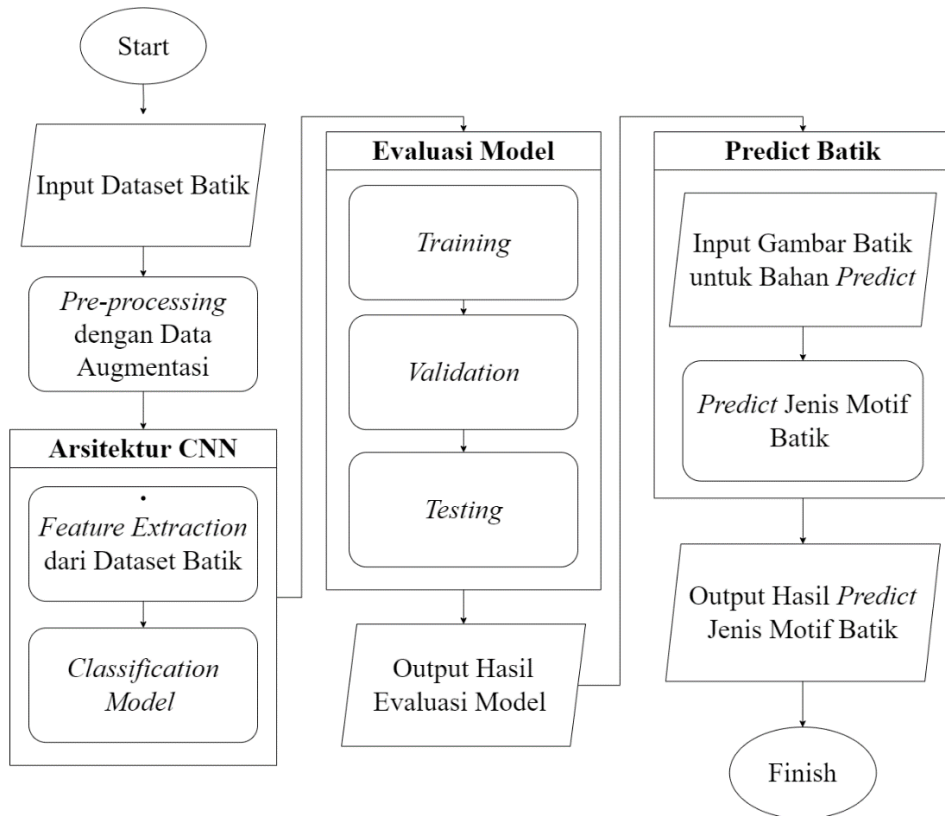
b. *Pooling layer*:

Tahapan berikutnya, merupakan lapisan yang berfungsi untuk mengurangi ukuran spasial fitur melalui pengurangan dimensi dari *feature maps* (*down sampling*). Adanya tahapan ini dapat mempercepat komputer untuk melakukan proses pelatihan model menjadi lebih efektif.

c. *Fully connected layer*:

Tahapan berikut ini merupakan lapisan yang berfungsi sebagai transformasi dimensi citra/gambar agar citra dapat diklasifikasikan secara linear. Lapisan ini menggunakan perkalian matriks untuk mendapatkan hasil keluaran. Sehingga, hasil dapat melakukan klasifikasi.

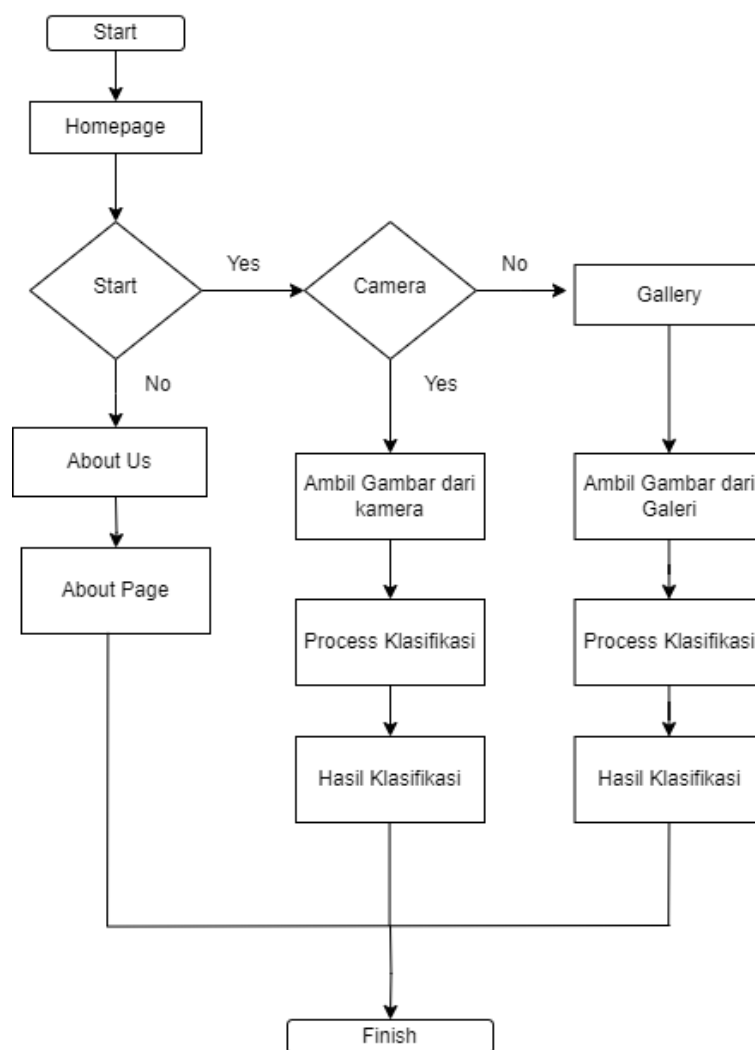
Setelah mengetahui penjelasan dari tahapan pembangunan CNN. Langkah yang dilakukan untuk merancang pada penilitan ini adalah dengan melakukan penyiapan dataset citra gambar batik yang didapat melalui *Kaggle* berupa citra batik yang terdiri dari enam kelas, yaitu ceplok, parang, nitik, megamendung, kawung, dan tambal. Seluruh dataset yang terkumpul akan dibagi menjadi tiga kategori data berdasarkan fungsinya yaitu data *training*, *validation*, dan *testing*. Setelah itu, dilakukan *pre-processing* berupa data augmentasi (meliputi *resize*, *brightness*, *shear*, *width*, *rotate*) yang dapat meningkatkan variasi dari jumlah data yang ada. Data citra selanjutnya akan diolah dengan arsitektur CNN yang digunakan. Pada tahap ini, citra akan diekstraksi fiturnya lalu fitur-fitur yang didapatkan akan digunakan untuk proses klasifikasi. Tahap selanjutnya adalah evaluasi model yang terdiri dari tiga proses yaitu *training*, *validation*, dan *testing*. Pada awalnya, akan dilakukan proses *training* yang berfungsi untuk melatih citra dengan data *training* yang sebelumnya telah disiapkan. Setelah itu data akan melalui tahap *validation* yang berfungsi untuk menguji kemampuan dari model klasifikasi yang telah dibuat dengan data *validation* yang telah disiapkan. Tahap evaluasi model kemudian ditutup dengan proses *testing* yang berfungsi sebagai simulasi penggunaan model klasifikasi dengan data yang belum pernah dilihat sebelumnya oleh model tersebut. Hasil evaluasi model klasifikasi bisa berbentuk grafik akurasi dan *loss* dari proses *training* dan *validation*, *confusion matrix*, dan *classification report* yang berisi nilai *precision*, *recall*, dan *f1-score*. Ketika hasil evaluasi model sudah tergolong baik, proses bisa dilanjutkan ke tahap melakukan identifikasi atau *predict* gambar batik. Tahap *predict* ini dilakukan dengan menambahkan citra bahan uji secara manual untuk selanjutnya diidentifikasi oleh model sesuai dengan motif yang dideteksi. Keseluruh alur kerja model klasifikasi batik yang sudah dijelaskan ini dapat dilihat melalui *flowchart* pada gambar di bawah ini.



Gambar 3.2 *Flowchart* Program CNN

Pada gambar 3.2 menjelaskan mengenai *flowchart* dari program CNN, ketika proses semua berjalan dan mendapatkan performa terbaik maka keluarlah *output* yang akan dilanjutkan pada aplikasi yang sudah dirancang. Setelah, program CNN sudah bisa berjalan tahap awal pada aplikasi adalah membuka laman *homepage* ada dua tombol yang bisa dipilih yaitu tombol mulai dan tombol *about us*. Kedua tombol tersebut bisa dipilih sesuai keinginan pengguna. Pengguna bisa memilih tombol mulai untuk memulai uji klasifikasi batik dan akan diarahkan pada halaman uji batik. Terdapat dua pilihan untuk mengambil gambar sebagai uji jenis batik yaitu dari kamera dan dari *gallery*. Jika pengguna memilih tombol kamera, tahap selanjutnya adalah pengambilan gambar menggunakan kamera *handphone* lalu setelah itu akan diproses klasifikasi menggunakan *Deep Learning*. Selanjutnya, akan menampilkan hasil dari klasifikasi berupa nama batik dan informasi singkat dari batik yang diuji pada halaman informasi.

Jika pengguna memilih tombol *gallery*, pengguna akan diarahkan pada halaman *gallery* yang mana pengguna bisa memilih foto batik yang telah dimiliki. Pada bagian ini batik akan diolah atau diproses menggunakan proses klasifikasi dengan metode Deep Learning. Selanjutnya, dari hasil proses sebelumnya akan menghasilkan tampilan hasil dari klasifikasi berupa nama batik dan informasi singkat dari batik yang diuji. Ketika pengguna memilih tombol *about us*, akan diarahkan pada halaman tentang yang berisi informasi singkat dari pembuat aplikasi ini seperti dijelaskan pada gambar 3.4:



Gambar 3.3 Flowchart Aplikasi BatiQu

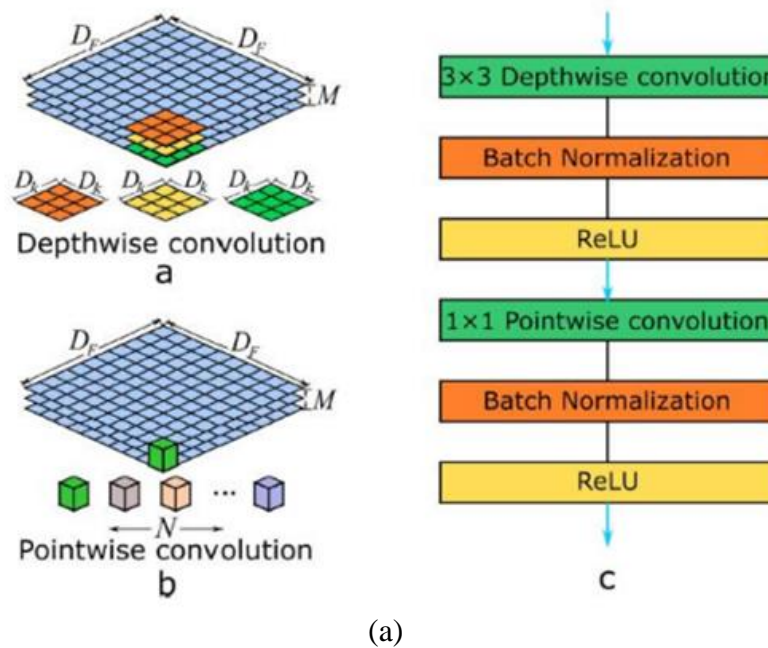
3.3 Pengujian Komponen (Kalibrasi)

Pengujian sistem akan dilakukan dengan melihat arsitektur yang menghasilkan persentase terbaik dan mampu meningkatkan performa sistem dalam mengklasifikasikan jenis batik. Pengujian ini akan menggunakan arsitektur *Convolutional Neural Network* yang merupakan salah satu metode dari Deep Learning dan dirancang untuk mengolah sebuah data dari bentuk citra dua dimensi [21]. Dari beberapa penelitian terdahulu, klasifikasi menggunakan CNN telah mampu mencapai nilai akurasi lebih tinggi dengan berbagai metode. Maka dengan itu, penelitian ini akan menggunakan metode CNN dengan metode MobileNet V1, ResNet152 V2, ResNet101 V2 dan ResNet50 V2.

Sebelum masuk lebih dalam mengenai tujuan dalam penelitian, maka alangkah baiknya mengetahui terlebih dahulu mengenai metode MobileNet dan juga ResNet. Seperti yang sudah dijelaskan di atas, bahwa metode MobileNet masuk ke dalam arsitektur *Convolutional Neural Network* (CNN). Dimana CNN merupakan salah satu jenis arsitektur yang biasa digunakan dalam klasifikasi dan pengenalan objek/citra. Tujuan metode CNN adalah melatih metode jaringan syaraf tiruan dimana hasil yang diharapkan dapat meminimalkan data *loss* pada hasil prediksi dengan data asli. Salah satu metode yang ada dalam *Convolutional Neural Network* (CNN) adalah MobileNet. MobileNet merupakan sebuah metode arsitektur yang dirancang dengan *streamlined architecture* menggunakan konsep *depthwise separable convolutions* yang dipisahkan secara mendalam dengan tujuan agar membangun *convolutional neural networks* yang ringan dan mudah dalam mengimplementasikan kebutuhan *mobile* dan *embedded vision* [15]. MobileNet dibuat berdasarkan *Depthwise Separable Convolutions* untuk mengurangi komputasi di *layer* awal [16].

Konsep *depthwise separable convolutions* sendiri ditujukan untuk dapat mengurangi jumlah kompleksitas dari parameter dan komputasi yang digunakan saat pelatihan model. Dalam konsep ini terdapat beberapa pembagian konvolusi dengan dua tahapan, yaitu *Depthwise Convolution* dan *Pointwise Convolution*. *Depthwise Convolution* sendiri dalam perancangan, diaplikasikan pada kernel atau filter dalam jumlah kecil yang masing-masing input dibuat secara terpisah. *Convolution Pointwise* merupakan kebalikan dari *Convolution Depthwise* dimana

perancangan diaplikasikan pada kernel atau filter dalam jumlah kecil yang masing-masing *output* dari *Convolution Depthwise* digabungkan sehingga dapat menghasilkan *output* terakhir. Pengaplikasian MobileNet pada metode *Convolutional Neural Network* untuk klasifikasi motif batik secara *real-time* dengan melakukan beberapa variasi model pada citra menjadi usulan dalam pemilihan konsep sistem. Pada gambar 3. 4 merupakan metode MobileNet dengan konvolusi dan metode parameter sebagai berikut :

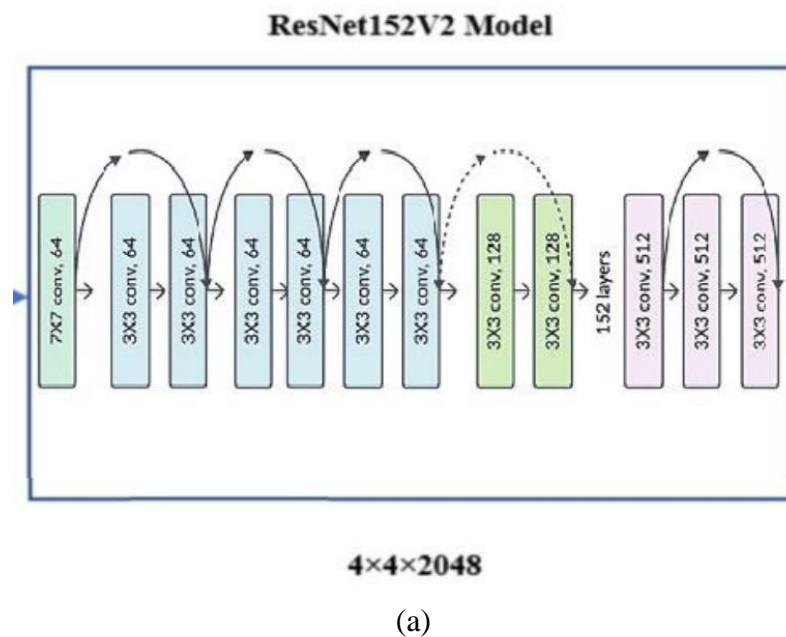


| Layer (type) | Output Shape | Param # |
|---|--------------------|---------|
| mobilenet_1.00_224 (Functional) | (None, 7, 7, 1024) | 3228864 |
| global_average_pooling2d (GlobalAveragePooling2D) | (None, 1024) | 0 |
| flatten (Flatten) | (None, 1024) | 0 |
| batch_normalization (Batch Normalization) | (None, 1024) | 4096 |
| dense (Dense) | (None, 256) | 262400 |
| dropout (Dropout) | (None, 256) | 0 |
| dense_1 (Dense) | (None, 128) | 32896 |
| dropout_1 (Dropout) | (None, 128) | 0 |
| classification (Dense) | (None, 6) | 774 |

Gambar 3.4 (a) Konvolusi arsitektur MobileNet (b) Arsitektur MobileNet V1 dengan model parameter

Arsitektur yang kedua dalam pengujian komponen terdapat ResNet152 V2 sebagai pembanding untuk arsitektur MobileNet. *Residual Network* sendiri atau biasa disebut dengan ResNet merupakan salah satu jenis arsitektur yang cukup populer dalam metode CNN. Adapun arsitektur ini dibuat oleh Kaiming He et al [17]. Residual Neural Network merupakan arsitektur CNN dengan variasi *layer* yang beragam diantaranya layer 18, 34, 50, 101 dan 152. Karakteristik arsitektur pada ResNet adalah untuk menghindari terjadi hilangnya masalah pada *gradient*. Arsitektur ResNet melakukan pemetaan identitas inputan data citra dengan cara melewati beberapa *layer* pada saat komputasi dengan mengurangi fitur yang dipelajari dari *layer* yang dimasukkan. Prinsip kerja pada ResNet yaitu membangun jaringan *layer* lebih dalam daripada lapisan jaringan yang biasanya dengan menemukan jumlah lapisan yang optimal.

Pada penelitian ini menggunakan ResNet152 V2 dengan memiliki *layer* yang hampir sama dengan *layer* lainnya seperti contohnya [18]. ResNet152 V2 masuk kedalam layer kelima dari arsitektur ResNet yang memiliki 152 lapisan. Tahap pertama dari arsitektur ResNet yaitu konvolusi 7 x 7 dengan filter berjumlah 64 dan 2 stride. Tahap kedua yang dilakukan adalah *max pooling* dengan ukuran matriks 3 x 3 dan 2 stride. [19] Tahapan ini dilakukan secara berulang seperti yang dijelaskan pada gambar 3.5 sebagai berikut:



| Layer (type) | Output Shape | Param # |
|---|----------------------|----------|
| resnet152v2 (Functional) | (None, 10, 10, 2048) | 58331648 |
| global_average_pooling2d (GlobalAveragePooling2D) | (None, 2048) | 0 |
| flatten (Flatten) | (None, 2048) | 0 |
| dense (Dense) | (None, 128) | 262272 |
| classification (Dense) | (None, 6) | 774 |

(b)

Gambar 3.5 (a) Arsitektur ResNet125 (b) Model parameter

Tujuan klasifikasi jenis batik menggunakan CNN yaitu pengujian dapat menghasilkan tingkat akurasi dalam rentang 90% dari data uji yang telah dimasukkan. Data uji dari masing-masing jenis batik akan diproses menggunakan tahapan *pre-processing* pada data *training* dan data *testing*. Untuk metode pengujian pada tahapan *pre-processing* dilakukan dengan membandingkan hasil akurasi dan data loss dengan teknik data augmentasi yang berbeda-beda seperti *resize*, *crop*, *rotate*, *flip*. Selanjutnya adalah menguji sistem klasifikasi jenis batik dengan nilai data *loss* seminimal mungkin. Untuk penelitian yang akan dirancang pada sistem ini, data *loss* dari seluruh pengujian mendekati 0,1. Hal ini dikarenakan nilai data *loss* yang semakin kecil akan menunjukkan bahwa sistem melakukan klasifikasi dengan kesalahan yang minim. Selain itu, pengujian juga akan dilakukan dengan cara memperhatikan *evaluate performance*. Dimana performanya dapat dilihat dari *epoch* dan *learning rate*. *Epoch* merupakan *hyperparameter* yang dirancang untuk mendefinisikan berapa banyak algoritma pembelajaran yang akan bekerja melalui dataset pelatihan. Dari pelatihan pembelajaran tersebut dapat menunjukkan besar kecilnya perubahan dalam setiap pengoptimalan pembelajaran dataset. *Learning rate* sendiri merupakan indikator dalam proses *training* dataset yang perlu diperhatikan dan bertujuan untuk menghitung waktu proses pembelajaran. Semakin tinggi hasil nilai dari *learning rate* maka dapat diartikan bahwa waktu yang dibutuhkan akan semakin cepat dalam proses pembelajaran

dataset, namun hasil dari akurasi akan semakin rendah. Akurasi dapat mempengaruhi hasil akhir prediksi suatu sistem yang akan dirancang.

Pengujian selanjutnya adalah aplikasi BatiQu sebagai alat uji batik dalam bentuk perangkat lunak. Pengujian aplikasi ini dilakukan untuk melihat apakah akurasi yang dibentuk di *Deep Learning* selaras dengan yang ada di aplikasi. Aplikasi BatiQu akan mendeteksi pola dasar yang ada pada kain batik seperti titik, garis, dan diagonal sesuai dengan model *Deep Learning* yang telah dilatih [20]. Hasil uji dari aplikasi BatiQu dapat dilihat dari kecocokan gambar kain batik dengan halaman informasi yang telah dibuat. Komponen lainnya yang akan diuji adalah perangkat keras BatiQu *Box* yang berfungsi sebagai tempat untuk bahan uji kain batik dan *handphone*. *Box* ini dibuat dengan ukuran 12 cm, lebar 12 cm, dan tinggi 17 cm untuk memberi jarak antara kamera *handphone* dengan kain batik. Alasan dibuatnya *box* adalah menyelaraskan sudut dan pencahayaan saat proses klasifikasi jenis batik.

3.4 Jadwal Pengerjaan

Tabel 3.2 Jadwal Pengerjaan *Capstone Design*

| Kegiatan | Progress | 2022 | | | | 2023 | | | | | | |
|--------------------------------------|----------|-----------|---------|----------|----------|---------|----------|-------|-------|-----|------|------|
| | | September | Oktober | November | Desember | Januari | Februari | Maret | April | Mei | Juni | Juli |
| Fase1 | | | | | | | | | | | | |
| Studi Literature | 100% | | | | | | | | | | | |
| Pemilihan Arsitektur | 100% | | | | | | | | | | | |
| Pengambilan Data | 80% | | | | | | | | | | | |
| Perancangan Spesifikasi & Verifikasi | 100% | | | | | | | | | | | |
| Fase2 | | | | | | | | | | | | |
| Perancangan Program Sistem | 0% | | | | | | | | | | | |
| Uji Coba Sistem Klasifikasi | 0% | | | | | | | | | | | |
| Pembuatan Aplikasi | 0% | | | | | | | | | | | |
| Uji Coba Aplikasi | 0% | | | | | | | | | | | |
| Analisa dan Penulisan | 0% | | | | | | | | | | | |
| Sidang Tugas Akhir | 0% | | | | | | | | | | | |

Dari tabel 3.2 tersebut terlihat bahwa sebelumnya fase 1 sudah selesai yang ditandai dengan warna merah dan ditutup dengan mengunggah dokumen proposal *Capstone Design*. Saat ini pengerjaan sudah masuk ke dalam fase 2 yang ditandai dengan warna kuning. Selanjutnya akan ditampilkan tabel hasil implementasi pengerjaan *Capstone Design* sebagai berikut:

Tabel 3.3 *Final Jadwal Pengerjaan Capstone Design*

| Kegiatan | Progress | 2022 | | 2023 | | | | | | | | |
|-----------------------------|----------|----------|----------|---------|----------|-------|-------|-----|------|------|---------|--|
| | | November | Desember | Januari | Februari | Maret | April | Mei | Juni | Juli | Agustus | |
| Fase 1 | | | | | | | | | | | | |
| Pengambilan Data | 100% | | | | | | | | | | | |
| Fase 2 | | | | | | | | | | | | |
| Perancangan Program Sistem | 100% | | | | | | | | | | | |
| Uji Coba Sistem Klasifikasi | 100% | | | | | | | | | | | |
| Pembuatan Aplikasi | 100% | | | | | | | | | | | |
| Uji Coba Aplikasi | 100% | | | | | | | | | | | |
| Analisa dan Penulisan | 100% | | | | | | | | | | | |
| Sidang Tugas Akhir | 100% | | | | | | | | | | | |

Pada tabel 3.2 terlihat progress pengerjaan dari *Capstone Design* yang tercatat hingga bulan Agustus 2023. Pada tabel 3.3 terlihat cukup banyak perbedaan dibandingkan dengan tabel 3.2 yang sebelumnya sudah ditampilkan. Pada bagian uji coba sistem klasifikasi yang awalnya direncanakan selesai bulan Februari harus ditambah hingga bulan Juli karena terdapat kesulitan dalam memperoleh hasil klasifikasi yang baik dari beberapa metode klasifikasi yang sudah dicoba. Selain itu sumber dataset dengan kualitas baik yang terbatas juga cukup mempersulit pengerjaan sehingga pengumpulan dataset baru bisa diselesaikan pada bulan Maret 2023. Penambahan waktu dari uji coba sistem klasifikasi ini berdampak pada penambahan waktu juga pada pembuatan aplikasi dan uji coba aplikasi.

3.5 Kesimpulan dan Ringkasan CD-3

Pengklasifikasian batik membutuhkan metode sistem terbaik untuk diimplementasikan agar *output* yang dihasilkan sesuai dengan tujuan dalam penelitian ini. Pilihan sistem pertama yaitu menggunakan produk A yang mana sistem ini menggunakan bantuan teknologi AI (*Artificial Intelligence*). Selanjutnya adalah produk B yang mana sistem ini menggunakan model *Deep Learning* yang telah dilatih. Kedua produk sistem ini akan dibandingkan dan dipilih satu sistem terbaik untuk dikembangkan. Beberapa parameter yang menjadi kriteria pengujian ini dimulai dari segi ekonomi, sistem yang mudah dimodifikasi, ketersediaan komponen, dan penggunaan yang lebih efisien untuk digunakan.

Dari kedua metode sistem yang telah dibandingkan, hasil sistem terbaik yang didapatkan adalah menggunakan produk B. Sistem ini dipilih karena memiliki keunggulan salah satunya adalah sistem ini tidak memerlukan biaya produksi yang besar, tidak memerlukan komponen tambahan, dan memiliki performa yang bergantung pada akurasi metode *Deep Learning*.

Dari sistem yang dipilih, selanjutnya adalah rancangan dari model *Deep Learning*. Rancangan *Deep Learning* yang digunakan adalah metode *Convolutional Neural Network* (CNN) dengan cara kerja seperti *pre-processing*, *training* dan *validation*. Selanjutnya terdapat rancangan *mobile application* yang memiliki dua skenario pengambilan gambar yaitu melewati kamera dan galeri.

Dari rancangan yang dibuat, metode *Convolutional Neural Network* (CNN) akan diuji dan dipilih sesuai dengan arsitektur yang memiliki presentase terbaik. Pengujian dilakukan pada empat arsitektur yaitu arsitektur ResNet152 V2, ResNet101 V2, ResNet50 V2 dan MobileNet V1. Perbedaan dari empat arsitektur tersebut adalah jumlah lapisan *layer* dimana untuk ResNet152 V2 memiliki hingga 152 layer dengan komputasi 6.6ms, ResNet101 V2 menggunakan layer 101 layer dengan komputasi 5.4ms, ResNet50 V2 berisikan layer sebanyak 50 dengan komputasi sebesar 4.4ms sedangkan untuk MobileNet V1 bisa dimodifikasi sesuai kebutuhan dengan komputasi lebih cepat sebesar 3.4ms. Pengujian selanjutnya terdapat pada aplikasi BatiQu beserta BatiQu Box. Pengujian pada aplikasi BatiQu dilakukan untuk dapat melihat apakah akurasi yang dibentuk di *Deep Learning* selaras dengan yang ada di aplikasi. Pengujian BatiQu *Box* dilakukan untuk menyelaraskan sudut dan pencahayaan saat proses klasifikasi jenis batik.

BAB 4

IMPLEMENTASI

4.1 Implementasi Sistem

Implementasi sistem merupakan tahapan atau prosedur yang dilakukan untuk dapat menyelesaikan perancangan sistem yang telah dibuat dan akan dilakukan pengujian. Dalam implementasi sistem juga disiapkan sistem yang akan dijalankan. Tujuan implementasi sistem ini agar peneliti dapat menyelesaikan desain sistem yang telah dirancang dalam dokumen sebelumnya serta mendokumentasikan prosedur yang diperlukan. Memastikan serta memperhitungkan bahwa sistem dapat beroperasi dengan baik serta memenuhi parameter yang sudah disediakan.

4.1.1 Deep Learning

Berikut beberapa cara kerja dari *Deep Learning* pada sistem yang dirancang:

4.1.1.1 Pengumpulan data

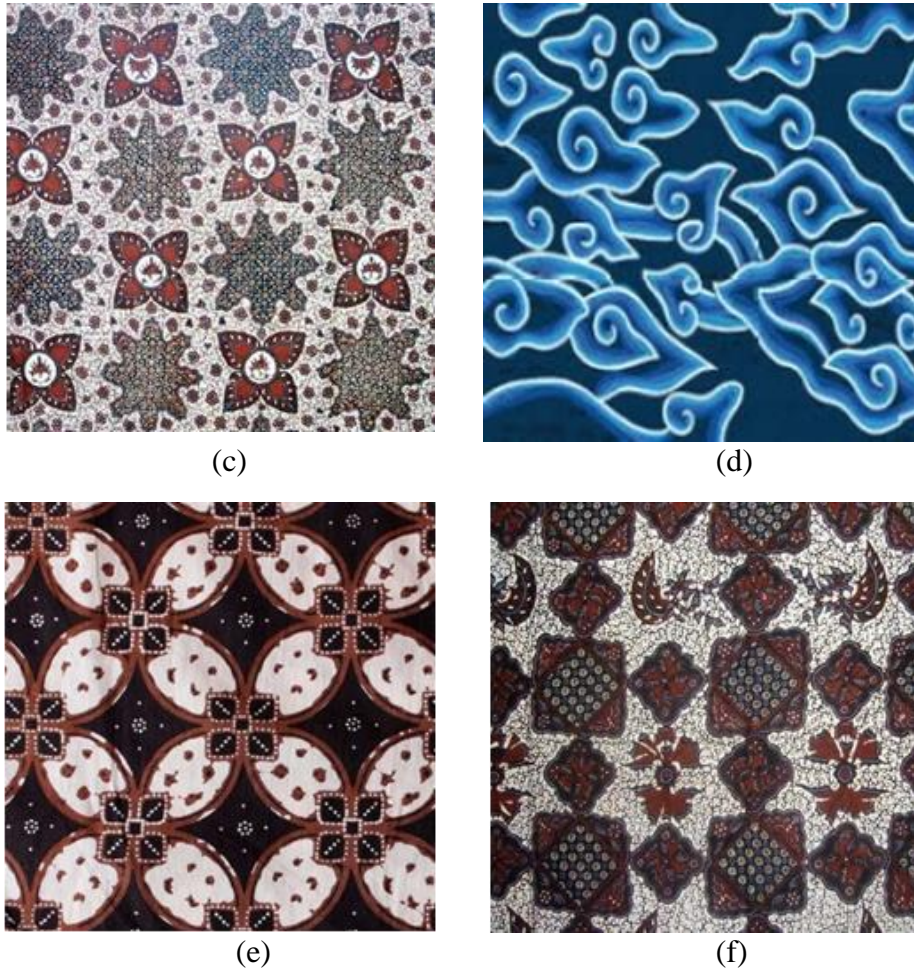
Data yang dikumpulkan adalah gambar batik dengan jumlah 660 gambar yang dikelompokkan menjadi 6 kelas yaitu batik kawung, batik ceplok, batik tambal, batik megamendung, batik parang dan batik nitik. Setelah data dibagi menjadi beberapa kelompok kemudian data dibagi lagi menjadi data *training*, data *testing* dan juga data *validasi*. Tujuan pengelompokkan data menjadi tiga folder untuk dilakukan pelabelan di setiap datanya. Pada data *training* berjumlah 480 gambar, data validasi berjumlah 120 data dan untuk data *testing* berjumlah 60 gambar.



(a)



(b)



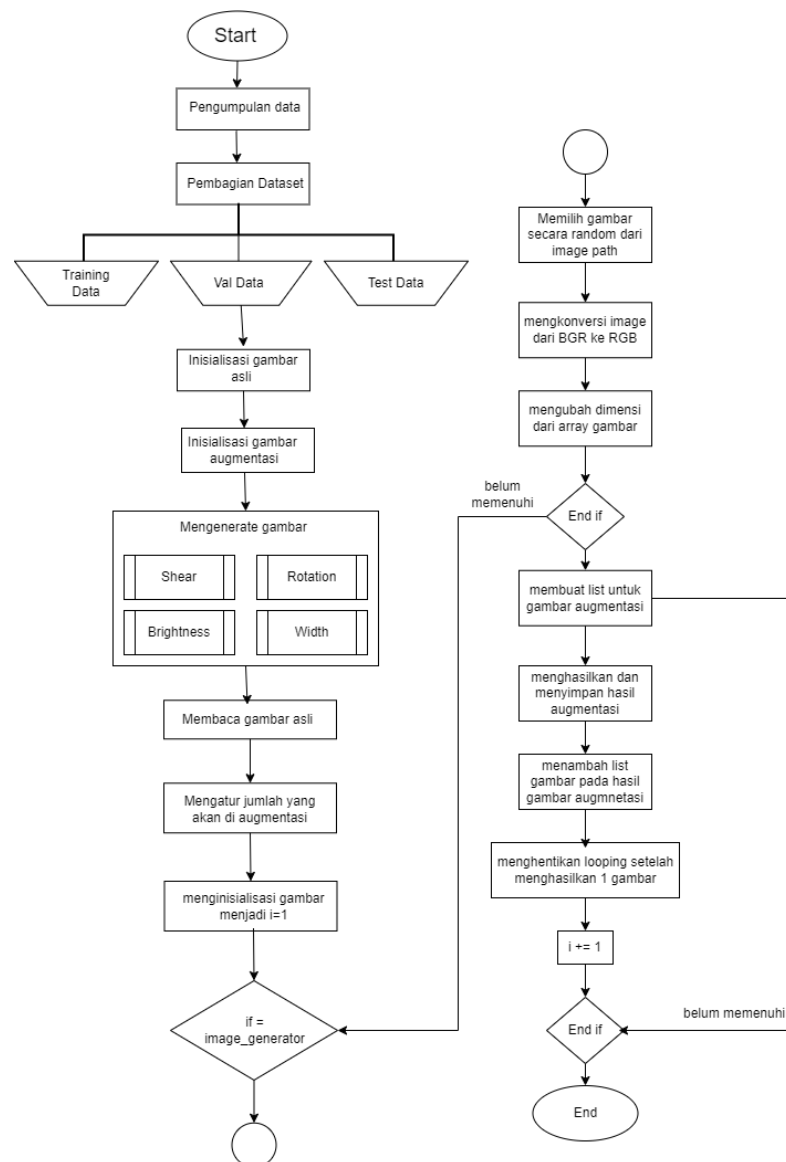
Gambar 4.1 (a)batik tambal (b)batik parang (c)batik nitik (d)batik megamendung (e)batik kawung (f)batik ceplok

Gambar 4.1 di atas adalah contoh gambar dari dataset batik yang berisi batik dari masing-masing kelas yaitu batik tambal, batik parang, batik nitik, batik megamendung, batik kawung, dan batik ceplok. Selanjutnya, data yang telah dikumpulkan akan di proses ke tahap selanjutnya yaitu tahap *pre-processing*.

4.1.1.2 Pre-processing

Setelah proses dari pengumpulan data, tahapan selanjutnya adalah adanya *pre-processing*. *Pre-processing* adalah tahapan awal pengolahan data gambar pada *Convolutional Neural Network* (CNN). Teknik *pre-processing* digunakan untuk mempersiapkan data *input* sebelum memasuki proses bagian *input* metode CNN. Tujuan dari *pre-processing* adalah untuk meningkatkan kualitas gambar maupun kinerja model dengan meminimalkan informasi yang tidak valid dan mempertahankan informasi penting yang dapat dilanjutkan ke dalam proses

selanjutnya. Augmentasi data merupakan salah satu metode untuk dapat mengurangi *overfitting* dengan meningkatkan ukuran citra dataset dan akurasi dengan tujuan meminimumkan data agar komputasi lebih cepat. Dengan augmentasi data juga dapat memodifikasi dan memperbanyak citra agar menciptakan variasi sampel data baru sehingga kualitas dan keragaman data lebih banyak sebelum dimasukkan ke dalam sistem. Di bawah ini akan ditampilkan dari proses atau *flowchart pre-processing*.



Gambar 4.2 Flowchart Pre-processing dengan Augmentasi data

Pada gambar 4.2 menjelaskan mengenai *flowchart* dengan augmentasi data, dimana sebelum melakukan *pre-processing* dataset terlebih dahulu dibagi menjadi tiga data bagian. Daataset yang telah terkumpul berjumlah 660 data. Kemudian,

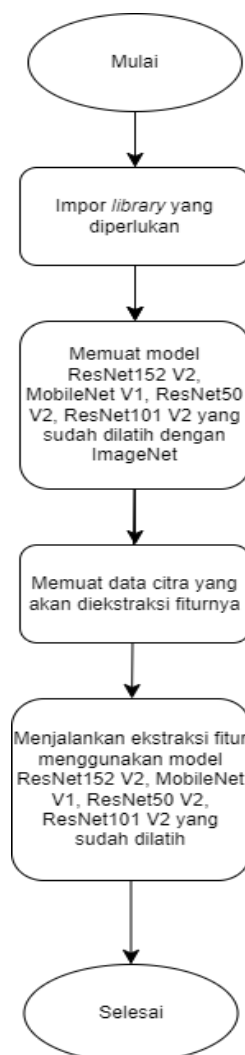
setelah dataset sudah terbagi masuklah kedalam *pre-processing* dengan mengaugmentasi data. Dari data asli dibaca kemudian diaugmentasi dengan jumlah yang diinginkan. Sehingga menghasilkan *output*, data yang pertama data *training* berisikan citra gambar dengan jumlah 400 gambar dalam setiap kelasnya dan untuk total keseluruhan citra gambar data *training* berjumlah 2400 citra. Data yang kedua yaitu data validasi berisikan citra gambar dengan data gambar yang berbeda berjumlah 100 gambar dalam setiap kelasnya sehingga total dalam data validasi berjumlah 600 citra. Terakhir, untuk data *testing* berisikan 50 citra gambar tiap kelasnya dan untuk total citra gambar pada data *testing* berjumlah 300 citra gambar teraugmentasi. Total citra gambar keseluruhan berjumlah 3.300 gambar yang sudah teraugmentasi. Setelah dari proses *pre-processing* masuklah ke dalam tahap selanjutnya.

Tahapan setelah adanya *pre-processing* terdapat proses *preparing* data. Dalam *preparing* data masih mengenerator data citra gambar menggunakan *rescale*, *featurewise std normalization*, *horizontal*, dan *vertical flip*. *Rescale* merupakan parameter untuk mengubah intensitas piksel dalam gambar. Dimana dalam nilai setiap piksel dalam gambar memiliki rentang 0 sampai dengan 255 maka setiap nilai data citra gambar akan dikali $1/255$ [21]. *Horizontal* merupakan parameter yang berguna untuk membalik gambar secara *horizontal* dengan acak dan begitu juga pada *vertical*. Pada tahapan *horizontal* jika diatur sebagai *True*, maka gambar akan di balik secara *horizontal* dan *vertical* dengan kemungkinan 50% yang berarti gambar juga bisa tidak terbalik secara sempurna. *Featurewise_std_normalization* adalah parameter yang digunakan untuk menormalisasikan fitur citra gambar dengan menggunakan *standard* deviasi pada gambar. Setelah *preparing* data maka masuklah ke dalam tahap *feature extraction* dan juga *classification*.

4.1.1.3 Feature Extraction

Feature Extraction adalah adalah proses untuk mengambil fitur atau karakteristik dari data yang dimasukkan (dapat berupa teks atau gambar). Fitur atau karakteristik yang diambil berupa sifat-sifat khusus dari data tersebut yang dapat membedakan antara data satu dengan data yang lainnya seperti tekstur, warna, bentuk, dan pola. Secara umum proses ekstraksi fitur pada CNN langsung dimulai

dengan memuat model arsitektur yang digunakan. Hal ini disebabkan karena pada CNN itu ekstraksi fiturnya menjadi satu dengan metode klasifikasinya secara otomatis [22]. Kemudian, dataset yang akan diuji dimuat di dalam model tersebut untuk diekstraksi fiturnya dengan model arsitektur yang digunakan. Pada aplikasi BatiQu ini, ekstraksi fiturnya menggunakan basis arsitektur ResNet152 V2, ResNet50 V2, ResNet101 V2 atau MobileNet V1 yang sudah dilatih dengan menggunakan dataset *ImageNet*. Penggunaan bobot ImageNet ini dapat membantu dalam proses ekstraksi fitur dan dapat meningkatkan performa dari sistem klasifikasi BatiQu.



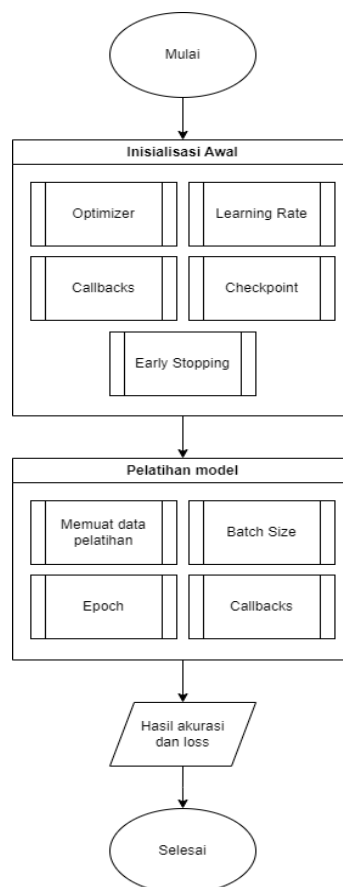
Gambar 4.3 *Flowchart Feature Extraction*

Seperti yang terlihat pada *flowchart* pada gambar 4.3, *flowchart* tersebut menjelaskan tentang alur program dalam melakukan ekstraksi fitur dengan arsitektur ResNet152 V2, ResNet50 V2, ResNet101 V2 atau MobileNet V1. Secara

keseluruhan arsitektur ResNet152 V2, ResNet50 V2, ResNet101 V2, dan MobileNet V1 melalui proses yang sama dalam menjalankan ekstraksi fitur.

4.1.1.4 Classification

Tahapan terakhir dari proses pemodelan setelah melewati tahapan *pre-processing* dan *feature extraction* adalah *classification*. *Classification* atau klasifikasi pada metode *Convolutional Neural Network* (CNN) memiliki suatu pengertian sebagai proses pengelompokan dan identifikasi suatu citra atau objek gambar ke dalam kelas tertentu. Klasifikasi menjadi bentuk pembelajaran dalam metode CNN agar menempatkan kelas-kelas berdasarkan fitur yang telah dipelajari setelah melewati pelatihan atau proses *training*. Dalam hal ini, klasifikasi yang akan digunakan dari hasil akhir *feature extraction* adalah arsitektur ResNet152 V2. Arsitektur ResNet152 V2 memiliki hyperparameter dan juga parameter yang akan digunakan seperti *Optimizer*, *Learning Rate*, *Callbacks*, *Batch Size*, dan *Epoch*. Nilai hyperparameter serta parameter dapat memengaruhi *output* model dari proses pelatihan programnya.



Gambar 4.4 Flowchart alur proses klasifikasi

Pada gambar 4.4 menjelaskan mengenai *flowchart* alur proses klasifikasi diawali dengan inisialisasi variabel yang diperlukan. Tahap awal dimulai dengan menambahkan *optimizer*, *learning rate*, *callbacks*, *checkpoint* dan *early stopping*. Dalam tahapan ini, *optimizer* digunakan untuk mengoptimalkan bobot model agar *loss* yang dihasilkan antara nilai *output* dengan hasil nilai dari *neuron* dapat diminimalisir saat proses *training* [18][23]. Sedangkan, *learning rate* merupakan *hyperparameter* untuk mengatur nilai perubahan dari bobot model selama proses pelatihan. Selanjutnya, terdapat parameter *callbacks* yang berfungsi untuk memanggil fungsi *checkpoint* dan *early stopping*. *Checkpoint* sendiri memiliki peran untuk menyimpan hasil performa terbaik pada saat pembelajaran model. *Early stopping* juga membantu jalannya pelatihan saat sistem perlu dihentikan untuk mencegah terjadinya *overfitting* [24][25].

Selanjutnya adalah mengatur pelatihan dari model yang telah dibuat. Tahapan pertama adalah memuat data yang akan digunakan untuk melakukan proses *training*, *validation*, dan *testing*. Setelah itu, dilakukan pengaturan pada *batch size* dan *epoch*. *Batch size* berfungsi untuk menginterpretasikan jumlah data pelatihan dalam satu batch yang dapat memengaruhi tingkat akurasi. Sedangkan, *epoch* merupakan jumlah pemrosesan model untuk dari seluruh dataset yang telah dilatih oleh program.

Dalam penelitian ini, terdapat parameter performansi yang dapat digunakan sebagai tolak ukur dalam melihat kinerja dari sistem tersebut. Parameter-parameter tersebut di antaranya adalah *confusion matrix* yang digunakan untuk melihat hasil pengujian kinerja serta kualitas dari model yang telah dilatih untuk proses klasifikasi. Hasil dari *confusion matrix* dapat digunakan sebagai perhitungan dalam nilai *accuracy*, *precision*, *recall*, dan *F1-Score* [26]. Berikut merupakan tabel 4.1 yang menjelaskan mengenai *confusion matrix*:

Tabel 4.1 Tabel *Confusion Matrix*

| <i>Confusion Matrix</i> | | <i>True Class</i> | |
|-------------------------|-----------------|----------------------------|----------------------------|
| | | <i>Positive</i> | <i>Negative</i> |
| <i>Predicted Class</i> | <i>Positive</i> | <i>True Positive</i> (TP) | <i>False Positive</i> (FP) |
| | <i>Negative</i> | <i>False Negative</i> (FN) | <i>True Negative</i> (TN) |

Accuracy merupakan parameter atau patokan yang digunakan untuk melihat tingkat keakuratan dan prediksi yang benar (*positive* dan *negative*) dari model klasifikasi yang telah dibuat [26][27]. Berikut merupakan persamaan untuk memperoleh nilai *accuracy* pada 4.1:

$$Accuracy = \frac{(TP + TN)}{TP + FP + FN + TN} \quad (4.1)$$

Precision merupakan nilai perbandingan antara prediksi benar positif dengan semua hasil prediksi positif yang benar maupun salah pada pembelajaran model [27]. Berikut merupakan persamaan untuk memperoleh nilai *precision* pada 4.2:

$$Precision = \frac{TP}{TP + FP} \quad (4.2)$$

Recall atau *sensitivity* merupakan nilai untuk melihat perbandingan antara prediksi benar positif dengan seluruh data prediksi benar positif [26]. Berikut merupakan persamaan untuk memperoleh nilai *recall* pada 4.3:

$$Recall = \frac{TP}{TP + FN} \quad (4.3)$$

F1-Score merupakan perhitungan nilai antara *precision* dan *recall* untuk menunjukkan keseluruhan performansi model klasifikasi yang telah dipelajari [27]. Berikut merupakan persamaan untuk memperoleh nilai *F1-Score* pada 4.4:

$$F1 - Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (4.4)$$

Selain persamaan untuk melakukan pengukuran performansi seperti di atas, terdapat kasus lain dimana *confusion matrix* dapat bersifat lebih dari dua kelas atau biasa disebut *confusion matrix multiclass*. Metode ini merepresentasikan konsep yang sama namun pengukuran kinerja model klasifikasi yang dibuat dapat digunakan pada beberapa kelas. Berikut tabel 4.2 yang merupakan contoh *confusion matrix multiclass* untuk enam kelas:

Tabel 4.2 *Confusion Matrix 6 class*

| | | | | | | | |
|--------------|----|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| Actual Class | C1 | TP₁ | FP ₂ /FN ₁ | FP ₃ /FN ₁ | FP ₄ /FN ₁ | FP ₅ /FN ₁ | FP ₆ /FN ₁ |
| | C2 | FP ₁ /FN ₂ | TP₂ | FP ₃ /FN ₂ | FP ₄ /FN ₂ | FP ₅ /FN ₂ | FP ₆ /FN ₂ |
| | C3 | FP ₁ /FN ₃ | FP ₂ /FN ₃ | TP₃ | FP ₄ /FN ₃ | FP ₅ /FN ₃ | FP ₆ /FN ₃ |
| | C4 | FP ₁ /FN ₄ | FP ₂ /FN ₄ | FP ₃ /FN ₄ | TP₄ | FP ₅ /FN ₄ | FP ₆ /FN ₄ |
| | C5 | FP ₁ /FN ₅ | FP ₂ /FN ₅ | FP ₃ /FN ₅ | FP ₄ /FN ₅ | TP₅ | FP ₆ /FN ₅ |
| | C6 | FP ₁ /FN ₆ | FP ₂ /FN ₆ | FP ₃ /FN ₆ | FP ₄ /FN ₆ | FP ₅ /FN ₆ | TP₆ |
| | | C1 | C2 | C3 | C4 | C5 | C6 |
| | | Prediction | | | | | |

4.1.2 Mobile Application

Kamera pada aplikasi ini berfungsi sebagai pengambilan gambar batik melalui kamera handphone pengguna untuk selanjutnya di proses menggunakan *Deep Learning*. Setelah mengambil gambar dari kamera, gambar yang terambil akan langsung di proses oleh *Deep Learning* dan akan dialihkan ke halaman informasi sesuai jenis batik yang telah dilatih. Selanjutnya ada fitur galeri yang berfungsi sebagai pengambilan gambar batik melalui galeri *handphone*. Pengguna bisa menggunakan gambar batik yang tersedia di masing – masing galeri handphone pengguna untuk selanjutnya di proses langsung oleh metode *Deep Learning* dan dialihkan ke halaman informasi masing – masing jenis batik.

Metode *Deep Learning* dibuat dengan arsitektur ResNet152 V2. Arsitektur ResNet152 V2 digunakan karena model ini dapat menghasilkan performa yang lebih baik dibandingkan arsitektur ResNet50 V2, ResNet101 V2 dan MobileNet

V1. Metode ini dilatih menggunakan 660 gambar sebagai dataset utama. Metode *Deep Learning* dapat membedakan enam jenis batik dari gambar yang telah diambil.

Implementasi yang dilakukan adalah berupa implementasi antarmuka yang dilakukan pada setiap fitur. Implementasi antarmuka dilakukan dengan XML (*Extensible Markup Language*) dan disimpan dalam bentuk file XML (*Extensible Markup Language*). Implementasi lain yang dilakukan adalah berupa implementasi pada *Deep Learning*. Implementasi dilakukan untuk membuat aplikasi dapat memprediksi dari gambar batik yang diambil dari kamera atau *gallery* sesuai yang dibuat pada rangkaian *Deep Learning*. Implementasi ini memiliki keterbatasan pada jenis batik dimana pada *Deep Learning* yang dibuat hanya mampu memprediksi enam kelas yaitu batik parang, batik ceplok, batik nitik, batik megamendung, batik tambal, dan batik kawung. Pembatasan dilakukan karena dataset batik yang tersedia itu terbatas dan kualitas gambar yang didapat tidak merata sehingga hanya dipilih enam jenis batik.

4.1.3 BatiQu Box

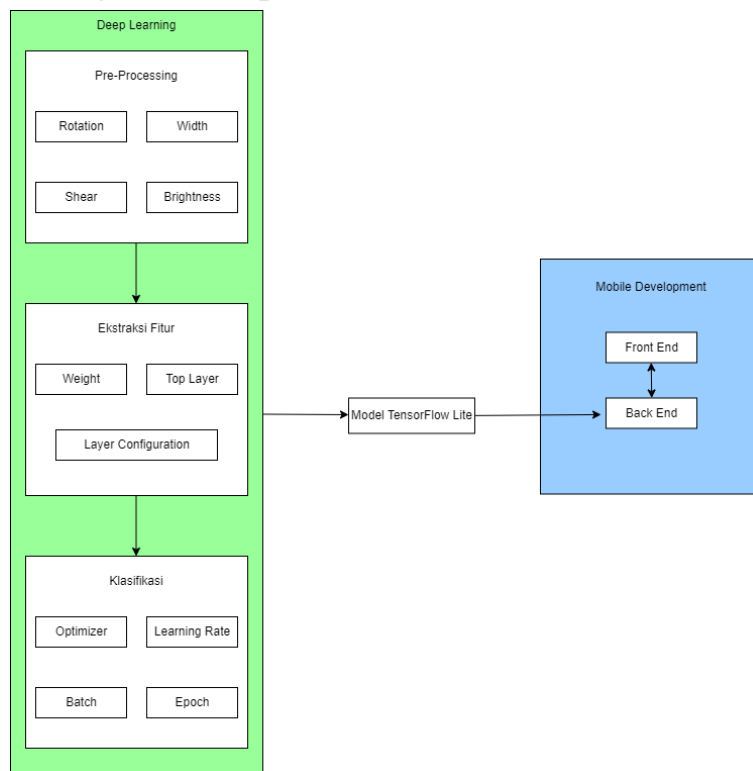
BatiQu Box digunakan untuk membantu proses klasifikasi batik terhadap kelas-kelasnya serta menjaga pencahayaan dan posisi pengambilan sampel tidak berubah. BatiQu Box terbuat dari kayu dengan tiga bagian. Bagian pertama terletak pada posisi paling atas yang berfungsi sebagai penutup atas. Bagian pertama disertai dengan lubang untuk pengambilan gambar dengan kamera *handphone* dengan diameter dua cm serta terdapat rangkaian LED sebagai penerangan utama. Selanjutnya pada bagian kedua terletak pada tengah yang berfungsi sebagai jarak antara kamera dan kain batik. Bagian terakhir pada *box* terletak pada bawah sekaligus sebagai penutup bagian bawah dan berfungsi sebagai penghalang cahaya luar agar tidak merubah hasil pengambilan gambar.



Gambar 4.5 Implementasi *Box* secara real

Gambar di atas menunjukkan wadah dari *Box* BatiQu yang bertujuan untuk menampung rangkaian LED dan kain batik yang akan diuji. Dari gambar di atas terlihat bagian dari BatiQu *Box* berurutan dari sebelah kiri yaitu bagian atas yang menampung rangkaian LED dan terdapat lubang untuk kamera *handphone*. Lalu, ada bagian tengah *box* yang berfungsi untuk memberikan jarak antara kamera *handphone* dengan kain batik yang diuji. Terakhir terdapat bagian bawah sebagai tutup dari *Box* BatiQu.

4.2 Analisis Pengerjaan Implementasi Sistem



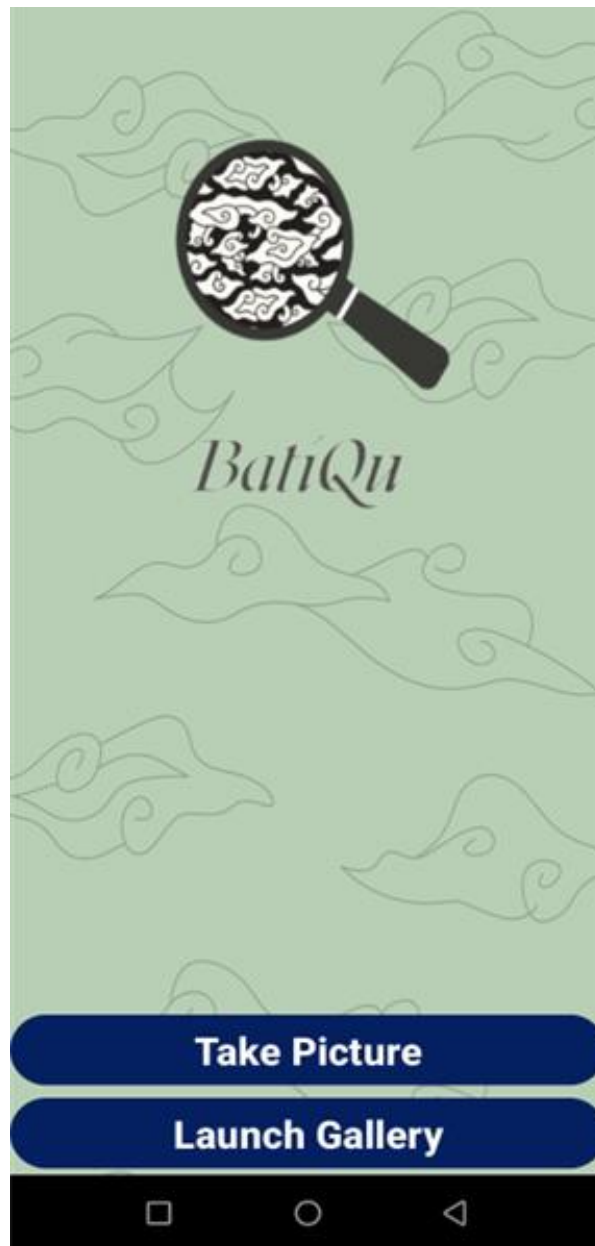
Gambar 4.6 Implementasi Sistem

Pada gambar 4.6 di atas menunjukkan diagram implementasi sistem aplikasi BatiQu. Sistem terdiri dari dua subsistem utama yaitu *Deep Learning* dan *Mobile Development*. Subsistem *Deep Learning* adalah proses awal dari pengolahan gambar jenis batik yang selanjutnya akan di proses pada subsistem *Mobile Development* bersamaan dengan BatiQu box. Proses pada subsistem *Deep Learning* dimulai dari tahap *pre-processing* yang mana pada tahap tersebut terdapat proses *Rotation, Width, Shear dan Brightness*. Selanjutnya ada tahap ekstraksi fitur dengan proses *Weight, Top Layer dan Layer Configuration*. Lalu tahap terakhir dari sub sistem *Deep Learning* adalah Klasifikasi. Proses dari tahap Klasifikasi yaitu *Optimizer, Learning Rate, Batch, dan Epoch*. Setelah semua proses dilakukan, selanjutnya dilakukan ekstrasi file menggunakan *tensorflow lite*.

Subsistem *Mobile Development* dikembangkan setelah subsistem *Deep Learning* dibuat dan sudah dibentuk model *tensorflow lite*. Aplikasi harus bisa menyesuaikan jenis format data yang diuji agar bisa di proses oleh model *tensorflow lite*. Selanjutnya subsistem *Mobile development* dijalankan bersama dengan BatiQu Box untuk membantu pengujian dari semua subsistem yang telah dijelaskan. Setelah itu hasil dari pengerjaan *Mobile development* dapat terlihat hasilnya.

4.3 Hasil Akhir Sistem

1. Tampilan antarmuka Kamera dan Gallery



Gambar 4.7 Tampilan kamera dan *gallery*

Pada gambar 4.7 di atas merupakan tampilan antarmuka dari fitur utama aplikasi BatiQu yaitu Kamera dan *Gallery*. Tampilan antarmuka ini juga berfungsi sebagai halaman uji dari klasifikasi jenis batik. Untuk menggunakan fitur ini, pengguna cukup tekan tombol *Take Picture* atau *Launch Gallery* dan pengguna bisa langsung ambil gambar batik baik dengan cara memotret kain batik maupun dari *gallery handphone* pengguna.

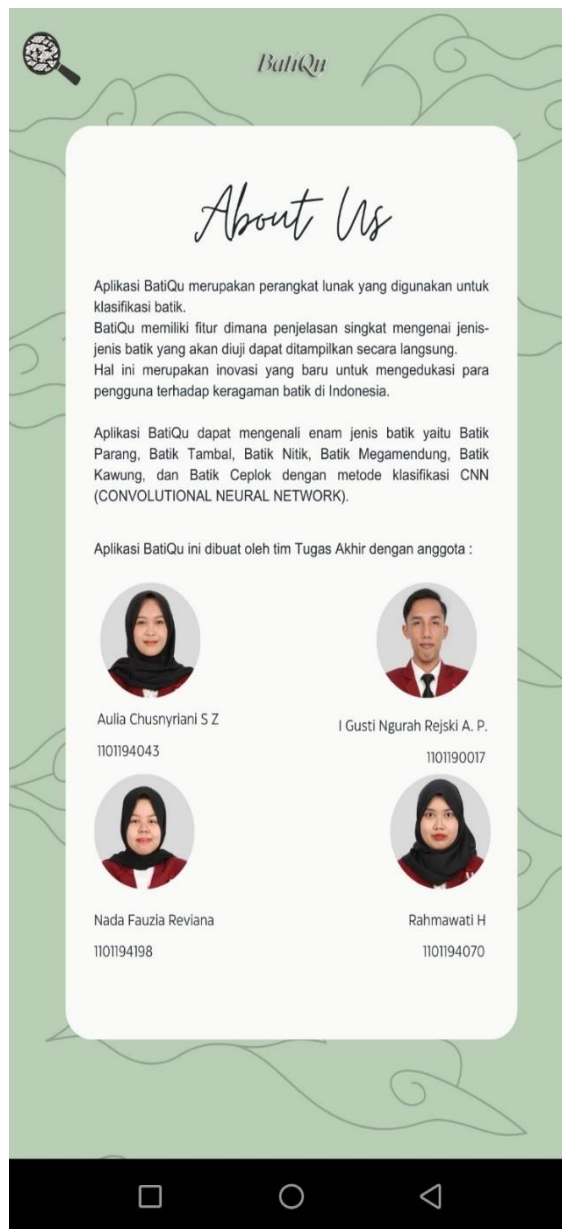
2. Tampilan antarmuka Halaman Informasi Jenis Batik



Gambar 4.8 Halaman Informasi

Gambar 4.8 di atas adalah salah satu contoh dari hasil klasifikasi jenis batik. Halaman ini akan tampil setelah gambar batik berhasil diambil dan proses klasifikasi dapat berjalan. Halaman informasi batik ini bisa berubah sesuai dengan gambar batik yang diujikan.

3. Tampilan Antarmuka *About Us*



Gambar 4.9 Halaman *About Us*

Pada gambar 4.9 di atas adalah tampilan antarmuka dari halaman *About Us*. Halaman ini akan tampil setelah pengguna menekan tombol *About Us* pada *Homepage*. Halaman *About Us* berisikan definisi dari aplikasi seperti kegunaan dari aplikasi BatiQu dan terdapat data diri dari pembuat aplikasi BatiQu.

4.4 Kesimpulan dan Ringkasan CD-4

Implementasi dari *Deep Learning* dimulai dengan pengumpulan data. Data yang dikumpulkan adalah gambar batik dengan jumlah 660 gambar yang

dikelompokkan dalam enam kelas yaitu batik kawung, batik ceplok, batik tambal, batik megamendung, batik parang dan batik nitik. Data batik yang telah dikumpulkan terbagi lagi menjadi 3 data bagian yaitu data *training*, *testing* dan juga data *validasi*. Selanjutnya terdapat proses *pre-processing* yang menjadi awal dari pengolahan data gambar pada *Convolutional Neural Network* (CNN). Proses selanjutnya adalah augmentasi data yang bertujuan untuk memodifikasi dan menciptakan variasi sample data baru. Dari proses augmentasi, data batik yang asli berjumlah 660 menjadi 3300 citra yang terbagi pada 3 data bagian. Implementasi dari proses *Feature Extraction*. Proses ini menggunakan arsitektur MobileNet V1, ResNet50 V2, ResNet101 V2, dan ResNet152 V2 sebagai perbandingan. Hasil dari perbandingan arsitektur nantinya akan dimasukkan pada *Mobile Application* dan menjadi hasil akhir proses klasifikasi. Dalam hasil proses ini menggunakan arsitektur ResNet152 V2 yang memiliki beberapa hyperparameter dan juga parameter yang akan digunakan seperti *Optimizer*, *Learning Rate*, *Callbacks*, *Batch Size*.

Implementasi dari *Mobile Application* BatiQu terdapat dua bagian yaitu Fitur Utama dan Fitur Tambahan. Fitur Utama dari *Mobile Application* BatiQu adalah kamera dan galeri diikuti dengan halaman informasi jenis batik. Proses klasifikasi dari gambar batik yang telah diambil dimulai dari *pre-processing*, ekstraksi fitur, klasifikasi dan dilanjut dengan hasil halaman informasi jenis batik. Fitur Tambahan dari *Mobile Application* adalah halaman antarmuka *About Us*. Pengujian dari aplikasi BatiQu dilihat dari pengambilan gambar, dilanjut dengan proses klasifikasi sampai beralih ke halaman informasi jenis batik.

Implementasi terakhir terdapat BatiQu *Box* yang terdiri dari tiga bagian utama dalam pembuatannya. Bagian atas BatiQu *Box* terdiri dari lubang kamera dan rangkaian LED sebagai penerangan utama. Selanjutnya bagian tengah BatiQu *Box* yang berfungsi sebagai jarak antara kamera dan kain batik. Bagian terakhir dari BatiQu *Box* yaitu penutup yang berfungsi sebagai penghalang cahaya luar agar tidak merubah hasil pengambilan gambar. Pengujian dari BatiQu *Box* dilakukan bersamaan dengan aplikasi BatiQu.

BAB 5

PENGUJIAN SISTEM

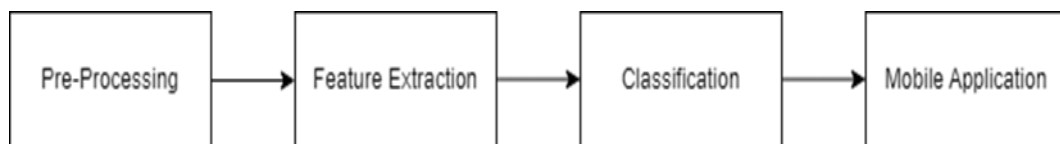
5.1 Skema Pengujian Sistem

5.1.1 Pengujian SUS keseluruhan fitur

Pengujian ini dilakukan untuk melihat fungsionalitas dari sebuah aplikasi. Model pengujian yang digunakan adalah SUS (*System Usability Scale*). SUS (*System Usability Scale*) adalah sebuah alat uji yang dikembangkan oleh John Broke. Konsep dari alat uji ini adalah menggunakan sepuluh pertanyaan global tentang ketergunaan dari aplikasi. Setiap pertanyaan terdapat 5 skala penilaian yaitu “Sangat Tidak Setuju”, “Tidak Setuju”, “Ragu-ragu”, “Setuju”, dan “Sangat Setuju” [28]. Penilaian ini membuatuhkan jawaban dari responden untuk menilai performa dari aplikasi. Skor yang baik pada pengujian ini berada di atas 68.

5.1.2 Pengujian Deep Learning

Pengujian selanjutnya akan dilakukan untuk melihat performa dari model yang sudah dirancang sebelumnya. Pengujian bertujuan untuk menghasilkan output dari sampel citra gambar yang sudah dilatih menggunakan sistem. Adapun skema pengujian seperti yang dipaparkan pada gambar 5.1 berikut:



Gambar 5.1 Skema pengujian *Deep Learning*

Pengujian ini juga dilakukan menggunakan spesifikasi hardware yang dapat dilihat pada tabel 5.1 di bawah ini:

Tabel 5.1 Spesifikasi Hardware

| | |
|--------------------------------|--------------------------------|
| <i>Processor</i> | Intel Core i7-9750H |
| <i>RAM</i> | 16 GB DDR4 |
| <i>Storage</i> | 512 GB SSD M.2 NVME + 1 TB HDD |
| <i>Operating System</i> | Windows 11 |

5.2 Proses Pengujian

Proses pengujian adalah proses yang digunakan sebagai tolak ukur dari penilaian dan mengevaluasi dari kinerja suatu sistem melalui beberapa sub-sistem yang telah dirancang sebelumnya. Tujuan utama dalam pengujian adalah untuk memastikan perancangan yang telah dibuat dapat berjalan sesuai dengan kebutuhan. Untuk mengetahui kualitas sistem dapat dilihat dari pengujian performa yang akan dilakukan yaitu pada bagian *pre-processing*, *feature extraction*, klasifikasi dan *mobile application*.

5.2.1 Pre-processing

Tahap *pre-processing* data merupakan awal suatu proses yang dilakukan dengan memiliki tujuan untuk mendapatkan data citra yang lebih baik dan lebih banyak yang nantinya data *output* tersebut akan dilanjutkan pada tahap berikutnya. Berikut merupakan penjelasan dari *pre-processing* yang digunakan:

1. *Brightness*

Brightness merupakan parameter yang digunakan untuk mengatur rentang kecerahan dengan menghasilkan variasi gambar dengan tingkat kecerahan yang berbeda-beda. Pada '*brightness*' yang digunakan memiliki *tupple*. Dimana *tupple* berisikan dua nilai yaitu (0.1, 1.5), maka dapat dijelaskan bahwa kecerahan gambar bervariasi antara 50% sampai dengan 150% dari kecerahan asli data gambar yang dimasukkan.

2. *Rotation*

Rotation merupakan parameter yang berguna untuk mengubah rotasi gambar secara acak. Tujuan dari *rotation* untuk membantu dalam meningkatkan variasi citra gambar guna mempermudah dalam pembelajaran model selanjutnya. Jika '*rotation*' bernilai 0.1 yang berarti 10% maka menunjukkan gambar dapat diputar dalam rentang -10 hingga 10 derajat secara acak.

3. *Shear*

Shear merupakan parameter yang digunakan untuk memiringkan posisi citra gambar atau mentransformasi pergeseran gambar sehingga gambar bisa berbentuk

jajar genjang [29]. Contohnya, jika menggunakan ‘shear’ bernilai 0.2 maka citra gambar dimiringkan secara acak hingga 20% dari gambar sebelumnya.

4. *Width*

Width merupakan parameter yang digunakan untuk melakukan pergeseran posisi pada lebar gambar. Pergeseran dilakukan pada gambar acak dimana gambar tersebut akan diubah posisinya ke kiri atau kekanan. Contohnya, jika menggunakan ‘width’ bernilai 0.2 maka citra gambar digeser secara acak hingga 20% dari lebar total dari gambar sebelumnya.

Setelah penjelasan bobot-bobot yang digunakan sebagai parameter dalam *pre-processing*. Berikut adalah langkah-langkah pengujian dimana pengujian *Pre-processing*:

1. Melakukan *pre-processing* secara augmentasi data pada setiap parameter.
2. Setiap hasil dari *pre-processing* akan dimasukkan kedalam *folder* yang berbeda-beda.
3. Yang terakhir, setiap *pre-processing* diuji dengan melihat hasil *acuraccy* dan *loss* pada sistem serta dipilih yang terbaik untuk dilanjutkan kepengujian pada tahap selanjutnya.

5.2.2 *Feature Extraction*

Feature extraction memiliki beberapa parameter yang bisa diuji dan dibandingkan untuk mengetahui parameter mana yang dapat menghasilkan performa yang paling optimal. Parameter dari *feature extraction* tersebut di antaranya ‘*include_top*’ dan *pooling layer*.

1. Parameter ‘*include_top*’

‘*include_top*’ merupakan parameter yang mengatur kondisi *output* yang terhubung dengan layer *top layer* [30]. *Top layer* atau yang sering disebut sebagai *fullu connected layer* adalah lapisan yang bertanggung jawab dalam mengambil fitur-fitur yang telah diekstraksi untuk digunakan pada proses klasifikasi [31]. Kondisi ini diatur dengan menulis argumen ‘*include_top=True*’ untuk menyertakan *top layer* dengan *output*, dan ‘*include_top=False*’ untuk menghilangkan *top layer* tersebut.

Langkah pengujian :

Skenario pengujian parameter '*include_top*' dilakukan dengan menggunakan dua jenis argumen yaitu '*include_top=False*' dan '*include_top=True*'. Dari pengujian tersebut akan dilihat perbandingan dari hasil akurasi dan lossnya.

2. Parameter *pooling layer*

Pooling layer merupakan parameter yang mengatur jenis operasi *pooling* yang berfungsi untuk mengurangi dimensi spasial sehingga nilai-nilai fitur yang dihasilkan layer konvolusi dapat digabung menjadi satu nilai/dimensi [32]. Pengurangan dimensi spasial pada *pooling layer* juga dapat mengurangi daya komputasi yang diperlukan untuk memproses data melalui pengurangan dimensi dari *feature map* (*downsampling*), sehingga waktu komputasi menjadi semakin cepat [32]. Terdapat beberapa operasi *pooling* yang bisa diterapkan pada model klasifikasi yang dibuat, seperti *max pooling*, *average pooling*, dan *global average pooling*.

Ketiga operasi pooling ini memiliki karakteristiknya masing-masing. Pertama ada *max pooling* yang merupakan operasi pooling yang bekerja dengan cara memilih nilai maksimum dari area *feature map* yang terjangkau oleh filter. Lalu ada *average pooling* yang merupakan operasi pooling yang bekerja dengan cara memilih nilai rata-rata dari area *feature map* yang terjangkau oleh filter. Filter pada *max pooling* dan *average pooling* dapat diatur dengan menggunakan *pool size* yang ukurannya disesuaikan dengan model yang dibuat dan tugas yang dilakukan. Yang terakhir yaitu *global average pooling*. Cara kerja dari *global average pooling* hampir sama dengan *average pooling*, namun perbedaannya terletak pada ruang lingkup operasinya. *Global average pooling* mengambil nilai rata-rata dari seluruh area *feature map*.

Langkah pengujian :

Skenario pengujian parameter *pooling layer* dilakukan dengan menggunakan dua jenis metode yaitu *max pooling* dan *average pooling*. Pengujian akan dijalankan dengan kondisi nilai *epoch*=100 dan akan menggunakan beberapa nilai *pool size* yang berbeda-beda. Dari pengujian tersebut akan dilihat perbandingan dari hasil akurasi dan lossnya.

5.2.3 Classification

Tahap klasifikasi merupakan proses pengelompokan suatu citra ke dalam kelas-kelas tertentu. Pengujian ini memiliki cara kerja yaitu dataset batik yang telah dikumpulkan akan diklasifikasikan sesuai kelasnya. Hal ini didukung menggunakan perancangan metode *Convolutional Neural Network* atau CNN dengan arsitektur ResNet152 V2. Beberapa *hyperparameter* yang digunakan di antaranya *optimizer*, *learning rate*, *batch size*.

1. *Optimizer*

Optimizer merupakan algoritma yang digunakan untuk memaksimalkan bobot model agar menekan *loss* yang didapatkan antara nilai keluaran dengan nilai dari *neuron*. *Optimizer* dalam metode CNN memiliki beberapa jenis seperti RMSprop, Adam, Adamax, dan Nadam. *Optimizer* RMSprop merupakan modifikasi dari *optimizer* Adagrad dan berfungsi untuk mengubah akumulasi gradien menjadi perubahan adaptif dalam proses optimasi [33]. Selanjutnya, terdapat *optimizer* Adam yang merupakan perluasan dari algoritma sebelumnya yaitu SGD klasik dan dapat memperbarui bobot pada setiap layer [30]. Selain itu, *optimizer* Adamax merupakan modifikasi dan pengembangan dari *optimizer* Adam yang dapat mempercepat proses pembelajaran model dibandingkan menggunakan Adam [34]. Terakhir adalah *optimizer* Nadam yang menggabungkan dua algoritma optimasi antara Adam dengan momentum Nesterov [35]. *Optimizer* Nadam sendiri bersifat lebih stabil selama proses pelatihan.

Langkah pengujian:

Skenario pengujian pada *optimizer* dilakukan dengan membandingkan hasil *accuracy* dan *loss* dari penggunaan *optimizer* RMSprop, Adam, Adamax, dan Nadam. Pengujian ini menggunakan *hyperparameter* yang telah digunakan sebelumnya pada proses *pre-processing* dan *feature extraction* dengan *learning rate* 0,0001 dan *batch size* 64. Hasil dari pengujian ini, *optimizer* dengan performa terbaik akan digunakan sebagai *hyperparameter* untuk pengujian berikutnya pada *learning rate* dan *batch size*.

2. *Learning Rate*

Learning rate adalah salah satu *hyperparameter* yang digunakan untuk membantu saat proses *training*. *Learning rate* sendiri dapat diimplementasikan untuk menghitung nilai ketepatan bobot citra atau objek gambar. Pemilihan pada nilai *learning rate* mempengaruhi kecepatan dan kestabilan dalam pelatihan model [23].

Langkah pengujian:

Skenario pengujian pada *learning rate* dilakukan dengan menggunakan empat jenis nilai yaitu 0.01, 0.001, 0.0001, dan 0,00001. Pengujian ini menggunakan *optimizer* terbaik dari hasil pengujian sebelumnya dengan *batch size* 64.

3. *Batch Size*

Batch size bertugas untuk menentukan jumlah data pelatihan dalam setiap iterasi saat proses pelatihan. Nilai dari *batch size* dapat memengaruhi tingkat akurasi yang akan dihasilkan nantinya [36].

Langkah pengujian: Skenario pengujian pada *batch size* dilakukan dengan menggunakan empat jenis nilai yaitu 16, 32, 64, dan 128.

5.2.4 *Mobile Application*

Mobile application memiliki beberapa parameter yang diuji untuk melihat apakah dari parameter yang diujikan dapat berjalan dengan lancar. Parameter yang dapat diujikan yaitu dari fitur utama *Mobile Application* yaitu parameter kamera, *gallery* dan halaman informasi serta dari fitur tambahan yaitu *About Us*. Lalu terdapat parameter lain yang diujikan seperti uji fungsionalitas seluruh Fitur Aplikasi dengan menggunakan metode SUS (*System Usability Scale*).

5.2.5 Pengujian pada Fitur Utama Aplikasi

a. Pengujian Tombol Kamera

Untuk pengujian tombol kamera pada aplikasi dilakukan pada *Android Studio* dan *handphone*. Langkah pengujian sebagai berikut:

1. Buka *Android Studio*.
 2. Buka Project yang telah dibuat sebelumnya.
 3. Tekan tombol *play* untuk *run* pada emulator yang disediakan oleh *Android Studio*.
 4. Tekan tombol Mulai untuk memulai aplikasi.
 5. Tekan tombol kamera dan mulai potret dari simulasi yang telah disediakan oleh *Android Studio*.
 6. Gambar yang telah dipotret akan ditampilkan kembali pada *ImageView* aplikasi.
 7. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan *debugging* pada *handphone*.
 8. *Debugging* pada *handphone* dilakukan menggunakan kabel USB sebagai perantara antara *Android Studio* dengan *handphone*.
- b. Pengujian tombol *Gallery*

Untuk pengujian tombol *gallery* pada aplikasi dilakukan pada *Android Studio* dan *handphone*. Langkah pengujian sebagai berikut :

1. Buka *Android Studio*.
 2. Buka *Project* yang telah dibuat sebelumnya.
 3. Tekan tombol *play* untuk *run* pada emulator yang disediakan oleh *Android Studio*.
 4. Tekan tombol Mulai untuk memulai aplikasi.
 5. Tekan tombol galeri dan mulai pilih gambar dari galeri yang telah disediakan oleh *Android Studio*.
 6. Gambar yang telah dipilih akan ditampilkan kembali pada *ImageView* aplikasi.
 7. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan *debugging* pada *handphone*.
 8. *Debugging* pada *handphone* dilakukan menggunakan kabel USB sebagai perantara antara *Android Studio* dengan *handphone*.
- c. Pengujian Halaman Informasi

Untuk pengujian halaman informasi pada aplikasi dilakukan *pada Android Studio dan handphone*. Pengujian ini dilakukan setelah melakukan pengambilan gambar untuk selanjutnya di klasifikasikan oleh *Deep Learning*. Langkah pengujian sebagai berikut:

1. Buka *Android Studio*.
2. Buka *Project* yang telah dibuat sebelumnya.
3. Tekan tombol *play* untuk *run* pada emulator yang disediakan oleh *Android Studio*.
4. Tekan tombol mulai untuk memulai aplikasi.
5. Tekan tombol kamera atau galeri dan mulai pilih gambar.
6. Dari gambar yang dipilih akan langsung di proses oleh *Deep Learning* dan selanjutnya akan diarahkan pada halaman informasi yang dibuat sesuai kelas batik.
7. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan *debugging* pada *handphone*.
8. *Debugging* pada *handphone* dilakukan menggunakan kabel USB sebagai perantara antara *Android Studio* dengan *handphone*.

5.2.6 Pengujian pada Fitur Tambahan Aplikasi

b. Pengujian Tombol *About Us*

Untuk pengujian tombol kamera pada aplikasi dilakukan pada *Android Studio* dan *handphone*. Langkah pengujian sebagai berikut:

1. Buka *Android Studio*.
2. Buka *Project* yang telah dibuat sebelumnya.
3. Tekan tombol *play* untuk *run* pada emulator yang disediakan oleh *Android Studio*.
4. Tekan tombol *About Us* untuk melihat laman *About Us* yang telah dibuat.
5. Halaman *About Us* berhasil ditampilkan.
6. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan *debugging* pada *handphone*.
7. *Debugging* pada *handphone* dilakukan menggunakan kabel USB sebagai perantara antara *Android Studio* dengan *handphone*.

5.2.7 Pengujian pada seluruh sistem Aplikasi

Pengujian menggunakan metode SUS (*System Usability Scale*) dilakukan dengan membuat kuesioner menggunakan *Google Form*. Kuesioner ini berisikan sepuluh pertanyaan terkait fungsionalitas dari aplikasi BatiQu beserta video tutorial dari pemakaian aplikasi BatiQu. Kuesioner disebar secara online kepada masyarakat yang telah melihat video tutorial dan telah mencoba aplikasi BatiQu dengan target responden berjumlah lima belas orang. Hasil daripada kuesioner merupakan pengalaman pribadi setelah melihat video tutorial dan mencoba aplikasi BatiQu bersamaan dengan BatiQu *Box*. Adapun sepuluh pertanyaan kuesioner terdapat pada table 5.2 di bawah ini:

Tabel 5.2 Pertanyaan Kuesioner

| No | Pertanyaan |
|----|---|
| 1 | Video Tutorial Alat mudah dipahami oleh siapa saja |
| 2 | Saya merasa sistem ini rumit untuk digunakan |
| 3 | Saya merasa sistem ini mudah untuk digunakan |
| 4 | Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini. |
| 5 | Saya merasa fitur-fitur sistem ini berjalan dengan semestinya. |
| 6 | Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini. |
| 7 | Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat. |
| 8 | Saya merasa sistem ini membingungkan |
| 9 | Saya merasa tidak ada hambatan dalam menggunakan sistem ini |
| 10 | Seberapa besar kemungkinan Anda merekomendasikan aplikasi ini? (skala 1- 10) |

5.2.8 Pengujian BatiQu Box

Pengujian dilakukan bersamaan dengan aplikasi BatiQu. Pengujian dilakukan bersamaan karena *Box* BatiQu berfungsi untuk menjaga kondisi pencahayaan agar gambar yang diambil tetap stabil. Langkah pengujian *Box* BatiQu sebagai berikut:

1. Siapkan *Box* BatiQu dan aplikasi BatiQu.
2. Siapkan kain batik yang akan diuji dan letakkan pada bagian bawah *Box*.
3. Nyalakan LED dengan tekan tombol On pada saklar di sisi samping bagian atas *Box* BatiQu.
4. Siapkan handphone yang sudah terunduh aplikasi BatiQu dengan posisi kamera sudah sesuai dengan lubang pada bagian atas *Box*.
5. Potret kain batik dan tunggu hasil dari uji batik

5.3 Hasil Pengujian

5.3.1 *Pre-processing*

- a. Pengaruh *Pre-processing* terhadap *Accuracy* Arsitektur

Tahap pertama yang dilakukan dalam pengujian adalah dengan dilakukannya pengujian pada empat jenis augmentasi data. Pada pengujian ini akan menggunakan parameter berupa *target size* berukuran 300×300 , dengan *batch size* 64, *optimizer* RMSprop, *learning rate* 0.0001 pada arsitektur MobileNet V1, ResNet50 V2, ResNet101 V2, dan ResNet152 V2. Pada saat pengujian berlangsung dataset hanya diuji menggunakan satu jenis augmentasi data. Data hasil dari augmentasi dimasukkan kedalam pengujian *pre-processing* untuk mengetahui pengaruh masing-masing jenis augmentasi data pada *accuracy*. Adapun hasil pengujian yang terbaik akan digunakan pada pengujian selanjutnya dengan arsitektur yang akan diuji pada pengujian selanjutnya. Hasil pengujian terbaik dalam augmentasi data dapat dilihat pada tabel 5.3 di bawah ini:

Tabel 5.3 (a) Hasil Pengujian *Pre-processing* MobileNet V1 (b) Hasil Pengujian *Pre-processing* ResNet50 V2 (c) Hasil Pengujian *Pre-processing* ResNet101 V2 (d) Hasil Pengujian *Pre-processing* ResNet152 V2

| MobileNet V1 | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|--------------------------------|----------------------------------|------------------------------|---------------------------------------|-----------------------------------|---------------------------------|
| <i>Brigthness</i> | 95,79% | 0,2014 | 70% | 1,6043 | 56,67% |
| <i>Rotation</i> | 94% | 0,2266 | 66,83% | 1,6305 | 57,86% |
| <i>Rotation 45°</i> | 87,71% | 0,3832 | 67% | 0,9910 | 59,33% |
| <i>Shear</i> | 95,83% | 0,1185 | 73% | 1,0695 | 65,67% |
| <i>Width</i> | 98% | 0,0596 | 64% | 1,6834 | 57,67% |

(a)

| ResNet50 V2 | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|--------------------------------|----------------------------------|------------------------------|---------------------------------------|-----------------------------------|---------------------------------|
| <i>Brigthness</i> | 100% | 0,0116 | 85,67% | 0,4741 | 76,33% |
| <i>Rotation</i> | 99,96% | 0,0164 | 82,83% | 0,5875 | 79,26% |
| <i>Rotation 45°</i> | 100% | 0,0113 | 84,67% | 0,4954 | 77,67% |
| <i>Shear</i> | 100% | 0,0154 | 88,33% | 0,4342 | 81,67% |
| <i>Width</i> | 99,96% | 0,0080 | 87,33% | 0,5562 | 81,33% |

(b)

| ResNet101 V2 | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|--------------------------------|----------------------------------|------------------------------|---------------------------------------|-----------------------------------|---------------------------------|
| <i>Brigthness</i> | 100% | 0,0096 | 98% | 0,4628 | 77,67% |
| <i>Rotation</i> | 99,96% | 0,0098 | 89,67% | 0,4773 | 75% |
| <i>Rotation 45°</i> | 99,96% | 0,0056 | 87,67% | 0,4334 | 73,33% |
| <i>Shear</i> | 100% | 0,0095 | 89,83% | 0,3912 | 86% |
| <i>Width</i> | 99,96 | 0,0079 | 87,67% | 0,4122 | 77,67% |

(c)

| ResNet152 V2 | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|--------------------------|----------------------------------|------------------------------|---------------------------------------|-----------------------------------|---------------------------------|
| <i>Brigthness</i> | 99,96% | 0,0207 | 87% | 0,5549 | 80% |
| <i>Rotation</i> | 100% | 0,0122 | 87% | 0,4549 | 79% |

| | | | | | |
|-------------------------------|--------|--------|--------|--------|------------|
| Rotation 45° | 99,87% | 0,0232 | 88,33% | 0,4916 | 80,33% |
| Shear | 100% | 0,0108 | 87,83% | 0,4199 | 88% |
| Width | 100% | 0,0129 | 85,33% | 0,5298 | 83% |

(d)

Pada table 5.3 di atas dijelaskan bahwa hasil pengujian pada arsitektur MobileNet V1, ResNet50 V2, ResNet101 V2, dan ResNet152 V2. Hasil di atas dapat dilihat bahwa seluruh jenis augmentasi data memiliki pengaruh terhadap performa dari model yang sudah dirancang sebelumnya. Untuk hasil pengujian *accuracy* yang optimal didapatkan pada model arsitektur ResNet152 V2. Dari hasil pengujian *pre-processing* secara augmentasi data di tabel hasil akurasi terbaik ada pada proses augmentasi yaitu *Shear* dengan nilai akurasi dicapai sebesar 95,83% dari data *train* dengan akurasi test bernilai 65,67% pada arsitektur MobileNet V1.

Pengujian kedua dilakukan pada arsitektur lain yaitu ResNet50 V2 dengan jenis augmentasi *shear*, hasil pengujian menghasilkan *accuracy train* sebesar 100% dengan *accuracy test* bernilai 81,67%. Pengujian ketiga juga dilakukan pada arsitektur ResNet101 V2 yang menghasilkan *train accuracy* sebesar 100% dan *test accuracy* sebesar 86%. Untuk hasil pengujian *accuracy* terbaik masih sama terdapat pada *Shear* dengan nilai akurasi dicapai sebesar 100% dari data *train* dengan *accuracy test* bernilai 88% pada arsitektur ResNet152 V2. Dengan ini menunjukkan bahwa model mampu mempelajari dengan baik pada keempat arsitektur menggunakan augmentasi data *shear* sehingga dapat lanjut ke pengujian selanjutnya.

b. Pengaruh 'Target Size' terhadap Accuracy Arsitektur

Tahap selanjutnya setelah mendapatkan hasil pengujian *pre-processing* terbaik. Tahap yang dilakukan dalam pengujian adalah dengan melakukan proses mengubah nilai *target size* dengan nilai 300, 224 dan 128. *Target size* akan menggunakan *batch size* 64, *optimizer* RMSprop, *learning rate* 0,0001 pada arsitektur MobileNet V1, ResNet50 V2, ResNet101 V2, dan ResNet152 V2. Hasil *target size* dengan jenis augmentasi *shear* yang terbaik akan digunakan pada pengujian selanjutnya dengan arsitektur yang akan diuji pada pengujian

selanjutnya. Adapun hasil dari *target size* terbaik dapat dilihat pada tabel 5.4 di bawah ini:

Tabel 5.4 (a)Hasil Pengujian *Target size* MobileNet V1 (b)Hasil Pengujian *Target size* ResNet50 V2 (c) Hasil Pengujian *Target size* ResNet101 V2 (d) Hasil Pengujian *Target size* ResNet152 V2

| <i>Target size</i> MobileNet V1 | <i>Train</i> <i>Accuracy</i> | <i>Train</i> <i>Loss</i> | <i>Validation</i> <i>Accuracy</i> | <i>Validation</i> <i>Loss</i> | <i>Test</i> <i>Accuracy</i> |
|---------------------------------------|---------------------------------|-----------------------------|--------------------------------------|----------------------------------|--------------------------------|
| 300 | 95,83% | 0,1185 | 73% | 1,0695 | 65,67% |
| 224 | 94,88% | 0,1824 | 66% | 1,5847 | 54% |
| 128 | 90,96% | 0,2723 | 64,83% | 1,6280 | 65% |

(a)

| <i>Target size</i> ResNet50 V2 | <i>Train</i> <i>Accuracy</i> | <i>Train</i> <i>Loss</i> | <i>Validation</i> <i>Accuracy</i> | <i>Validation</i> <i>Loss</i> | <i>Test</i> <i>Accuracy</i> |
|--------------------------------------|---------------------------------|-----------------------------|--------------------------------------|----------------------------------|--------------------------------|
| 300 | 100% | 0,0154 | 88,33% | 0,4342 | 81,67% |
| 224 | 99,96% | 0,0184 | 87,50% | 0,5509 | 74% |
| 128 | 99,54% | 0,0571 | 82% | 0,727 | 72,67% |

(b)

| <i>Target size</i> ResNet101 V2 | <i>Train</i> <i>Accuracy</i> | <i>Train</i> <i>Loss</i> | <i>Validation</i> <i>Accuracy</i> | <i>Validation</i> <i>Loss</i> | <i>Test</i> <i>Accuracy</i> |
|---------------------------------------|---------------------------------|-----------------------------|--------------------------------------|----------------------------------|--------------------------------|
| 300 | 100% | 0,0095 | 89,83% | 0,3912 | 86% |
| 224 | 100% | 0,0145 | 87,17% | 0,4444 | 77% |
| 128 | 99,46% | 0,0372 | 83% | 0,6793 | 72% |

(c)

| <i>Target size</i> ResNet152 V2 | <i>Train</i> <i>Accuracy</i> | <i>Train</i> <i>Loss</i> | <i>Validation</i> <i>Accuracy</i> | <i>Validation</i> <i>Loss</i> | <i>Test</i> <i>Accuracy</i> |
|---|---------------------------------|-----------------------------|--------------------------------------|----------------------------------|--------------------------------|
| 300 | 100% | 0,0108 | 87,83% | 0,4199 | 88% |
| 224 | 99,79% | 0,0301 | 84,33% | 0,5085 | 82,67% |
| 128 | 99,58% | 0,0430 | 78,50% | 0,6926 | 76,33% |

(d)

Tabel 5.4 di atas menjelaskan bahwa hasil pengujian *target size* pada arsitektur MobileNet V1, ResNet50 V2, ResNet101 V2 dan ResNet152 V2. Dari hasil pengujian tabel dapat dilihat hasil akurasi terbaik ada pada *target size* 300 di arsitektur MobileNet V1 dengan *accuracy* sebesar 95,83% dari data *train* dengan *test accuracy* bernilai 65,67%. Pengujian *target size* pada arsitektur selanjutnya dilakukan di ResNet50 V2 dengan hasil *target size* 300 yang didapatkan *train accuracy* sebesar 100% dan *test accuracy* 81,67%. Hasil pengujian ketiga dari *target size* yang telah dilakukan pada arsitektur Resnet101 V2 adalah *target size* 300 dengan hasil *train accuracy* sebesar 100% dan *test accuracy* 86%. Hasil dari pengujian terakhir *target size* pada arsitektur ResNet152 V2, hasil terbaik ada di *target size* bernilai 300 dengan menghasilkan nilai *accuracy* 100% pada data *train* dan *accuracy* 88% pada data *test*. Kesimpulan dari keempat pengujian menunjukkan bahwa *target size* dan *pre-processing* yang diujikan mampu mempelajari model dengan baik sehingga dapat lanjut ke pengujian selanjutnya.

5.3.2 Feature Extraction

- Pengaruh '*Include_top*' terhadap *Accuracy* Arsitektur MobileNet V1

Tabel 5.5 Hasil Pengujian *Include Top*

| <i>Include_top</i> | <i>Train</i> <i>Accuracy</i> | <i>Train</i> <i>Loss</i> | <i>Validation</i> <i>Accuracy</i> | <i>Validation</i> <i>Loss</i> | <i>Test</i> <i>Accuracy</i> |
|--------------------|---------------------------------|-----------------------------|--------------------------------------|----------------------------------|--------------------------------|
| True | 49,83% | 1,4502 | 20,5% | 1,7842 | 25,67% |
| False | 89,88% | 0,2868 | 60,5% | 1,2958 | 59% |

Pada tabel 5.5 di atas menjelaskan mengenai hasil pengujian parameter '*include_top*' dengan arsitektur MobileNet V1 memperoleh hasil yang optimal pada saat nilainya diatur ke *False* dengan *test accuracy* sebesar 59%. Secara keseluruhan,

hasil pengujian pada *train accuracy*, *train loss*, dan *validation loss* juga memiliki selisih yang cukup besar dengan perbedaan nilai hingga lebih dari 50%.

b. Pengaruh '*Pooling Layer*' terhadap *Accuracy* Arsitektur MobileNet V1

Tabel 5.6 Hasil Pengujian *Pooling Layer*

| <i>Pooling Layer</i> | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|-------------------------------|-----------------------|-------------------|----------------------------|------------------------|----------------------|
| <i>Max Pooling (2,2)</i> | 82,17% | 1,1372 | 50,67% | 3,345 | 51,67% |
| <i>Max Pooling (3,3)</i> | 83,33% | 0,5385 | 55,67% | 1,4713 | 48,33% |
| <i>Max Pooling (4,4)</i> | 75,63% | 1,2971 | 50,33% | 3,0275 | 53% |
| <i>Average Pooling (2,2)</i> | 87,17% | 0,6888 | 55,83% | 3,5265 | 56,33% |
| <i>Average Pooling (3,3)</i> | 78,62% | 1,724 | 44% | 5,9298 | 39,33% |
| <i>Average Pooling (4,4)</i> | 82,33% | 0,855 | 51,33% | 3,1227 | 57% |
| <i>Global Average Pooling</i> | 89,88% | 0,2868 | 60,5% | 1,2958 | 59% |
| <i>None</i> | 80,42% | 1,1102 | 51,33% | 4,3062 | 46% |

Dari tabel 5.6 di atas, terlihat bahwa hasil yang paling optimal diperoleh pada saat *pooling layer* diatur dengan menggunakan *Global Average Pooling* karena memperoleh *test accuracy* sebesar 59%.

c. Pengaruh '*Include_top*' terhadap *Accuracy* Arsitektur ResNet152 V2

Tabel 5.7 Hasil Pengujian *Include Top*

| <i>Include_top</i> | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|--------------------|-----------------------|-------------------|----------------------------|------------------------|----------------------|
| True | 72,12% | 1, 2143 | 59,83% | 1,3755 | 58,67% |

| | | | | | |
|--------------|--------|--------|--------|--------|------------|
| False | 99,92% | 0,0137 | 86,67% | 0,4628 | 85% |
|--------------|--------|--------|--------|--------|------------|

Dari table 5.7 di atas terlihat bahwa pengujian parameter '*include_top*' dengan arsitektur ResNet152 V2 memperoleh hasil yang optimal pada saat nilainya diatur ke '*False*' dengan test *accuracy* sebesar 85%. Hasil test *accuracy* antara nilai '*True*' dan '*False*' memiliki selisih yang sangat besar hingga 26,33%. Selain itu, secara keseluruhan nilai '*False*' juga unggul pada hasil pengujian yang lain seperti pada *train accuracy*, *train loss*, *validation accuracy*, dan *validation loss*.

d. Pengaruh '*Pooling Layer*' terhadap Accuracy Arsitektur ResNet152 V2

Tabel 5.8 Hasil Pengujian *Pooling Layer*

| <i>Pooling Layer</i> | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|-------------------------------|-----------------------|-------------------|----------------------------|------------------------|----------------------|
| Max Pooling (2,2) | 100% | 0,0003 | 87,83% | 0,7146 | 80% |
| Max Pooling (3,3) | 100% | 0,0001 | 88,33% | 0,8208 | 82% |
| Max Pooling (4,4) | 100% | 0,0001 | 89,5% | 0,7459 | 83% |
| Average Pooling (2,2) | 100% | 0,0377 | 90% | 0,6469 | 83,33% |
| Average Pooling (3,3) | 100% | 0,0002 | 89,17% | 0,6125 | 84,67% |
| Average Pooling (4,4) | 99,96% | 0,0012 | 88,33% | 0,5322 | 83,67% |
| Global Average Pooling | 99,92% | 0,0137 | 86,67% | 0,4628 | 85% |
| None | 100% | 1,2537 | 88,5% | 0,8271 | 82,33% |

Dari tabel 5.8 di atas, hasil pengujian *pooling layer* yang paling optimal dengan arsitektur ResNet152 V2 diperoleh pada saat diatur menjadi *Global Average Pooling* dengan nilai test *accuracy* mencapai 85%. Secara keseluruhan hasil pengujian pada pooling layer ini tidak memiliki selisih nilai yang cukup signifikan karena perbedaannya kurang dari 5%.

5.3.3 Classification

a. Pengaruh '*optimizer*' terhadap *Accuracy* Arsitektur ResNet152 V2

Tahap pertama dalam pengujian *classification* adalah membandingkan empat jenis *optimizer* yaitu RMSprop, Adam, Adamax, dan Nadam. Untuk pengujian *optimizer* akan menggunakan parameter yang telah digunakan pada tahap *pre-processing* dan *feature extraction*. Pada pengujian *optimizer* telah ditetapkan bahwa parameter yang digunakan pada tahapan *pre-processing* adalah jenis augmentasi *Shear* dengan target size 300 x 300. Pada bagian *feature extraction*, parameter yang ditetapkan yaitu *include top* '*false*' dan *Global Average Pooling*. Tahapan ini memiliki pengaturan pada *learning rate* 0.0001, *batch size* 64, *epoch* 100 dengan menggunakan *early stopping*. Berikut merupakan hasil pengujian *optimizer* terhadap arsitektur ResNet152 V2:

Tabel 5.9 Hasil Pengujian *Optimizer*

| <i>Optimizer</i> | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|------------------|-----------------------|-------------------|----------------------------|------------------------|----------------------|
| RMSprop | 99,87% | 0,019 | 89,17% | 0,4479 | 89,67% |
| Adam | 99,96% | 0,0226 | 87,17% | 0,4044 | 86,33% |
| Adamax | 99,96% | 0,0306 | 87,67% | 0,4066 | 86,67% |
| Nadam | 99,96% | 0,031 | 87,83% | 0,3948 | 84% |

Pada tabel 5.9, dapat dilihat untuk hasil pengujian *optimizer* RMSprop, Adam, Adamax, dan Nadam. Dari pengujian terhadap empat jenis *optimizer* yang telah dilakukan, hasil *test accuracy* terbaik didapatkan dengan nilai 89,67% pada *optimizer* RMSprop. Namun, untuk hasil *accuracy train* pada RMSprop bernilai 99,87% dimana kondisi tersebut lebih rendah dibandingkan *optimizer* lainnya dengan selisih 0,09. Pada tahapan pengujian selanjutnya, *optimizer* yang akan digunakan adalah RMSprop.

b. Pengaruh '*Learning Rate*' terhadap *accuracy* Arsitektur ResNet152 V2

Tahap selanjutnya setelah menentukan *optimizer* terbaik adalah pengujian pada nilai *learning rate* dengan empat perbandingan antara 0.01, 0.001, 0.0001,

0.00001. Pengaturan untuk nilai *hyperparameter* ini menggunakan *optimizer* RMSprop seperti hasil pengujian sebelumnya dengan *batch size* 64 dan *epoch* 100. Berikut merupakan hasil pengujian *learning rate* terhadap arsitektur ResNet152 V2:

Tabel 5.10 Hasil Pengujian *Learning Rate*

| <i>Learning Rate</i> | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|----------------------|-----------------------|-------------------|----------------------------|------------------------|----------------------|
| 0.01 | 99,5% | 0,0116 | 86,5% | 0,8681 | 83,33% |
| 0.001 | 100% | 0,0014 | 89,83% | 0,6017 | 86,33% |
| 0.0001 | 99,87% | 0,019 | 89,17% | 0,4479 | 89,67% |
| 0.00001 | 98,37% | 0,1281 | 87,17% | 0,4006 | 80,33% |

Tabel 5.10 di atas merupakan hasil pengujian untuk *learning rate*. Pengujian ini memiliki empat variasi *learning rate* dan dapat dilihat bahwa hasil terbaik didapatkan pada nilai 0.0001. Pengujian *learning rate* tersebut memiliki hasil *test accuracy* sebesar 89,67%. sedangkan pada *train accuracy* memiliki hasil 99,87%. Hal tersebut mempunyai nilai lebih rendah dibandingkan *learning rate* 0.001 yang mencapai 100%. Selanjutnya, pengujian akan menggunakan *learning rate* 0.0001.

c. Pengaruh 'Batch Size' terhadap *accuracy* Arsitektur ResNet152 V2

Tahap terakhir pada pengujian *classification* adalah membandingkan empat variasi *batch size* antara 16, 32, 64, dan 128. Pengujian ini menggunakan *hyperparameter* optimal yang sudah disesuaikan mulai dari *optimizer* RMSprop dan *learning rate* 0.0001. Hasil pengujian *batch size* terbaik akan berpengaruh untuk keseluruhan dari tahapan *classification*. Berikut merupakan hasil pengujian *batch size* terhadap arsitektur ResNet152 V2:

Tabel 5.11 Hasil Pengujian *Batch Size*

| <i>Batch Size</i> | <i>Train Accuracy</i> | <i>Train Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> | <i>Test Accuracy</i> |
|-------------------|-----------------------|-------------------|----------------------------|------------------------|----------------------|
| 16 | 99,96% | 0,0047 | 88,17% | 0,5251 | 86,67% |

| | | | | | |
|------------|--------|--------|--------|--------|---------------|
| 32 | 99,92% | 0,0101 | 88,33% | 0,4527 | 89,33% |
| 64 | 99,87% | 0,019 | 89,17% | 0,4479 | 89,67% |
| 128 | 100% | 0,0222 | 87,33% | 0,4512 | 84,33% |

Pada tabel 5.11 di atas merupakan hasil pengujian *batch size* 16, 32, 64, dan 128. Setelah melakukan pengujian untuk keempat nilai tersebut *batch size* dengan nilai 64 memiliki hasil akurasi tertinggi pada *test accuracy* sebesar 89,67%. Untuk hasil akhir pada pengujian *classification* digunakan *batch size* 64 dengan hasil *train accuracy* sebesar 99,87%. Sehingga, dapat disimpulkan untuk hasil pengujian *hyperparameter* yang optimal yaitu menggunakan *optimizer* RMSprop, *learning rate* 0.0001, dan *batch size* 64.

5.3.4 Mobile Application

a. Pengujian pada Kamera dan Gallery

Hasil pengujian Kamera dan *Gallery* dilakukan dengan melihat kamera dan *gallery* berhasil diakses dan gambar batik berhasil potret (jika dilakukan dengan fitur kamera) atau gambar batik berhasil diambil (jika dilakukan dengan fitur *gallery*). Gambar batik yang telah berhasil diambil akan di klasifikasikan

menggunakan *Deep Learning* dan akan dialihkan pada halaman informasi sesuai jenis batik yang diprediksikan.



Gambar 5.2 halaman pengambilan gambar


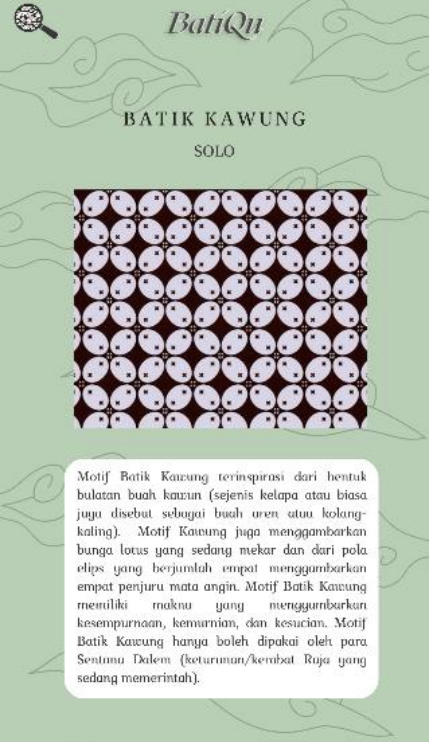
Gambar 5.2 di atas adalah halaman pengambilan gambar batik baik dari kamera maupun dari galeri. Halaman tersebut juga mejadi pengujian untuk tombol kamera dan *gallery* sebagai fitur utama dari aplikasi BatiQu. Pengujian dilakukan untuk melihat apakah kamera atau *gallery* dapat diakses dan gambar yang diambil dapat ditampilkan pada *ImageView*.





b. Pengujian pada Halaman Informasi


Pengujian dilakukan dengan menggunakan model Resnet152 dan versi dengan menggunakan model MobileNet. Halaman informasi batik berisikan informasi singkat dari jenis batik yang telah di klasifikasi kan sesuai dengan kelas yang di latih oleh *Deep Learning*. Pada halaman ini berisikan asal usul, keunikan, dan cara pakai pada acara adat. Tujuan pembuatan halaman ini untuk menambah

pengetahuan dari masing – masing jenis batik. Berikut adalah hasil dari halaman informasi dari masing – masing jenis batik yang dijelaskan pada tabel 5.12:

Tabel 5.12 Hasil halaman informasi

| No | Halaman Informasi | Jenis Batik |
|----|--|--------------|
| 1 |  <p>Batik Ceplok DAERAH ISTIMEWA YOGYAKARTA</p> <p>Batik Ceplok adalah salah satu motif batik yang sangat kuno dan terinspirasi dari motif pada hiasan di candi Hindu dan Buddha dengan bentuk lingkaran, kotak serta garis miring. Pola dasar batik ini merupakan pola dasar dari batik kawung. Batik Ceplok memiliki arti berkumpulnya segala sesuatu serta memiliki arti keseimbangan dan kesetaraan dari orang yang memakai batik ini.</p> <p>Jenis batik ceplok ada beragam seperti ceplok truntum, ceplok kawung, ceplok kesatrian, ceplok sriwedari, ceplok parang dan ceplok grampol.</p> <p>Korunkan lain dari batik ini adalah motif yang bisa diaplikasikan selain ke kain seperti hiasan cangkir, ornamen furniture, sampul buku dan lain sebagainya.</p> | Batik Ceplok |
| 2 |  <p>Batik Kawung SOLO</p> <p>Motif Batik Kawung terinspirasi dari bentuk bulatan buah kawung (sejenis kelapa atau biasa juga disebut sebagai buah aren atau kolangkaling). Motif Kawung juga menggambarkan bunga lotus yang sedang mekar dan dari pola elips yang berjumlah empat menggambarkan empat penjuru mata angin. Motif Batik Kawung memiliki makna yang menggambarkan kesempurnaan, kemurnian, dan kesucian. Motif Batik Kawung hanya boleh dipakai oleh para Sentana Dalem (keturunan/kerabat Raja yang sedang memerintah).</p> | Batik Kawung |

| | | |
|---|---|-------------------|
| 3 |  <p><i>BatiQu</i></p> <p>BATIK MEGAMENDUNG CIREBON</p>  <p>Batik Megamendung adalah salah satu motif batik dengan motif awan dengan makna setiap nuansa harus dapat menahan amarah pada dirinya pada saat dalam kondisi terpuruk, sedih dan tertekan seperti awan mendung yang sejuk. Batik Megamendung dibawa oleh bangsawan China ke wilayah Cirebon dan merujuk pada Pernikahan Sunan Gunung Jati dengan ratu Ong Tien. Dari pernikahan ini membawa kebudayaan baru yang salah satunya adalah motif awan pada piring, keramik dan kain.</p> | Batik Megamendung |
| 4 |  <p><i>BatiQu</i></p> <p>BATIK NITIK DAERAH ISTIMEWA YOGYAKARTA</p>  <p>Arti nitik dalam bahasa Jawa adalah Titik yang dipahami selayaknya tanda baca. Motif batik ini melambungkan keunikan-nya seperti daun, sulur, dan bunga, serta digambarkan membentuk pola geometris. Keunikan dari batik ini adalah pembuat batik ini adalah rukyat kaci dengan inisiatif melazim penajuban belanda pada tahun 1700-an. Batik nitik ini dibuat untuk menanggulangi mahalnya kain Patola India. Batik Nitik memiliki banyak motif sesuai dengan kekhasannya masing-masing. Beberapa motif diantaranya yaitu nitik cakar, nitik jonggrang, sekar kemuning, kembang kenthay, dan lain sebagainya.</p> | Batik Nitik |

| | | |
|---|--|--------------|
| 5 |  <p><i>BatiQu</i></p> <p>BATIK PARANG JAWA TENGAH</p>  <p>Batik parang adalah salah satu motif batik paling tua di Indonesia dengan makna pantung mengenai dalam mengarungi kehidupan serta menggambarkan jalinan hidup yang tak pernah putus. Batik Parang biasanya dijadikan sebuah lembar kain yang bernama Jarik yang berfungsi sebagai bawahan dalam acara pernikahan khususnya di pernikahan adat jawa. Namun ada jenis motif parang yang tidak bisa dijadikan motif Jarik dalam pernikahan adat jawa yaitu jenis motif batik parang rusak. Motif parang rusak tidak bisa digunakan sebagai kain Jarik karena dipercaya bisa membawa sial pada pernikahan dan membuat pernikahan penuh dengan perselisihan hingga berujung ke perpisahan.</p> | Batik Parang |
| 6 |  <p><i>BatiQu</i></p> <p>BATIK TAMBAL DAERAH ISTIMEWA YOGYAKARTA</p>  <p>Nama motif Batik Tambal berasal dari katas Tambal yang berarti menutup suatu kekurangan/menambal/memperbaiki. Batik Tambal dibuat dengan cara menggabungkan berbagai motif batik dalam bidang berbentuk segitiga yang disusun rapi. Batik Tambal memiliki makna yang mengajarkan kita untuk selalu memperbaiki diri menjadi lebih baik. Dahulu, batik tambal dipercaya bisa membantu kesembuhan orang yang sakit</p> | Batik Tambal |

Hasil pengujian dari model tersebut dapat dilihat juga pada tabel 5.13 di bawah ini:

Tabel 5.13 Hasil pengujian model Aplikasi

| No | Jenis Batik | Resnet 152 V2 | | | | |
|----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 1 | 2 | 3 | 4 | 5 |
| 1 | Kawung | Kawung | Kawung | Kawung | Kawung | Kawung |
| 2 | Parang | Parang | Parang | Parang | Parang | Parang |
| 3 | Tambal | Tambal | Tambal | Tambal | Tambal | Tambal |
| 4 | Nitik | Kawung | Kawung | Tambal | Nitik | Nitik |
| 5 | Megamen dung | Mega mendung | Mega mendung | Mega mendung | Mega mendung | Mega mendung |
| 6 | Ceplok | Tambal | Ceplok | Ceplok | Ceplok | Ceplok |

Pengujian dilakukan lima kali dengan lima skema yang berbeda. Dimulai dari skema pengujian pertama dengan menggunakan sampel gambar dari data train. Dilanjut dengan skema pengujian dengan menggunakan sampel gambar dari data *validation*. Selanjutnya adalah skema pengujian menggunakan sampel gambar dari data *test*. Tiga skema ini dilakukan untuk memastikan akurasi yang dihasilkan oleh *Deep Learning* dan Aplikasi setara. Sampel gambar yang diambil berasal dari hasil augmentasi dengan jenis *shear* dimana jenis ini membuat gambar miring ke kiri atau ke kanan namun hanya dari satu sudut saja. Pengujian dilanjut dengan pengambilan gambar menggunakan kamera dan BatiQu *box* lalu dilanjutkan dengan pengambilan gambar melalui galeri.

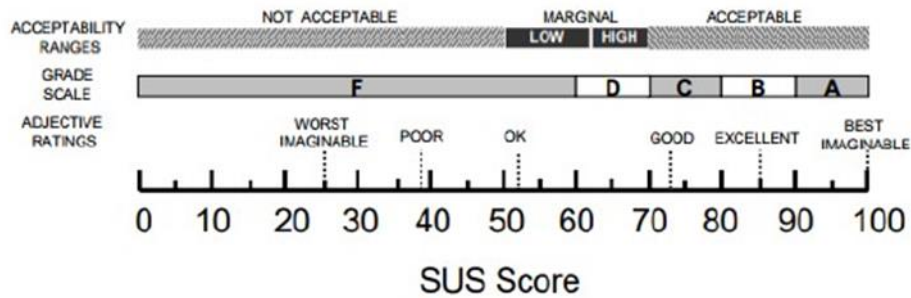
c. Pengujian tombol *About Us*



Gambar 5.3 Hasil Pengujian *About Us*

Gambar 5.3 adalah halaman *about us* yang berhasil diakses dari halaman utama aplikasi. Halaman ini berisi informasi singkat dari aplikasi dan identitas dari pembuat aplikasi BatiQu.

5.3.5 Pengujian Keseluruhan Fungsi dari Aplikasi BatiQu



Gambar 5.4 Perhitungan *score* metode SUS

Gambar 5.4 di atas [37] menunjukkan skor dari metode SUS (*System Usability Scale*) dengan skala 0 hingga 100. Penilaian ini bisa dibagi ke tiga bagian, yaitu *acceptability ranges*, *grade scale* dan *adjective rating*. Keterangan dari setiap rentang pada skor SUS (*System Usability Scale*) ini juga tertera di atasnya. Contoh pada *adjective ratings* yang memiliki keterangan pada masing – masing rentang skor seperti rentang 0 hingga 25 dengan keterangan sangat buruk. Dilanjut dengan 25 hingga 39 dengan keterangan buruk. Selanjutnya rentang 40 hingga 52 dengan keterangan OK. Dilanjut dengan 53 hingga 72 dengan keterangan *good* yang menandakan fungsionalitas dari aplikasi atau sistem yang dibuat sudah berjalan dengan baik. Rentang nilai ini juga menjadi standar nilai untuk metode SUS bagi penilaian sebuah aplikasi atau sistem yang dibuat. Pada pengujian ini menggunakan *adjective ratings* untuk melihat bagaimana kinerja aplikasi.

Dari kuisioner yang telah disebar pada responden, didapatkan masing – masing skor dari jawaban yang telah diisi. Langkah selanjutnya adalah pengumpulan skor dari masing – masing jawaban sesuai dengan keterangan pilihan jawaban. Untuk skor nilai serta keterangan pilihan jawaban dari kuisioner dapat dilihat pada tabel 5.14 di bawah ini:

Tabel 5.14 Pilihan Jawaban kuisioner

| No | Skala <i>Likert</i> | Skor Nilai |
|----|---------------------|------------|
| 1 | Sangat Tidak Setuju | 1 |
| 2 | Tidak Setuju | 2 |
| 3 | Ragu – Ragu | 3 |

| | | |
|---|---------------|---|
| 4 | Setuju | 4 |
| 5 | Sangat Setuju | 5 |

Terdapat beberapa aturan dalam perhitungan skor nilai kuisioner SUS untuk menentukan seberapa baik aplikasi dapat berjalan. Berikut aturan saat perhitungan skor nilai pada kuisioner SUS [38] :

1. Untuk setiap pertanyaan pada urutan ganjil kurangi dengan nilai satu. Contoh pertanyaan 1 memiliki skor 4. Maka kurangi 4 dengan 1 sehingga skor pertanyaan 1 adalah 3.
2. Untuk setiap pertanyaan pada urutan genap kurangi nilai dari lima. Contoh pertanyaan 2 memiliki skor 1. Maka kurangi 5 dengan 1 sehingga skor pertanyaan 2 adalah 4.
3. Tambahkan nilai-nilai dari pertanyaan bernomor genap dan ganjil. Kemudian hasil penjumlahan tersebut dikali dengan 2,5

Pernyataan di atas dapat dilihat dalam bentuk rumus perhitungan skor dalam metode SUS seperti berikut :

$$\text{skor SUS} = ((Q1 - 1) + (5 - Q2) + (Q3 - 1) + (5 - Q4) + (Q5 - 1) + (5 - Q6) + (Q7 - 1) + (5 - Q8) + (Q9 - 1) + (5 - Q10)) \times 2.5$$

Setelah mengetahui skor dari masing – masing responden, langkah selanjutnya adalah mencari rata – rata skor dengan cara menjumlahkan semua hasil skor SUS dibagi dengan jumlah responden seperti yang tertulis pada *Equation 1* dibawah ini:

$$\bar{x} = \frac{\sum x}{n} \quad (5.1)$$

Keterangan :

\bar{x} = skor rata – rata

$\sum x$ = jumlah skor SUS

n = jumlah responden

Hasil rekapitulasi skor dengan metode SUS pada 15 responden dapat dilihat pada tabel 5.15 di bawah ini:

Tabel 5.15 Hasil rekapitulasi SUS

| Skor Hasil Hitung (Data Contoh) | | | | | | | | | | Jumlah | Nilai (Jumlah x 2.5) |
|---------------------------------|----|----|----|----|----|----|----|----|-----|--------|-------------------------|
| Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 | | |
| 3 | 4 | 3 | 4 | 3 | 4 | 4 | 4 | 4 | 0 | 33 | 83 |
| 3 | 4 | 3 | 4 | 4 | 3 | 4 | 4 | 3 | 0 | 32 | 80 |
| 4 | 4 | 4 | 3 | 4 | 3 | 4 | 3 | 4 | 0 | 33 | 83 |
| 3 | 3 | 3 | 1 | 3 | 1 | 3 | 3 | 3 | 0 | 23 | 58 |
| 3 | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 3 | 0 | 28 | 70 |
| 3 | 3 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 0 | 25 | 63 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 | 36 | 90 |
| 3 | 0 | 3 | 1 | 4 | 1 | 4 | 0 | 3 | 0 | 20 | 50 |
| 3 | 3 | 4 | 3 | 3 | 1 | 3 | 3 | 0 | 0 | 24 | 60 |
| 3 | 4 | 3 | 3 | 4 | 3 | 4 | 4 | 3 | 0 | 31 | 78 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 | 36 | 90 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 | 36 | 90 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 | 37 | 93 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 | 36 | 90 |
| 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 0 | 28 | 70 |
| Skor Rata-rata (Hasil Akhir) | | | | | | | | | | | 76 |

5.4 Analisis Hasil Pengujian

5.4.1 Analisis Pengujian *Pre-processing*

Analisis hasil pengujian *pre-processing* menggunakan empat model arsitektur yaitu MobileNet, ResNet50 V2, ResNet101 V2 dan ResNet152. Hasil pengujian pada ke empat arsitektur nantinya akan dilanjutkan analisis pengujian ekstraksi fitur. Analisis pengujian pertama menggunakan arsitektur MobileNet V1 yang telah diuji dengan membandingkan jenis *pre-processing* secara augmentasi data dan mendapatkan nilai *accuracy test* mencapai 65,67% pada *pre-processing* jenis *shear*. Hasil analisis menunjukknn bahwa input citra dengan masukan *pre-processing* jenis *shear* dapat bekerja dengan baik pada model MobileNetV1. Hasil *accuracy test* tersebut menggunakan *shear* dengan kombinasi *hyperparameter* jenis *optimizer* RMSprop, *learning rate* 0,0001, *batch size* 64, dan *target size* 300.

Analisis pengujian kedua dilakukan pada arsitektur ResNet152 dengan membandingkan jenis augmentasi. Adapun jenis augmentasi yang digunakan sama pada pengujian sebelumnya. Pengujian *pre-processing* pada arsitektur ResNet152 menggunakan *hyperparameter* yang sama seperti yang dilakukan pada pengujian *pre-processing* MobileNet V1. Hasil dari pengujian kedua pada jenis *pre-processing* menggunakan ResNet152 didapatkan pada jenis *shear* dengan nilai akurasi tes sebesar 88%. Selain pengujian pada jenis augmentasi data, penelitian ini juga melakukan pengujian pada *target size*. Hasil pengujian pada *target size* menghasilkan nilai akurasi yang sama pada masing-masing model arsitektur yaitu pada MobileNet bernilai 65,67% sedangkan ResNet152 V2 bernilai 88%. Berikut adalah rekapan hasil pengujian paling baik yang telah dilakukan pada pengujian jenis augmentasi dan target size yang dapat dilihat pada tabel 5.16:

Tabel 5.16 Rekap hasil pengujian *pre-processing*

| Arsitektur | Jenis <i>pre-processing</i> | <i>Target size</i> | <i>Optimizer</i> | <i>Learning rate</i> | <i>Test Accuracy</i> |
|--------------|-----------------------------|--------------------|------------------|----------------------|----------------------|
| MobileNet V1 | <i>Shear</i> | 300×300 | RMSprop | 0,0001 | 65,75% |

| | | | | | |
|-------------------------|--------------|------------------|---------|--------|--------|
| ResNet152 V2 | <i>Shear</i> | 300×300 | RMSprop | 0,0001 | 88% |
| ResNet101 V2 | <i>Shear</i> | 300×300 | RMSprop | 0,0001 | 86% |
| ResNet50 V2 | <i>Shear</i> | 300×300 | RMSprop | 0,0001 | 81,67% |

Tabel 5.16 menunjukkan hasil performa yang dihasilkan dari penerapan jenis augmentasi dan *target size* pada model sistem yang dirancang. Jenis *pre-processing* secara augmentasi yang paling signifikan adalah *Shear*. *Target size* yang optimal untuk digunakan selanjutnya berada pada nilai 300. Sehingga dapat disimpulkan bahwa *Shear* dan *target size* 300 dalam hasil pengujian ini merupakan hasil pengujian yang paling baik terhadap peningkatan performa model. Jenis *pre-processing* dan *target size* optimal pada arsitektur ResNet152 V2. Dapat diartikan bahwa pengujian pada *pre-processing* sudah bisa dilanjutkan ketahap pengujian selanjutnya.

5.4.2 Analisis Pengujian *Feature Extraction*

5.4.2.1 Pengujian Parameter '*Include Top*'

Pada pengujian ini, perubahan nilai parameter '*include top*' memberikan pengaruh yang sama terhadap arsitektur ResNet152 V2 dan MobileNet V1. Pada saat bernilai *True*, terjadi penurunan performa yang cukup signifikan secara keseluruhan. Penyebab penurunan performa ini bisa disebabkan karena model berfokus untuk mengklasifikasikan citra batik berdasarkan kategori yang ada dalam dataset imagenet yang digunakan sebagai bobot *pre-trained* (*weights*). Pada dataset imagenet tidak terdapat gambar yang mewakili motif batik yang digunakan sehingga performa model menjadi rendah. Pada saat menggunakan nilai *False*, model akan menghasilkan performa yang baik karena klasifikasi yang dijalankan hanya berfokus pada dataset citra batik yang digunakan. Pada skenario dengan nilai *False*, bobot *pre-trained* imagenet tetap digunakan, namun hanya untuk membantu model dalam memahami fitur-fitur dari dataset imagenet untuk diterapkan dalam klasifikasi citra batik. Dari perbandingan hasil pengujian ini, dapat disimpulkan

bahwa nilai *false* pada parameter '*include top*' mampu menghasilkan performa yang paling baik.

5.4.2.2 Pengujian Parameter '*Pooling Layer*'

Pada pengujian ini, perubahan pengaturan *pooling layer* memberikan pengaruh yang cukup besar pada arsitektur ResNet152 V2 dan MobileNet V1. Penggunaan *pooling layer* pada kedua arsitektur ini terlihat memberikan peningkatan yang signifikan dibandingkan ketika tanpa menggunakan *pooling layer*. Seperti yang terlihat pada hasil pengujian sebelumnya di tabel 5.6 dan 5.8, terjadi penurunan nilai *train loss* dan *validation loss* yang signifikan saat diterapkan operasi *pooling layer*. Peningkatan ini disebabkan karena *pooling layer* dapat mengurangi dimensi dari *feature map* atau bisa juga disebut *downsampling*. *Downsampling* ini membantu mengurangi jumlah parameter dari *tensor input* sehingga dapat mengurangi *overfitting* [14]. Selain itu, perubahan nilai *pool size* juga memberikan pengaruh pada akurasi dan *loss* yang dihasilkan walaupun tidak signifikan. Dari seluruh operasi *pooling* yang diuji, operasi *global average pooling* mampu menghasilkan performa yang paling baik dengan nilai test accuracy mencapai 85%. Berikut ini adalah rekapan hasil pengujian *feature extraction* pada arsitektur MobileNet V1 dan ResNet152 V2 dengan performa terbaik:

Tabel 5.17 Rekapan Hasil Pengujian *Extraction Feature*

| Arsitektur | Nilai ' <i>include top</i> ' | <i>Pooling Layer</i> | <i>Test Accuracy</i> |
|--------------|------------------------------|-------------------------------|----------------------|
| MobileNet V1 | <i>False</i> | <i>Global Average Pooling</i> | 59% |
| ResNet152 V2 | <i>False</i> | <i>Global Average Pooling</i> | 85% |

Tabel 5.17 menunjukkan perbandingan performa antara arsitektur MobileNet V1 dengan ResNet152 V2. Pada tabel tersebut telah diterapkan nilai *false* pada parameter '*include top*' dan operasi *global average pooling*. Dari perbandingan performa kedua arsitektur yang diuji, dapat disimpulkan bahwa arsitektur ResNet152 V2 lebih baik dibandingkan dengan MobileNet V1. Dengan ini pengujian selanjutnya akan dijalankan hanya dengan menggunakan arsitektur

ResNet152 V2 dengan nilai *false* pada parameter '*include top*' dan operasi *global average pooling*

5.4.3 Analisis Pengujian *Classification*

Setelah melewati pengujian pada tahapan *pre-processing* dan *feature extraction*, telah ditetapkan bahwa arsitektur dengan akurasi paling optimal yaitu menggunakan ResNet152 V2. Analisis pertama pada tahapan *classification* adalah pengujian *optimizer* dengan empat jenis antara RMSprop, Adam, Adamax, dan Nadam. Pada penelitian sebelumnya, hasil pengujian menggunakan RMSprop mendapatkan performa terbaik dibandingkan *optimizer* lainnya [39]. Penelitian tersebut terbukti sesuai dengan hasil pengujian yang telah dilakukan. *Optimizer* RMSprop mampu menghasilkan performa terbaik dibandingkan *optimizer* lainnya dengan tingkat test *accuracy* sebesar 89,67%. Walaupun performa terbaik dihasilkan oleh RMSprop, tetapi selisih antara hasil pengujian *optimizer* lainnya tidak signifikan. Hal ini menandakan bahwa *optimizer* lainnya menunjukkan hasil yang hampir sebanding dan mampu mencapai hasil yang baik.

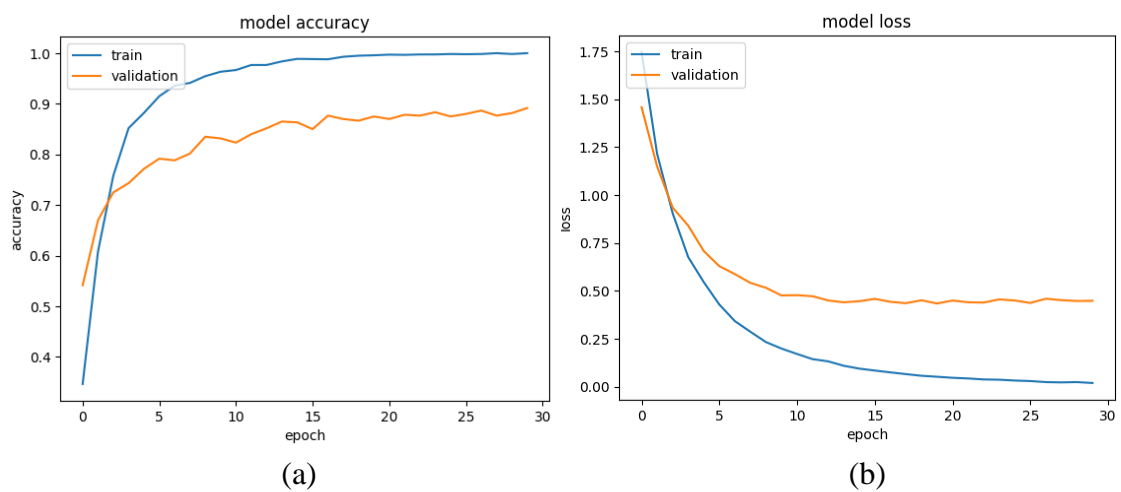
Analisis kedua pada tahapan *classification* adalah menguji nilai *learning rate* 0.01, 0.001, 0.0001, dan 0.00001. Pada pengujian *learning rate* didapatkan hasil paling optimal ialah menggunakan nilai 0.0001. Pengaturan nilai pada *hyperparameter* ini mencapai tingkat akurasi tertinggi sebesar 89,67% pada hasil *test accuracy*. Pengujian ini dapat berkaitan dengan penelitian sebelumnya yang menyatakan bahwa model menggunakan *optimizer* RMSprop dengan *learning rate* bernilai 0.0001 mendapatkan akurasi paling optimal [40]. Pada pengujian *learning rate* ini juga dapat dilihat bahwa performa hasil dari nilai 0.001 menunjukkan perbedaan yang cukup signifikan yaitu sebesar 86,33% pada *test accuracy*. Dengan melihat hasil dari *learning rate* 0.0001, pelatihan model cenderung lebih stabil dan konsisten antara *accuracy* dan *loss* yang didapatkan sehingga meminimalisir terjadinya *overfitting*. Namun, hal ini juga memberikan pengaruh antara *learning rate* bernilai 0.001 dan 0.0001 yaitu dari segi waktu. Semakin kecil nilai *learning rate*, maka proses komputasi akan lebih lambat saat proses pembelajaran model [41].

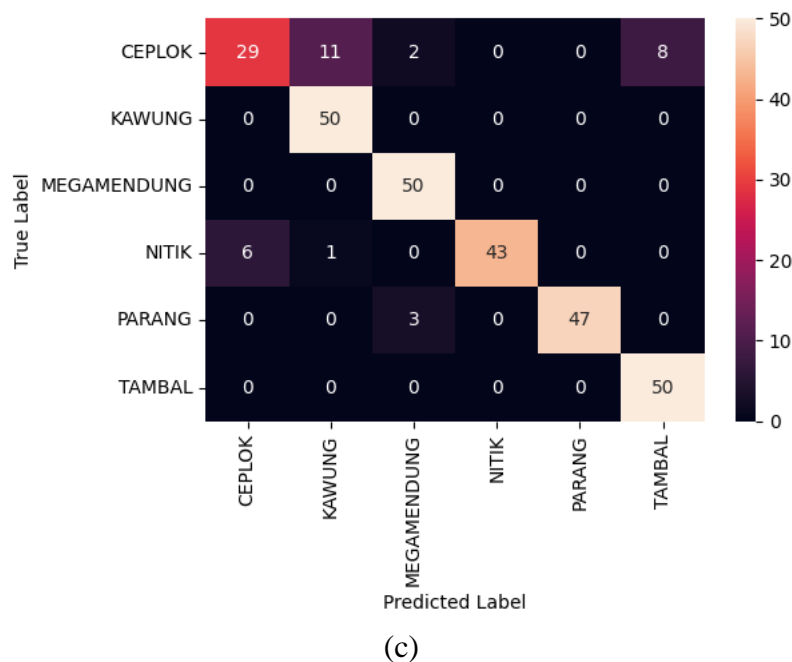
Analisis terakhir untuk tahapan *classification* adalah pengujian pada empat variasi *batch size* dengan nilai 16, 32, 64, dan 128. Pada penelitian ini, hasil performa nilai *test accuracy* tertinggi sebesar 89,67% didapatkan oleh *batch size* 64. Namun, pengujian pada performansi akurasi *batch size* 32 juga mendapatkan hasil yang tidak jauh berbeda yaitu sebesar 89,33% pada *test accuracy*. Kondisi ini juga didapatkan pada penelitian sebelumnya bahwa nilai *batch size* 64 memberikan hasil yang optimal walaupun pengujian tersebut tidak terlalu berbeda antara akurasi *batch size* lainnya [42]. Meskipun hasil akurasi antara *batch size* 64 dan 32 tidak terlalu signifikan, namun *batch size* 64 memiliki efisiensi pelatihan lebih unggul dikarenakan pemrosesan data dalam satu iterasi lebih banyak. Berikut merupakan rekapan hasil pengujian paling optimal untuk tahapan *classification* yang dapat dilihat pada tabel 5.18:

Tabel 5.18 Rekapan hasil pengujian *classification*

| Arsitektur | Optimizer | Learning Rate | Batch Size | Test Accuracy |
|---------------------|------------------|----------------------|-------------------|----------------------|
| ResNet152 V2 | RMSprop | 0.0001 | 64 | 89,67% |

Dari tabel 5.18 dapat disimpulkan bahwa hasil rekapan pengujian *classification* pada arsitektur ResNet152 V2 terbukti optimal untuk mengklasifikasikan jenis motif batik pada sistem yang dirancang. Selanjutnya, akan ditunjukkan visualisasi hasil grafik dan *confusion matrix* pada gambar 5.5 untuk memberikan ilustrasi lebih jelas mengenai kinerja model:





Gambar 5.5 (a)Grafik visualisasi *accuracy* (b)Grafik visualisasi *loss* (c)Hasil *Confusion matrix*

Pada gambar 5.5, hasil pada grafik *accuracy* dan *loss* menunjukkan performa yang cukup stabil. Hal ini dapat dilihat dari garis grafiknya yang tidak fluktuatif dan menunjukkan bahwa pembelajaran model cenderung tidak mengalami perubahan yang besar. Selanjutnya, pada gambar 5.5 terlihat hasil *confusion matrix* dari model yang dibuat dalam mengklasifikasikan enam jenis motif batik. *Confusion matrix* ini dibuat berdasarkan hasil testing. Secara keseluruhan, performa model cukup baik dalam memprediksi jenis motif batik, namun model masih melakukan banyak kesalahan dalam mengenali motif batik Ceplok. Hal ini dapat disebabkan karena motif pada batik Ceplok memiliki banyak kemiripan dengan jenis motif batik lainnya dan mayoritas telah dimodifikasi. Pada motif batik Nitik dan batik Parang, masih terdapat kesalahan walaupun tidak sebanyak motif batik Ceplok. Dari model yang dibuat terdapat hasil performansi berupa nilai *accuracy*, *precision*, *recall*, dan *F1-Score* yang dapat dilihat pada tabel 5.19:

Tabel 5.19 Hasil performansi Jenis batik

| Jenis Motif Batik | <i>Precision</i> | <i>Recall</i> | <i>F1-Score</i> |
|--------------------------|-------------------------|----------------------|------------------------|
| Ceplok | 0,83 | 0,58 | 0,68 |
| Kawung | 0,81 | 1,00 | 0,89 |
| Megamendung | 0,91 | 1,00 | 0,95 |
| Nitik | 1,00 | 0,86 | 0,92 |
| Parang | 1,00 | 0,94 | 0,97 |
| Tambal | 0,86 | 1,00 | 0,93 |
| <i>Accuracy</i> | | | 0,90 |

Dari tabel 5.19 di atas, dapat diketahui bahwa setiap jenis motif batik memiliki nilai *precision* yang baik. Pada jenis motif batik Ceplok, diperoleh nilai *recall* dan *F1-Score* yang rendah. Hal ini berkaitan dengan hasil *confusion matrix* pada motif batik Ceplok yang terdapat banyak kesalahan, sehingga mempengaruhi hasil perhitungan. Secara keseluruhan, *accuracy* yang didapatkan oleh sudah tergolong baik sebesar 0,90.

5.4.4 Analisis Pengujian Halaman Informasi *Mobile Application*

Dari hasil pengujian halaman informasi, terlihat bahwa prediksi dari enam jenis batik dimulai dari test pertama hingga test kelima menunjukkan kenaikan dari akurasi. Terlihat juga pada test pertama hingga ketiga masih terdapat kesalahan dalam menebak jenis batik. Hal ini disebabkan karna kemiripan dari segi motif dan segi kemiringan yang dihasilkan oleh hasil augmentasi jenis *shear*. Pada test keempat dan kelima menunjukkan kenaikan untuk akurasi prediksi pada enam jenis batik. Dari pengujian ini, dapat disimpulkan bahwa model ResNet152 V2 dapat menebak enam jenis batik dengan tepat. Pada test keempat dan kelima menunjukkan kenaikan untuk akurasi prediksi pada enam jenis batik meskipun masih ada jenis batik yang susah untuk ditebak sehingga menunjukkan kesalahan dalam menebak jenis batik. Kondisi ini disebabkan oleh posisi kain yang diuji, pola dari batik yang diuji serta jarak kamera dan kain batik yang membutuhkan jarak cukup jauh agar dapat terlihat keseluruhan motif batik. Dari pengujian ini, dapat disimpulkan bahwa model Resnet152 V2 dapat menebak enam jenis batik dengan tepat jika posisi kain dan posisi kamera diletakkan dalam posisi yang tepat.

5.4.5 Analisis Pengujian Keseluruhan Fungsi dari *Mobile Application*

Dari hasil rekapitulasi skor dengan metode SUS, didapatkan hasil akhir 76. Sesuai dengan gambar, terlihat bahwa aplikasi BatiQu menempati level *acceptable* pada sisi *Acceptability* dan Grade C. Sedangkan dari sisi *Adjective Ratings*, hasil rekapitulasi aplikasi BatiQu berada di posisi “GOOD”. Hal ini menunjukkan bahwa aplikasi BatiQu dapat diterima oleh masyarakat dan dapat bekerja dengan baik untuk dapat mengidentifikasi jenis batik.

5.5 Kesimpulan dan Ringkasan CD-5

5.5.1 Kesimpulan

Berdasarkan hasil pengujian penelitian pada tugas akhir klasifikasi jenis batik menggunakan *Deep Learning* berbasis Aplikasi, didapatkan kesimpulan bahwa hasil jenis augmentasi *shear* dan target size 300 yang digunakan pada model arsitektur ResNet152 V2 memiliki hasil akurasi yang lebih optimal dibandingkan hasil citra jenis augmentasi *shear* dan *target size* 300 pada model arsitektur MobileNet V1. Untuk hasil pengujian dari algoritma *Convolution Neural Network* model arsitektur ResNet152 V2 dapat bekerja dengan baik dan optimal untuk melakukan klasifikasi citra jenis batik dengan hasil nilai akurasi mencapai 85% dibandingkan dengan metode MobileNet V1 yang akurasinya lebih rendah. Akurasi tersebut dicapai dengan parameter ‘*include top*’ yang bernilai *false* dan menggunakan operasi *global average pooling*. Pada hasil pengujian *classification* untuk model arsitektur ResNet152 V2 dipilih menjadi model sistem yang paling optimal untuk dapat dilanjutkan pada *Mobile Application* dikarenakan hasil dari *accuracy test* mencapai 89,67%. Pada jenis *optimizer* RMSprop dengan *learning rate* 0,0001 dan *batch size* 64 yang digunakan di bagian *classification* pada model ResNet152 V2 menghasilkan hasil yang lebih tinggi. Terakhir pada bagian hasil aplikasi menunjukan bahwa performa kinerja aplikasi dapat meningkat jika posisi kain dan kamera tepat pada posisinya.

5.5.2 Saran

Berdasarkan hasil dari penelitian ini masih ada beberapa kekurangan yang perlu untuk ditingkatkan dan diperbaiki pada penelitian serta data selanjutnya.

Adapun saran yang diberikan yaitu penelitian selanjutnya diharapkan menggunakan dataset yang lebih banyak dengan resolusi gambar yang lebih baik terutama batik yang kurang dikenal oleh masyarakat agar model sistem dapat belajar lebih variatif. Menggunakan teknik *pre-processing* yang lain agar akurasi sistem dapat lebih variatif. Untuk bagian *extraction feature* peneliti selanjutnya diharapkan menggunakan model arsitektur yang lain seperti Xception, VGG, RegNet, *EfficientNet*, atau model arsitektur terbaru pada metode *Deep Learning* dengan konfigurasi layer yang bervariasi. Penambahan pada pengujian selanjutnya diharapkan menggunakan *optimizer* yang lebih variatif untuk dibandingkan agar sistem dapat meningkatkan hasil akurasi model. Pengembangan pada aplikasi yang terakhir, diharapkan untuk peneliti selanjutnya menambahkan fitur maps sebagai asal muasal batik, dan terintegrasi ke *E-commerce* dan artikel terkait.

DAFTAR PUSTAKA

- [1] F. Robi, R. Magdalena, and I. Wijayanto, "Rancang Bangun Aplikasi Deteksi Motif Batik Berbasis Pengolahan Citra Digital pada Platform Android," *e-Proceeding Eng.*, vol. 1, no. 1, pp. 310–318, 2014.
- [2] R. R. Imelda, "Pengenalan Motif Batik Pandeglang Menggunakan Deteksi Tepi Canny dan Metode K-NN Berbasis Android," *J. Teknol. Inf.*, vol. 16, no. 2, p. 83, 2021, doi: 10.35842/jtir.v16i2.404.
- [3] H. Ibda, "Strategi Memutus Mata Rantai Pembajakan Hak Cipta pada Seni Batik Nusantara," *Citra Ilmu*, vol. 17, no. 33, pp. 65–78, 2021.
- [4] A. A. Kasim and A. Harjoko, "Klasifikasi Citra Batik Menggunakan Jaringan Syaraf Tiruan Berdasarkan Gray Level Co-Occurrence Matrices (GLCM)," *Semin. Nas. Apl. Teknol. Inf.*, pp. 7–13, 2014.
- [5] S. Bin, H. Amir, S. Nur, A. Fitriani, and A. M. Anwar, "Deteksi Citra X-Ray Paru-Paru Terinfeksi COVID-19 dengan Algoritma CNN Berbasis Aplikasi Web," *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 17, no. 1, 2022, doi: 10.30872/jim.v17i1.6534.
- [6] E. Prasetyo, R. Purbaningtyas, R. D. Adityo, E. T. Prabowo, and A. I. Ferdiansyah, "Perbandingan Convolution Neural Network untuk Klasifikasi Kesegaran Ikan Bandeng pada Citra Mata," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 3, p. 601, 2021, doi: 10.25126/jtiik.2021834369.
- [7] H. M. Balaha, M. Saif, A. Tamer, and E. H. Abdelhay, "Hybrid Deep Learning and Genetic Algorithms Approach (HMB-DLGAHA) for The Early Ultrasound Diagnoses of Breast Cancer," *Neural Comput. Appl.*, vol. 34, no. 11, pp. 8671–8695, 2022, doi: 10.1007/s00521-021-06851-5.
- [8] Kartini Parmono, "Nilai Kearifan Lokal dalam Batik Tradisional Kawung," *Fak. Filsafat Univ. Gadjah Mada Yogyakarta*, vol. 1, no. 1, 2016.
- [9] N. E. Trimargawati, "Penerapan Hukum Hak Cipta Seni Batik Pekalongan sebagai Komoditas Internasional (Studi Upaya Pemerintah Kota Pekalongan

- Menjadikan Batik Pekalongan sebagai Komoditas Internasional),” *Law Reform*, vol. 5, no. 1, pp. 1–30, 2010, doi: 10.14710/lr.v5i1.669.
- [10] D. A. Widyastutiningrum, “Perlindungan Hukum terhadap Hak Cipta Batik Motif Ceplok Segoro Amarto di Kota Yogyakarta,” *JIPRO J. Intellect. Prop.*, vol. 2, no. 1, pp. 36–51, 2019, doi: 10.20885/jipro.vol2.iss1.art4.
 - [11] A. Ahmad Hania, “Mengenal Artificial Intelligence, Machine Learning, dan Deep Learning,” *J. Teknol. Indones.*, vol. 1, no. 1, pp. 1–6, 2017.
 - [12] M. S. Yamin Lubis, “Implementasi Artificial Intelligence pada System Manufaktur Terpadu,” *Semin. Nas. Tek. UISU*, vol. 4, no. 1, pp. 1–7, 2021.
 - [13] A. Hermawan, L. Lianata, Junaedi, and A. R. K. Maranto, “Implementasi Machine Learning sebagai Pengenal Nominal Uang Rupiah dengan Metode YOLOv3,” *SATIN - Sains dan Teknol. Inf.*, vol. 8, no. 1, pp. 12–22, 2022, doi: 10.33372/stn.v8i1.816.
 - [14] A. Kholik, “Klasifikasi Menggunakan Convolutional Neural Network (CNN) pada Tangkapan Layar Halaman Instagram,” *J. Data Min. dan Sist. Inf.*, vol. 2, no. 2, p. 10, 2021, doi: 10.33365/jdmsi.v2i2.1345.
 - [15] U. Khultsum, F. Sarasati, and G. Taufik, “Penerapan Metode Mobile-Net untuk Klasifikasi Citra Penyakit Kanker Paru-Paru,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 5, p. 1366, 2022, doi: 10.30865/jurikom.v9i5.4918.
 - [16] M. Suyuti, “Pengembangan Model Klasifikasi Mata Tertutup dan Terbuka dalam Identifikasi Kelelahan Menggunakan Arsitektur Mobile CNN,” 2023.
 - [17] K. H. Mahmud, S. Al Faraby, and Adiwijaya, “Klasifikasi Citra Multi-Kelas Menggunakan Convolutional Neural Network,” *e-Proceeding Eng.*, vol. 6, no. 1, pp. 2127–2136, 2019.
 - [18] S. A. Syifa and I. A. Dewi, “Arsitektur Resnet-152 dengan Perbandingan Optimizer Adam dan RMSProp untuk Mendeteksi Penyakit Paru-Paru,” *MIND (Multimedia Artif. Intell. Netw. Database) J.*, vol. 7, no. 2, pp. 139–150, 2022, doi: 10.26760/mindjournal.v7i2.139-150.

- [19] D. A. P. Oktavia, S. Rizal, and N. K. C. Pratiwi, "Klasifikasi Gejala Defisiensi Nutrisi pada Tanaman Padi Menggunakan CNN dengan Arsitektur Resnet-50," *e-Proceeding Eng.*, vol. 8, no. 6, pp. 3171–3175, 2022.
- [20] M. Ather, I. Hussain, B. Khan, Z. Wang, and S. Ding, "Woven Fabric Pattern Recognition and Classification Based on Deep Convolutional Neural Networks," *MDPI*, pp. 1–12, 2020.
- [21] N. D. Miranda, L. Novamizanti, and S. Rizal, "Convolutional Neural Network pada Klasifikasi Sidik Jari Menggunakan Resnet-50," *J. Tek. Inform.*, vol. 1, no. 2, pp. 61–68, 2020, doi: 10.20884/1.jutif.2020.1.2.18.
- [22] G. W. Intyanto, "Klasifikasi Citra Bunga dengan Menggunakan Deep Learning: CNN (Convolution Neural Network)," *J. Arus Elektro Indones.*, vol. 7, no. 3, p. 80, 2021, doi: 10.19184/jaei.v7i3.28141.
- [23] M. Afif, A. Fawwaz, K. N. Ramadhani, and F. Sthevanie, "Klasifikasi Ras pada Kucing menggunakan Algoritma Convolutional Neural Network(CNN)," *J. Tugas Akhir Fak. Inform.*, vol. 8, no. 1, pp. 715–730, 2020.
- [24] D. Hidayat and W. F. Al Maki, "Deteksi Glaukoma Menggunakan Metode Convolutional Neural Network Dan Grabcut Segmentation," *e-Proceeding Eng.*, vol. 9, no. 3, pp. 1855–1861, 2022.
- [25] U. Nur Oktaviana, R. Hendrawan, A. Dwi Khoirul Annas, and G. Wasis Wicaksono, "Klasifikasi Penyakit Padi berdasarkan Citra Daun Menggunakan Model Terlatih Resnet101," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1216–1222, 2021, doi: 10.29207/resti.v5i6.3607.
- [26] R. B. Anshori, "Klasifikasi Citra Kanker Serviks Menggunakan Arsitektur Deep Residual Network," 2022.
- [27] S. Septhyan, R. Magdalena, and N. K. Caecar Pratiwi, "Deep Learning untuk

- Deteksi Covid-19 , Pneumonia , dan Tuberculosis pada Citra Rontgen Dada Menggunakan CNN dengan Arsitektur Alexnet,” vol. 8, no. 6, pp. 2869–2878, 2022.
- [28] A. Lupita Dyayu, Beny, and H. Yani, “Evaluasi Usability Aplikasi PeduliLindungi Menggunakan Metode Usability Testing dan System Usability Scale (SUS),” *J. Manaj. Teknol. Dan Sist. Inf.*, vol. 3, no. 1, pp. 395–404, 2023, doi: 10.33998/jms.2023.3.1.720.
- [29] W. M. Pradnya D and A. P. Kusumaningtyas, “Analisis Pengaruh Data Augmentasi pada Klasifikasi Bumbu Dapur Menggunakan Convolutional Neural Network,” *J. Media Inform. Budidarma*, vol. 6, no. 4, p. 2022, 2022, doi: 10.30865/mib.v6i4.4201.
- [30] U. Mawaddah, H. Armanto, and E. Setyati, “Prediksi Karakteristik Personal Menggunakan Analisis Tanda Tangan dengan Menggunakan Metode Convolutional Neural Network (CNN),” *Antivirus J. Ilm. Tek. Inform.*, vol. 15, no. 1, pp. 123–133, 2021, doi: 10.35457/antivirus.v15i1.1526.
- [31] I. W. Suartika E. P, A. Y. Wijaya, and R. Soelaiman, “Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101,” *J. Tek. ITS*, vol. 5, no. 1, p. 76, 2016.
- [32] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, “Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle,” *J. Algoritma*, vol. 1, no. 1, pp. 45–56, 2020, doi: 10.35957/algoritme.v1i1.434.
- [33] B. Anthony and Yohannes, “Verifikasi Kinship dengan Arsitektur ResNet50,” *2ND MDP Student Conf. 2023*, pp. 265–273, 2023.
- [34] S. Hikmat Haji and A. Mohsin Abdulazeez, “Comparison of Optimization Techniques Based on Gradient Descent Algorithm: A Review,” *J. Archaeol. Egypt/Egyptology*, vol. 18, no. 4, pp. 2715–2743, 2021.
- [35] R. Guha, “Benchmarking Gradient Based Optimizers’ Sensitivity to

- Learning Rate,” *SSRN Electron. J.*, no. January, pp. 1–33, 2023, doi: 10.2139/ssrn.4318767.
- [36] N. Rochmawati, H. B. Hidayati, Y. Yamasari, H. P. A. Tjahyaningtjas, W. Yustanti, and A. Prihanto, “Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam,” *J. Inf. Eng. Educ. Technol.*, vol. 5, no. 2, pp. 44–48, 2021, doi: 10.26740/jieet.v5n2.p44-48.
- [37] A. Bangor, P. Kortum, and J. Miller, “Determining What Individual SUS Scores Mean : Adding an Adjective Rating Scale,” *J. Usability Stud.*, vol. 4, no. 3, pp. 114–123, 2009.
- [38] I. N. T. A. Putra, I. G. I. Sudipa, N. M. S. D. Sukerthi, and N. P. Y. Yunia, “Analisis User Experience pada Layanan Telekomunikasi Operator Seluler Menggunakan Metode System Usability Scale (SUS),” *Digit. Transform. Technol.*, vol. 3, no. 1, pp. 49–57, 2023.
- [39] N. K. Caecar Pratiwi, N. Ibrahim, Y. Nur Fu’adah, and S. Rizal, “Deteksi Parasit Plasmodium pada Citra Mikroskopis Hapusan Darah dengan Metode Deep Learning,” *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 9, no. 2, pp. 306–317, 2021, doi: 10.26760/elkomika.v9i2.306.
- [40] N. Azahro Choirunisa, T. Karlita, and R. Asmara, “Deteksi Ras Kucing Menggunakan Compound Model Scaling Convolutional Neural Network,” *Technomedia J.*, vol. 6, no. 2, pp. 236–251, 2021, doi: 10.33050/tmj.v6i2.1704.
- [41] A. Thohari and G. B. Hertantyo, “Implementasi Convolutional Neural Network untuk Klasifikasi Pembalap MotoGP Berbasis GPU,” *Proc. Conf. Electr. Eng. Telemat. Ind. Technol. Creat. Media*, pp. 50–55, 2018.
- [42] Y. Nur Fu’adah, I. Dawan Ubaidullah, N. Ibrahim, F. Frahma Taliningsing, N. Khofiya S Y, and M. A. Pramuditho, “Optimasi Convolutional Neural Network dan K-Fold Cross Validation pada Sistem Klasifikasi Glaukoma,”

ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron., vol. 10,
no. 3, pp. 728–741, 2022, doi: 10.26760/elkomika.v10i3.728.

LAMPIRAN CD-1

Curriculum Vitae 1

PERSONAL INFORMATION

Full Name : Aulia Chusnyriani Sani Zulkarnaen
Gender : Female
Birth Place/Date : Sragen, 7th December 2000
Nationality : Indonesia
Religion : Islam
Phone Number : 0857-0779-7467
Email : auliariiiani7@gamil.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor of Telecommunication Engineering
Semester : 8th

EDUCATION

| Institutions | City and Province | Year |
|---------------------------|--------------------|-----------------------|
| SMA Budi Utomo Jombang | Jombang, East Java | July 2016 – July 2019 |
| Universitas Telkom | Bandung, West Java | August 2019 - present |

PERSONAL ACHIEVEMENTS

| Awards | Year | Description |
|--------|------|-------------|
| - | - | - |
| - | - | - |

SUPPORTING ACTIVITIES AND TRAININGS

| Activities and Trainings | Period | Place |
|---|--------------|-------------------|
| Course Data Science Fundamental Program Kredensial Mikro Mahasiswa Indonesia (KMMI) | October 2021 | Online |
| Optimize Content Marketing Google ADS | April 2021 | Online |
| Training DQLab- Python for Data Professional Beginner | August 2021 | Online |
| Python Certification in Cisco | October 2020 | Telkom University |

ORGANIZATIONAL EXPERIENCE

| Organizations | Title | Period | Descriptions |
|---|----------------|-----------|--|
| BEM Telkom University | Member | 2020-2021 | Coordinating student activity units running at Telkom by holding activities and leadership |
| Biospin Laboratory Assistant | Member and HRD | 2021 | conduct workshops and recruit new members |
| System Communication Laboratory Assistant | HRD | 2022 | assisting the assistant coordinator and recruiting interns |

WORKING EXPERIENCE

| Work | Year | Description |
|---------------------------------------|------|--|
| Internship at PT SOLUSI247 Yogyakarta | 2022 | Work at Renewal Technology Development (especially on deep |

| | | |
|---|------|--|
| | | learning section) |
| Practicum Assistant in Laboratorium Dasar Siskom Laboratory | 2022 | Assistant of Communication System |
| Research Assistant in Laboratorium BioSpin | 2021 | Assistant of Biomedical Engineering |

SKILLS AND HOBBIES

Language Skills : Indonesian (Native), English (Advanced)

Computer Skills : Python, R, Microsoft Office, Digital Design

Hobbies and interests : Leadership, Analytical Thinking, Teamwork.

Others : Interested in doing field project or research, Interested in
learning something new about technology

Curriculum Vitae 2

PERSONAL INFORMATION

Full Name : I Gusti Ngurah Rejski Ariantaraputra
Gender : Male
Birth Place/Date: Denpasar, 12th August 2001
Nationality : Indonesia
Religion : Hindu
Phone Number : 0813-5302-0531
Email : gustiari2001@gmail.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor of Telecommunication Engineering
Semester : 8th

EDUCATION

| Institutions | City and Province | Year |
|--------------------|--------------------|-----------------------|
| SMAN 1 Sukawati | Gianyar, Bali | July 2016 – July 2019 |
| Universitas Telkom | Bandung, West Java | August 2019 - present |

PERSONAL ACHIEVEMENTS

| Awards | Year | Description |
|--------|------|-------------|
| - | - | - |
| - | - | - |

SUPPORTING ACTIVITIES AND TRAININGS

| Activities and Trainings | Period | Place |
|-------------------------------|-------------------------------|-------------------|
| Python Certification in Cisco | October 2020 | Bandung |
| Studi Independent with Huawei | September 2021 - January 2022 | Telkom University |

ORGANIZATIONAL EXPERIENCE

| Organizations | Title | Period | Descriptions |
|-----------------------------------|-----------------------------|-------------|---|
| UKM Kesenian Bali Widyacana Murti | Member of Logistic Division | 2020-2021 | Organized logistics needs |
| Senior Residents VIII | Member | 2020 - 2021 | Organized dormitory activities in Telkom University |
| Senior Residents IX | Member | 2021-2022 | Organized dormitory activities in Telkom University |

WORKING EXPERIENCE

| Work | Year | Description |
|---------------------------|------|--|
| Internship at Telkom Ubud | 2022 | Doing several works of maintenance and data processing about networks facilities |

SKILLS AND HOBBIES

Language Skills : Indonesian (Native), English (Advanced)

Computer Skills : C, Python, Java, Microsoft Office

Hobbies and interests : Learning about gadgets, leadership .

Curriculum Vitae 3

PERSONAL INFORMATION

Full Name : Nada Fauzia Reviana
Gender : Female
Birth Place and Date : Bekasi, 5th October 2001
Nationality : Indonesia
Religion : Islam
Phone Number : 0859-2151-8735
Email : nadafauzia18@gmail.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor of Telecommunication Engineering
Semester : 8th

EDUCATION

| Institutions | City and Province | Year |
|------------------------------------|-------------------------|-----------------------|
| 6 th Senior High School | South Tangerang, Banten | July 2016 – May 2019 |
| Universitas Telkom | Bandung, West Java | August 2019 - present |

PERSONAL ACHIEVEMENTS

| Awards | Year | Description |
|--------|------|-------------|
| - | - | - |

SUPPORTING ACTIVITIES AND TRAININGS

| Activities and Trainings | Period | Place |
|--------------------------|--------|-------|
| - | - | - |

ORGANIZATIONAL EXPERIENCE

| Organizations | Title | Period | Descriptions |
|---|--|----------------|--|
| ROAR 2020 | Event Member | 2019 - 2020 | Assisting the event coordinator to arrange the events rundowns. |
| ROAR 2021 | Event Coordinator | 2020 - 2021 | Organizing the whole event from the preparation until the closure of the event. |
| RANGER Telkom University (Tangerang Community in Telkom University) | Member of Human Resources Development Department | 2020 - 2021 | Develop the interests and talents of all Tangerang Entities studying at Telkom University. |
| ASTERISK 2021 | Member of Event and Execution Committee | 2021 - 2021 | Creating themes and concepts for the regeneration of Telecommunication Engineering students event. |
| Representative Council Member of HMTT Telco Tel-U | Council Staff of Aspiration Commission II | 2021 - 2022 | Accommodating the aspirations of Telecommunication Engineering students. |
| RANGER Telkom University | Head of Event Department | 2021 - 2022 | Successfully leading and organizing the members of event department in doing their job. |
| Representative Council Member of HMTT Telco Tel-U | Head of Aspiration Commission II | 2022 - Present | Directing the council staff in carrying out the work program of the aspiration commission. |

WORKING EXPERIENCE

| Work | Year | Description |
|---|------|---|
| Internship at ICON+ West Java Region | 2022 | Doing an analysis of the monitoring application developed by the company and understanding the flow and shortcomings of the application |

SKILLS AND HOBBIES

| | |
|----------------------|---|
| Soft Skills | : Analytical Thinking, Teamwork, Problem Solving, Communication, Socializing, Creative. |
| Hard Skills | : Digital Design, Expert in using Microsoft Excel and Microsoft Word. |
| Programming Language | : Python, Java, C++ |

Curriculum Vitae 4

PERSONAL INFORMATION

Full Name : Rahmawati Hidayah
Gender : Female
Birth Place and Date : Tulungagung, 25th July 2000
Nationality : Indonesia
Religion : Islam
Phone Number : 0812-9570-5528
Email : rahmawatihidayah18@gmail.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor of Telecommunication Engineering
Semester : 8th

EDUCATION

| Institutions | City and Province | Year |
|--------------------|--------------------|-----------------------|
| SMK Telkom Malang | Malang, East Java | July 2016 – July 2019 |
| Universitas Telkom | Bandung, West Java | August 2019 - present |

PERSONAL ACHIEVEMENTS

| Awards | Year | Description |
|--------|------|-------------|
| - | - | - |

SUPPORTING ACTIVITIES AND TRAININGS

| Activities and Trainings | Period | Place |
|--|----------------|--------|
| Short course “Dasar – Dasar Dukungan Teknis” from Coursera | September 2022 | Online |

ORGANIZATIONAL EXPERIENCE

| Organizations | Title | Period | Descriptions |
|---|-------------------------|--------|---|
| LINK-K Community | Member | 2016 | Involve to “Try Out” event that held from Telkom University for High School Student |
| BEM Telkom University | Member | 2020 | Making content for Instagram’s feed that contain money management and Entrepreneur |
| MobileComm Laboratory Assistant | Secretary and Treasurer | 2021 | Doing administration with money and regulation at laboratory |
| System Communication Laboratory Assistant | Secretary | 2022 | Doing administration with regulation |

WORKING EXPERIENCE

| Work | Year | Description |
|---|------|---|
| Internship at GAP Investment Corporation. | 2013 | Doing several works of maintenance and data processing about networks and BSCs. |
| Lab Assistant in Laboratorium DasarTeknik Elektro | 2013 | Assistant of Digital System Lab Work and Digital Signal Processing Lab Work. |

SKILLS AND HOBBIES

| | |
|--------------------------------|---|
| Language Skills (Advanced), | : Indonesian (Native), Javanese (Native) English |
| Computer Skills | : Python, Java, Microsoft Office |
| Hobbies and interests | : Sightseeing, Learning about newest Technology, Learning new language |
| Others | : Interested in learning something new about technology |

LAMPIRAN CD-2 DAN 3

Lampirkan dokumen pendukung dibuat berdasarkan dari rujukan dokumen yang berisikan tentang biaya pembuatan produk.

1. Kain potongan Mega mendung :
https://shopee.ee/8UZLYLhH9M?share_channel_code=1
2. Kain potongan Parang :
https://shopee.ee/3VAfbJeSvq?share_channel_code=1
3. Kain potongan Lereng, Sidomukti, Kawung :
https://shopee.ee/1fj1QHllHE?share_channel_code=1
4. Kain potongan Nitik : https://shopee.co.id/Kain-batik-cap-sogan-Jogja-motif-nitik-i.312216339.3955857566?sp_atk=6a06d5d0-3a54-4939-a9aa-1040c09c16b6&xptdk=6a06d5d0-3a54-4939-a9aa-1040c09c16b6
5. Kain potongan Ceplok : https://shopee.co.id/KAIN-BATIK-H.SANTOSO-GADING-MAS-i.231483665.6822325594?sp_atk=cac869ac-0a2a-4c25-9607-fd5984e97dfe&xptdk=cac869ac-0a2a-4c25-9607-fd5984e97dfe
6. Jasa Pembuatan Box : range harga dari 150.000 hingga 300.000 sudah disertai dengan pemasangan LED, PCB, Baterai, Kabel).
7. Jasa Konsultasi Aplikasi : range harga dari 200.000 hingga 500.000

LAMPIRAN CD-4 DAN 5

Tabel 5.20 Tabel biaya penelitian Tugas Akhir

| No. | Keperluan | Jumlah | Biaya | Total |
|---------|--------------------------------|--------|-------------|-------------|
| 1 | Google Colab Premium | 1 | Rp 170.0000 | Rp 170.0000 |
| 2 | Box Kayu | 1 | Rp 450.000 | Rp 450.000 |
| 3 | LED 12V 6 Mata | 6 | Rp 5.000 | Rp 30.000 |
| 4 | Kabel tembaga kecil merah+hita | 1 | Rp 20.000 | Rp 20.000 |
| 5 | Saklar plastik kecil | 1 | Rp 3.000 | Rp 3.000 |
| Total : | | | | Rp 673.000 |

Tabel 5.13 adalah tabel dari perincian biaya yang telah dikeluarkan untuk perancangan penelitian Tugas Akhir.