**BUKU TUGAS AKHIR**

**CAPSTONE DESIGN**



**KLASIFIKASI JENIS BATIK**

**MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

**Oleh :**

**Aulia Chusnyriani Sani Zulkarnaen / 1101194043**

**I Gusti Ngurah Rejski A. P / 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 MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE 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**

**Bandung, ..... 2023**

**Menyetujui,**

|  |  |
| --- | --- |
| Pembimbing 1 | Pembimbing 2 |
|  |  |
| Nur Ibrahim S.T., M.T. | Nor Kumalasari Caecar  Pratiwi S.T., M. T. |
| NIP. 14870047 | NIP. 20890017 |

LEMBAR PENGESAHAN

BUKU CAPSTONE DESIGN

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

**Telah disetujui dan disahkan sebagai bagian dari Capstone Design**

**Program S1 Teknik Telekomunikasi**

**Fakultas Teknik Elektro**

**Universitas Telkom**

**Bandung**

Disusun oleh:

**I Gusti Ngurah Rejski Ariantara Putra**

**1101190017**

**Bandung, ..... 2023**

**Menyetujui,**

|  |  |
| --- | --- |
| Pembimbing 1 | Pembimbing 2 |
|  |  |
| Nur Ibrahim S.T., M.T. | Nor Kumalasari Caecar  Pratiwi S.T., M. T. |
| NIP. 14870047 | NIP. 20890017 |

LEMBAR PENGESAHAN

BUKU CAPSTONE DESIGN

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

**Telah disetujui dan disahkan sebagai bagian dari Capstone Design**

**Program S1 Teknik Telekomunikasi**

**Fakultas Teknik Elektro**

**Universitas Telkom**

**Bandung**

Disusun oleh:

**Nada Fauzia Reviana**

**1101194198**

**Bandung, ..... 2023**

**Menyetujui,**

|  |  |
| --- | --- |
| Pembimbing 1 | Pembimbing 2 |
|  |  |
| Nur Ibrahim S.T., M.T. | R Yunendah Nur Fuadah, S.T, M.T. |
| NIP. 14870047 | NIP. 17900087 |

LEMBAR PENGESAHAN

BUKU CAPSTONE DESIGN

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

**Telah disetujui dan disahkan sebagai bagian dari Capstone Design**

**Program S1 Teknik Telekomunikasi**

**Fakultas Teknik Elektro**

**Universitas Telkom**

**Bandung**

Disusun oleh:

**Rahmawati Hidayah**

**1101194070**

**Bandung, ..... 2023**

**Menyetujui,**

|  |  |
| --- | --- |
| Pembimbing 1 | Pembimbing 2 |
|  |  |
| Nur Ibrahim S.T., M.T. | R Yunendah Nur Fuadah, S.T, M.T. |
| NIP. 14870047 | NIP. 17900087 |

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Aulia Chusnyriani Sani Zulkarnaen

NIM : 1101194043

Alamat : Jl.Sukanirus no.124, ds. Citeureup, Dayeuhkolot,

Kab. Bandung, Jawa Barat 40257

No. Telepon : 085707797467

Email : Auliariiani7@gmail.com

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

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

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

|  |
| --- |
| Bandung,.... 2023 |
|  |
| Aulia Chusnyriani Sani Zulkarnaen |
| 1101194043 |

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : I Gusti Ngurah Rejski Ariantara Putra

NIM : 1101190017

Alamat : Jalan Mangga II, RT.1/RW.2, Desa Sukapura, Kecamatan

Dayeuhkolot, Kabupaten Bandung, Jawa Barat

No. Telepon : 081353020531

Email : gustiari2001@gmail.com

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

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

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

|  |
| --- |
| Bandung, ... Juni 2023 |
|  |
| I Gusti Ngurah Rejski Ariantara P. |
| 1101190017 |

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Nada Fauzia Reviana

NIM : 1101194198

Alamat : Kost Arga Putri, Jalan Sukabirus RT/RW 03/15 No. 40,

Kel. Citeureup, Kec. Dayeuhkolot, Kab. Bandung,

Jawa Barat.

No. Telepon : 085921518735

Email : nadafauzia18@gmail.com

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

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

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

|  |
| --- |
| Bandung, ... Juni 2023 |
|  |
| Nada Fauzia Reviana |
| 1101194198 |

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Rahmawati Hidayah

NIM : 1101194070

Alamat : Jl. PGA No.2, Lengkong, Kec. Bojongsoang, Kabupaten Bandung, Jawa Barat 40287

No. Telepon : 081295705528

Email : rahmawatihidayah18@gmail.com

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

**KLASIFIKASI JENIS BATIK MENGGUNAKAN MACHINE LEARNING BERBASIS APLIKASI**

*BATIK TYPE CLASSIFICATION USING MACHINE LEARNING BASED APPLICATION*

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

A person wearing a black head scarf

Description automatically generated with low confidence

|  |
| --- |
| Bandung,... Juni 2023 |
| Rahmawati Hidayah |
| 1101194070 |

ABSTRAK

Format kertas, margin, penomoran halaman, penomoran Bab dan sub Bab sudah diatur secara format. Untuk mengubah level tulisan (dari Bab ke sub-Bab, ke paragraph atau sebaliknya) silahkan menggunakan tab styles pada tab Home di Microsoft word. Jika mengubah penomoran dan mengganti dengan yang manual, akan berakibat ke template.

Tulisan Heading tanpa penomoran menggunakan style “Style 1”. Penulisan Bab menggunakan style “Heading 1”, penulisan sub bab menggunakan “Heading 2” dan penulisan sub-sub-bab menggunakan “Heading 3”. Penulisan paragraph menggunakan style “Normal”. Tidak perlu mengubah spacing dan ukuran tulisan lagi.

Abstrak terdiri dari 3 paragraf : pragraf pertama menceritakan latar belakang ditambah 1 kalimat yang menceritakan masalah utama penelitian di akhir paragraph. Paragraf ke-2 merupakan solusi yang ditawarkan untuk mengatasi masalah yang ada pada paragraph 1, dan paragraph ketiga merupakan data kuantitatif maupun kualitatif dari hasil penelitian dengan ditambah kesimpulan singkat. Kata kunci di pisahkan dengan tanda koma, maksimal 5 kata kunci

Kata kunci : Abstrak, Penulisan, Style, Selingkung

ABSTRACT

Abstrak dalam Bahasa inggris. Jangan lupa tambahkan keyword

KATA PENGANTAR

Ditulis di halaman baru maksimal 1 halaman. (untuk pindah halaman ke halaman baru bisa langsung menggunakan ctrl+Enter

UCAPAN TERIMAKASIH

Ditulis di halaman baru maksimal 2 halaman

DAFTAR ISI

Daftar isi dibuat otomatis menggunakan tab “Reference-Table of Contents”. Update daftar isi tinggal menggunakan tombol “Update Table” pada daftar isi. (Nantinya tulisan ini dihapus saja).

[LEMBAR PENGESAHAN i](#_Toc130977705)

[LEMBAR PERNYATAAN ORISINALITAS ii](#_Toc130977706)

[ABSTRAK iii](#_Toc130977707)

[ABSTRACT iv](#_Toc130977708)

[KATA PENGANTAR v](#_Toc130977709)

[UCAPAN TERIMAKASIH vi](#_Toc130977710)

[DAFTAR ISI vii](#_Toc130977711)

[DAFTAR GAMBAR x](#_Toc130977712)

[DAFTAR TABEL x](#_Toc130977713)

[DAFTAR SINGKATAN x](#_Toc130977714)

[BAB 1 USULAN GAGASAN 1](#_Toc130977715)

[1.1 Latar Belakang Masalah 1](#_Toc130977716)

[1.2 Informasi Pendukung Masalah 1](#_Toc130977717)

[1.3 Analisis Umum 1](#_Toc130977718)

[1.3.1 Aspek Ekonomi 1](#_Toc130977719)

[1.3.2 Aspek Manufakturabilitas 1](#_Toc130977720)

[1.3.3 Aspek Lain (lengkapi sendiri) 1](#_Toc130977721)

[1.4 Kebutuhan yang Harus Dipenuhi 2](#_Toc130977722)

[1.4.1 Cara Memasukkan Tabel 2](#_Toc130977723)

[1.4.2 Cara memasukkan gambar 2](#_Toc130977724)

[1.4.3 Cara memasukkan sitasi 3](#_Toc130977725)

[1.5 Solusi Sistem yang Diusulkan 3](#_Toc130977726)

[1.5.1 Karakteristik Produk 3](#_Toc130977727)

[1.5.2 Skenario Penggunaan 4](#_Toc130977728)

[1.6 Kesimpulan dan Ringkasan CD-1 4](#_Toc130977729)

[BAB 2 DESAIN KONSEP SOLUSI 5](#_Toc130977730)

[2.1 Spesifikasi Produk 5](#_Toc130977731)

[2.2 Verifikasi 5](#_Toc130977732)

[2.2.1 Verifikasi Spesifikasi 1 5](#_Toc130977733)

[2.2.2 Verifikasi spesifikasi 2 5](#_Toc130977734)

[2.3 Kesimpulan dan Ringkasan CD-2 5](#_Toc130977735)

[BAB 3 DESAIN RANCANGAN SOLUSI 6](#_Toc130977736)

[3.1 Konsep Sistem 6](#_Toc130977737)

[3.1.1 Pilihan Sistem 6](#_Toc130977738)

[3.1.2 Analisis 6](#_Toc130977739)

[3.1.3 Sistem yang akan Dikembangkan 6](#_Toc130977740)

[3.2 Rencana Desain Sistem 6](#_Toc130977741)

[3.3 Pengujian Komponen (Kalibrasi) 6](#_Toc130977742)

[3.4 Jadwal Pengerjaan 6](#_Toc130977743)

[3.5 Kesimpulan dan Ringkasan CD-3 7](#_Toc130977744)

[BAB 4 IMPLEMENTASI 8](#_Toc130977745)

[4.1 Implementasi Sistem 8](#_Toc130977746)

[4.1.1 Sub-sistem 1 (boleh diganti dengan nama subsistemnya) 8](#_Toc130977747)

[4.1.2 Sub-sistem 2 8](#_Toc130977748)

[4.2 Analisis Pengerjaan Implementasi Sistem 8](#_Toc130977749)

[4.3 Hasil Akhir Sistem 8](#_Toc130977750)

[4.4 Kesimpulan dan Ringkasan CD-4 8](#_Toc130977751)

[BAB 5 PENGUJIAN SISTEM 9](#_Toc130977752)

[5.1 Skema Pengujian Sistem 9](#_Toc130977753)

[5.2 Proses Pengujian 9](#_Toc130977754)

[5.2.1 Proses Pengujian 1 9](#_Toc130977755)

[5.2.2 Proses Pengujian 2 9](#_Toc130977756)

[5.3 Analisis Hasil Pengujian 9](#_Toc130977757)

[5.3.1 Analisis Hasil Pengujian 1 9](#_Toc130977758)

[5.3.2 Analisis Hasil Pengujian 2 9](#_Toc130977759)

[5.4 Kesimpulan dan Ringkasan CD-5 9](#_Toc130977760)

[DAFTAR PUSTAKA 10](#_Toc130977761)

[LAMPIRAN CD-1 12](#_Toc130977762)

[LAMPIRAN CD-2 13](#_Toc130977763)

[LAMPIRAN CD-3 14](#_Toc130977764)

[LAMPIRAN CD-4 15](#_Toc130977765)

[LAMPIRAN CD-5 16](#_Toc130977766)

DAFTAR GAMBAR

[**Gambar 1.1 Tukang sayur berjualan** 2](#_Toc127802731)

DAFTAR TABEL

[**Tabel 1.1 Data mahasiswa contoh** 2](#_Toc127887321)

[**Tabel 2.1 Spesifikasi Produk** 5](#_Toc127887322)

DAFTAR SINGKATAN

FTE : Fakultas Teknik Elektro

S1TT : S1 Teknik Telekomunikasi

TULT : Telkom University Landmark Tower

Ini hanya contoh. Lebih baik jika berurtan berdasarkan abjad. Boleh ditambahkan daftar-daftar yang lain seperti DAFTAR SIMBOL dan daftar-daftar lain yang dianggap perlu (opsional).

Menambahkan daftar table dan daftar gambar secara otomatis, dapat menggunakan tab “References” lalu “Insert table of figures” dibagian “caption” dengan memilih jenis caption. Begitu juga daftar gambar.

# USULAN GAGASAN

## 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 UNESCO (*United Nations Educational, Scientific, and Cultural Organization*) sebagai hak kebudayaan intelektual bangsa Indonesia pada tanggal 2 Oktober 2009 [2]. Dengan diakuinya 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 batik telah dilakukan oleh beberapa peneliti terdahulu menggunakan tahapan *pre-processing* ekstraksi fitur *Geometric Moment Invariant* berbasis klasifikasi *k-Nearest Neighbor* (k-NN) dan nilai akurasi mencapai 80% lebih tinggi dibandingkan dengan tahapan *pre-processing* menggunakan ekstraksi fitur *Co-occurrence Matrix* [5]. Selanjutnya, terdapat penelitian menggunakan klasifikasi *Convolutional Neural Network* (CNN) yang memiliki tahapan *pre-processing* pada data *training* dan data *testing* dengan *optimizer* yang digunakan adam dan *sigmoid* sebagai *activator* serta fungsi *binary* *cross entropy* untuk mengurangi data loss yang nilai akurasinya mencapai 91,41% [6]. Selain itu, penelitian lain didapati melakukan tahapan *pre-processing* ekstraksi fitur warna maupun tekstur yang masukannya berupa tujuh fitur ekstraksi GLCM (*energy, homogeneity, contrast*) dan rata-rata nilai RGB (*red, green, blue*) [5]. Metode klasifikasi yang digunakan adalah *k-Nearest Neighbor* (k-NN) dan tercatat menghasilkan tingkat akurasi sebesar 91,25% [5].

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, dan Batik Megamendung dengan jumlah total 660 sampel 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. Dengan adanya teknologi ini dapat berkontribusi untuk melestarikan kebudayaan batik yang sangat beragam.

## 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*) (KWRI UNESCO, 2017). Seperti yang dilansir pada situs resmi UNESCO, Batik dianggap sebagai simbol dan budaya yang selalu menyertai kehidupan masyarakat Indonesia dari sejak lahir sampai meninggal dunia (UNESCO, 2022). Pemerintah juga ikut serta dalam menjaga kelestarian batik dengan mengeluarkan produk hukum. Berdasarkan yang ditulis pada situs resmi APPBI (Asosiasi Perajin dan Pengusaha Batik Indonesia) (APPBI, 2021), ada banyak produk hukum yang telah dikeluarkan oleh Pemerintah Indonesia yang berkaitan dengan batik diantaranya:

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

## 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:

### **Aspek Ekonomi**

Batik itu 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 dan 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. 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.

### **Aspek Edukatif**

Jenis batik di Indonesia berjumlah 5.489 corak yang tersebar dari Aceh ke Papua. Dari 5.489 corak yang berbeda tentu memiliki filosofi yang berbeda dari tiap corak batik yang ada. Dengan jenis batik sebanyak itu serta dengan filosofi yang berbeda pula di tiap corak, tentu tidak bisa menghafal secara rinci sebuah jenis batik hanya dengan sekali melihat corak batik tersebut. Maka dari itu penelitian ini membuat alat ini agar mempermudah masyarakat untuk mengetahui jenis batik dari melihat sebuah corak batik yang ada.

Dengan alat ini juga bisa mengedukasi tentang sejarah lengkap dari sebuah jenis batik yang terdeteksi kamera. Sehingga dapat mengetahui kapan batik ini terbentuk dan mulai dikenalkan ke masyarakat luas hingga dapat menjadi batik khas dari sebuah daerah. Dalam alat ini juga dijelaskan mengenai filosofi dari corak yang ada di dalam batik itu sebagai ilmu tambahan agar mengerti arti dari setiap corak batik yang ada.

## Kebutuhan yang Harus Dipenuhi

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

### **Perangkat Lunak**

1. Pada *Software Desktop*.
2. Sistem Operasi *Microsoft Windows* 11.
3. *Google Collab*.
4. *Android Studio Electric Flamingo* | 2022.2.1.
5. Metode CNN (*Convolutional Neural Network*).
6. Dataset.
7. Pada Smartphone

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

### **Perangkat Keras**

1. Pada Desktop
2. Laptop *Lenovo Legion* Y7000
3. *Processor* : *Intel Core* i7-9750H
4. Kartu Grafis : NVIDIA *GeForce* GTX 1650
5. RAM : 16 GB DDR4
6. *Storage* : 512 GB SSD M.2 NVME + 1 TB HDD
7. Pada Smartphone dan Box
8. Infinix Zero 5G
9. *Processor* : *MediaTek Dimensity* 900 MT6877
10. RAM : 8 GB
11. Memori internal : 128GB
12. Memori eksternal : *microSDXC* (*dedicated slot*)
13. Ukuran Box : 12 x 12 x 19 cm

## Solusi Sistem yang Diusulkan

Berdasarkan latar belakang masalah yang ada, terdapat dua pilihan solusi sistem yang dapat diterapkan diantaranya menggunakan Metode *Convolutional Neural Network* (CNN), Metode *K-Nearest Neighbor* (KNN), Metode *Gray Level Co-occurrence Matrix* (GLCM). Dari kedua solusi sistem ini terdapat beberapa fitur yang perlu dibandingkan untuk mengetahui solusi sistem yang optimal.

### **Karakteristik Produk**

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

1. Fitur Utama :

Solusi sistem pertama yang akan dirancang menggunakan metode *Convolutional Neural Network* (CNN). Metode CNN sendiri merupakan pengembangan dari *multilayer perceptron* (MLP) yang diolah menggunakan data dua dimensi untuk membentuk citra [10]. Dalam metode CNN terdapat hidden layer berupa konvolusi, ReLU, dan *Pooling*. Metode CNN memiliki kelebihan karena dapat melakukan ekstraksi fitur tersendiri seperti *texture* dan *color*. Namun, dibalik kelebihannya terdapat celah perbaikan dalam metode CNN yaitu hanya mampu digunakan pada data dua dimensi saja karena bersifat konvolusi [11].

1. Fitur Dasar :

* Pengambilan gambar dan mengunggah gambar dari galeri dengan waktu yang singkat serta mudah dalam melakukan klasifikasi gambar.
* Mudah untuk digunakan serta diakses.

1. Fitur Tambahan :

* Box untuk mendapatkan gambar citra dari motif batik.
* Dapat terhubung dengan aplikasi android untuk mengklasifikasikan motif batik setelah dilakukannya pengambilan gambar.

1. Sifat Solusi yang Diharapkan :

* Mudah untuk diakses karena hanya cukup meng-install dan dapat digunakan oleh berbagai kalangan dalam mengidentifikasikan motif batik.
* Mudah untuk digunakan, karena cukup melakukan pengunduhan aplikasi melalui smartphone kemudian memotret batik sehingga nanti dapat mengeluarkan hasil.
* Tidak membutuhkan perawatan yang intensif dan mengefisiensikan waktu dalam pengambilan gambar.

#### **Metode *K-Nearest Neighbor* (KNN) dengan Ekstraksi Fitur *Gray Level Co-occurrence Matrix* (GLCM)**

1. Fitur Utama :

Solusi sistem kedua yang akan dirancang menggunakan metode *K-Nearest Neighbor* (KNN). KNN merupakan algoritma yang bekerja dengan cara mencari K tetangga terdekat dengan menghitung jarak antara data melalui data training [13]. Algoritma KNN memiliki kelebihan jika dibandingkan dengan algoritma lain, yaitu lebih mudah dalam diimplementasikan, dapat digunakan untuk klasifikasi data dengan jumlah kelas yang beragam, dan dapat mengklasifikasikan data dalam jumlah besar dengan hasil yang baik [14]. Namun, algoritma KNN juga memiliki kekurangan yaitu sensitif terhadap jumlah tetangga dan jarak yang digunakan dalam perhitungan. Untuk ekstraksi fitur yang digunakan yaitu dengan Metode *Gray Level Co-occurrence Matrix* (GLCM). *Gray-level cooccurrence matrix* adalah sebuah metode ekstraksi fitur yang digunakan untuk mendapatkan fitur *texture*/pola pada suatu citra digital [15]. Kelebihan dari metode ini adalah mudah diaplikasikan untuk analisis secara tekstur dan mudah dibedakan karena mengubah warna *Red Green Blue* (RGB) menjadi *grayscale* atau keabuan. Namun kekurangan dari metode GLCM adalah input dari citra harus berupa citra grayscale, yang memiliki kelemahan yaitu komponen warna dari citra diabaikan [16].

1. Fitur Dasar :

* Pengambilan gambar dan mengunggah gambar dari galeri dengan waktu yang singkat serta mudah dalam melakukan klasifikasi gambar.
* Mudah untuk digunakan serta diakses.

1. Fitur Tambahan :

* Box untuk mendapatkan gambar citra dari motif batik.
* Dapat terhubung dengan aplikasi *android* untuk mengklasifikasikan motif batik setelah dilakukannya pengambilan gambar.

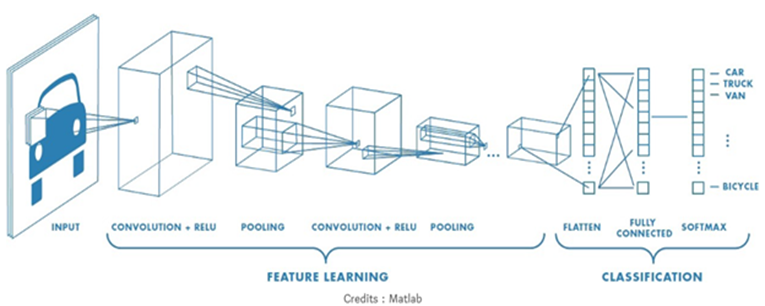
1. Sifat Solusi yang Diharapkan :

* Mudah untuk diakses karena hanya cukup meng-*install* dan dapat digunakan oleh berbagai kalangan dalam mengidentifikasikan motif batik.
* Mudah untuk digunakan, karena cukup melakukan pengunduhan aplikasi melalui *smartphone* kemudian memotret batik sehingga nanti dapat mengeluarkan hasil.
* Tidak membutuhkan perawatan yang intensif dan mengefisiensikan waktu dalam pengambilan gambar.

### **Skenario Penggunaan**

Dalam skenario penggunaan menjelaskan mengenai pilihan sistem yang akan digunakan dalam pembuatan aplikasi. Aplikasi ini dirancang agar memudahkan bagi kalangan umum untuk mengklasifikasikan beberapa motif batik beserta dengan penjelasan singkat. Aplikasi ini dapat menghasilkan identifikasi motif batik, contohnya batik ini merupakan Batik Parang, atau Batik Tambal, Batik Ceplok, Batik Kawung, Batik Nitik, dan Batik Megamendung.

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



**Gambar 1.1** Metode Convolutional Neural Network

Pada gambar 1.1 menjelaskan tentang tahapan dalam pembangunan arsitektur untuk klasifikasi motif batik:

* 1. *Convolutional layer* :

Pada tahapan layer ini masuk kedalam 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.

* 1. *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.

* 1. *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.

* 1. Output layer :

Tahap terakhir ini merupakan hasil akhir dari pengolahan lapisan-laspian sebelumnya. Dimana hasil dari lapisan ini sebagai prediksi masalah inputan layer.

#### **Metode *K-Nearest Neighbor* (KNN) dengan Ekstraksi Fitur *Gray Level Co-occurrence Matrix* (GLCM)**

Berikut merupakan tahapan dalam pembangunan arsitektur untuk klasifikasi motif batik:

* 1. Persiapan Data :

Tahapan yang pertama adalah dengan mempersiapkan data karena ketika kita menggunakan ekstraksi fitur GLCM, citra harus melalui proses reduksi dimensi yaitu pengubahan dimensi citra dari RGB menjadi citra *Grayscale* [17]. Selain reduksi dimensi, citra juga perlu melalui tahapan *pre-processing* yang lain seperti *rescale* untuk membuat keseluruhan dataset menjadi rapi dan terformat dengan baik.

* 1. Menentukan nilai K :

Tahapan berikutnya adalah menentukan jumlah tetangga terdekat atau yang disimbolkan dengan K. Secara umum berdasarkan hasil penelitian sebelumnya, nilai K yang semakin tinggi pada proses klasifikasi akan berpengaruh terhadap pengurangan jumlah efek *noise* yang semakin besar. Namun hal ini tidak selalu memberikan hasil yang paling optimal [17].

* 1. Menghitung jarak :

Selanjutnya akan dihitung jarak antara dataset yang sudah ada dengan data baru yang akan diuji. Pada umumnya, metode perhitungan jarak yang sering digunakan adalah *Euclidean Distance*, namun selain itu ada juga metode lain seperti *Manhattan* dan *Mahalanobis Distance*.

* 1. Menentukan tetangga terdekat:

Pada tahapan ini akan ditentukan K tetangga terdekat berdasarkan jaraknya yang terdekat dengan data baru yang diuji.

* 1. Menentukan kelas:

Pada tahapan ini, kelas dari data yang diuji akan ditentukan berdasarkan kelas dari data yang menjadi K terdekat dari data yang diuji.

* 1. Evaluasi model:

Pada tahapan ini akan dilakukan evaluasi performa dari metode KNN yang digunakan dengan melihat nilai akurasi, *precision*, *recall*, atau *F1-score*.

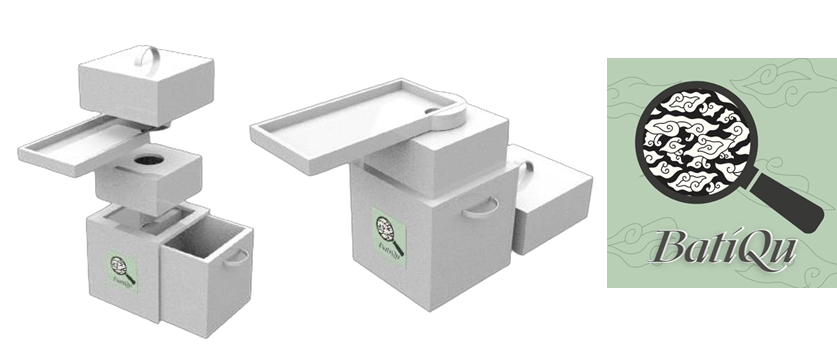
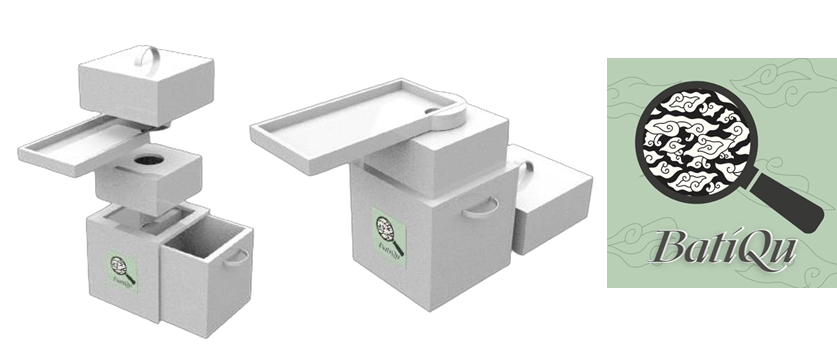
## Kesimpulan dan Ringkasan CD-1

Batik merupakan budaya Indonesia yang telah diakui dunia dan memiliki jenis motif yang beragam. Di dalam negeri, pemerintah telah ikut serta dalam menjaga kelestarian batik dengan membuat beberapa produk hukum sebagai jaminan. Dengan penelitian ini dibuatlah aplikasi mobile BatiQu yang berbasis machine learning dalam sistem klasifikasinya untuk mengklasifikasikan jenis batik yang ada dengan data yang telah disiapkan yaitu terdapat 6 motif batik. Adapun 6 batik tersebut adalah Batik Ceplok, Batik Kawung, Batik Megamendung, Batik Parang, Batik Nitik dan Batik Tambal. Selain digunakan untuk membantu klasifikasi jenis batik, aplikasi BatiQu juga bertujuan untuk membantu masyarakat agar lebih mengenal jenis batik yang beragam dengan memberikan informasi tambahan berdasarkan jenis Batik yang ada di Indonesia.

# DESAIN KONSEP SOLUSI

## Spesifikasi Produk

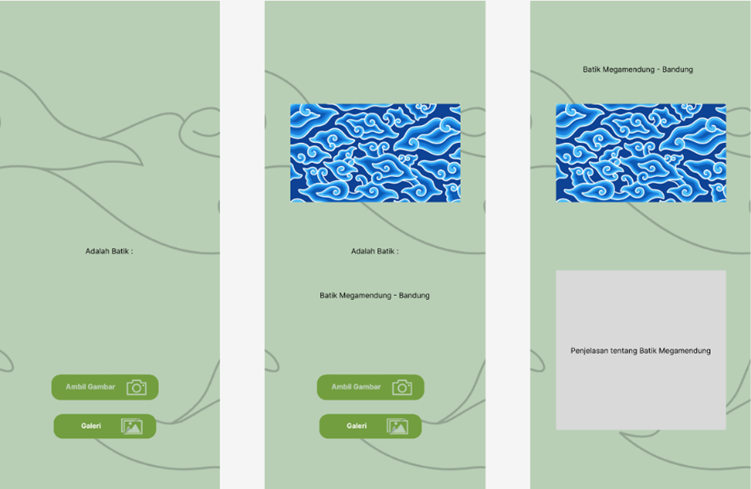
Pada Tugas Akhir ini, aplikasi BatiQu merupakan perangkat lunak yang akan digunakan untuk klasifikasi batik. Produk ini dirancang berbasis aplikasi *mobile* dengan fitur yang ramah digunakan bagi penggunanya dan dapat diunduh lewat *Google Playstore.* Pengoperasian dalam aplikasi ini cukup mudah karena hanya memerlukan empat tombol dan ditujukan untuk masyarakat dengan tujuan mempermudah masyarakat dalam mengidentifikasi jenis batik. Dari penjelasan di atas, aplikasi BatiQu memiliki fitur dimana penjelasan singkat mengenai jenis-jenis batik yang akan diuji 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 tidak berubah. Bentuk yang akan dirancang dari perangkat keras ini berupa *box.* *Box* BatiQu dirancang berbentuk kubus dengan panjang 12 cm, lebar 12 cm, dan tinggi 19 cm dengan bahan plastik ABS (*Acrylonitrile Butadiene Styrene*). 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. Tujuan pembuatan perangkat keras ini yaitu untuk menyelaraskan sudut dan pencahayaan saat proses klasifikasi batik agar keluaran yang dihasilkan sesuai parameter yang ditentukan. Gambar 2.1 merupakan *box* yang akan digunakan dalam pengambilan sample uji :

**Gambar 2.1** Box Pengujian Sampel Batik dan Logo BatiQu

Pada penjelasan dibawah ini yaitu mengenai Aplikasi BatiQu yanng memiliki dua tombol pada halaman utama yang tertera pada gambar 2.2

**Gambar 2.2** Homapage mock-up Aplikasi BatiQu

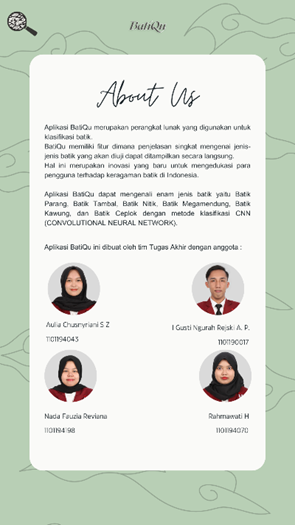
Tombol yang terlihat dari gambar diatas memiliki beragam fungsi seperti contoh pada tombol pertama memiliki fungsi untuk memulai prediksi jenis batik. Tombol terakhir atau “*About Us*” berfungsi informasi singkat mengenai aplikasi BatiQu dan informasi singkat tentang pembuat dari aplikasi ini. Dari tampilan halaman utama yang terlihat, pengguna bisa memilih antara dua tombol yang ada. 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 sample uji akan terlihat pada halaman informasi yang dibuat sesuai dengan kelas nya seperti gambar diatas. Halaman informasi berisikan informasi singkat tentang jenis batik yang terdeteksi seperti asal usul, keunikan dari pola batik dan bagaimana cara pakai dari batik tersebut.

Selanjutnya pengguna bisa tekan “*back*” tombol 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 seperti gambar 2.4 :

*****Gambar 2.4***Page About Us

Halaman “About Us” berisikan informasi singkat dari aplikasi BatiQu seperti fungsi singkat dan alasan dibuatnya aplikasi batiQu serta berisikan informasi diri dari pembuat aplikasi BatiQu. Harapan besar dengan adanya aplikasi ini, akan banyak membantu masyarakat dalam menyelesaikan beberapa identifikasi dari banyaknya jenis motif batik yang ada di Indonesia.

|  |  |  |
| --- | --- | --- |
| **No** | **Hal** | **Rincian** |
| 1 | Menu “Kamera” | Menu ini berfungsi untuk mengambil foto  kain batik yang akan diuji. |
| 2 | Menu “*Gallery*” | Menu ini berfungsi untuk memilih gambar batik dari galeri *handphone* pengguna untuk  diuji. |
| 3 | Menu “*About Us*” | 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 *Box* | Kotak ini berbahan Plastik ABS (*Acrylonitrile Butadiene Styrene)* dengan ukuran 12x12x19 cm dan memiliki lima tingkat untuk mengukur beberapa jarak  yang bervariasi. |
| 7 | Kain batik | Pengujian kain batik akan menggunakan delapan sampel batik yaitu Batik Parang, Batik Lereng, Batik Ceplok, Batik Kawung, Batik Nitik, Batik Megamendung, dan Batik Tambal dengan ukuran 15x15 cm  untuk setiap sampel jenis batiknya. |

**Gambar 2.5** Spesifikasi Produk

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.

## Verifikasi

Dengan adanya verifikasi produk dalam spesifikasi yaitu dapat meningkatkan kinerja serta efektivitas yang mampu diukur agar mengetahui apakah sistem dapat berjalan dengan baik. Verifikasi juga ditujukan untuk menjaga akurasi maupun meningkatan dengan mengurangi biaya untuk system.

### Verifikasi Spesifikasi 1

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

**Gambar 2.6** Verifikasi spesifikasi pertama

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

### Verifikasi spesifikasi 2

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

**Gambar 2.7** Verifikasi spesifikasi kedua

Tabel 2.3 mengenai verifikasi yang kedua dalam spesifikasi sistem. Sama halnya dengan verifikasi yang pertama, namun perbedaanya berada pada bagian pengujian yang mana verifikasi kedua menguji pada data loss yang akan dirancang pada sistem klasifikasi.

## 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 plastik ABS (*Acrylonitrile Butadiene Styrene*) ini diberi nama *Box* BatiQu dengan panjang 12 cm, lebar 12 cm, dan tinggi 19 cm. Spesifikasi produk ini akan diverifikasi kembali kemampuan serta keefektifannya untuk mengetahui apakah sistem berjalan dengan baik. Hal pertama adalah sistem dapat dimaksimalkan akurasinya hingga 90% saat mengklasifikasikan jenis batik. Selanjutnya, sistem klasifikasi mampu mendapatkan data loss terendah mendekati < 0,1.

# DESAIN RANCANGAN SOLUSI

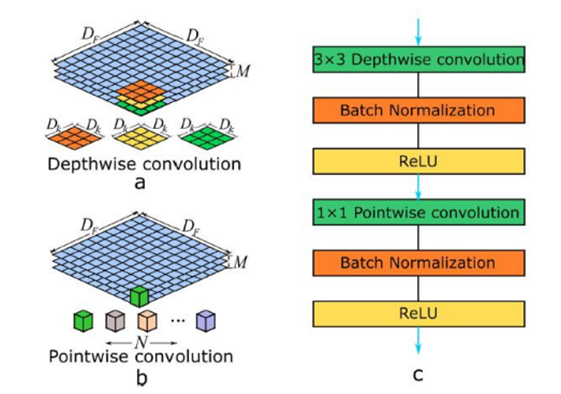
## Konsep Sistem

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.

### **Pilihan Sistem**

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

*Convolutional Neural Network* merupakan salah satu jenis arsitektur yang biasa digunakan dalam klasifikasi dan pengenalan objek/citra. Tujuan metode CNN adalah melatih model jaringan syaraf tiruan dimana hasil yang diharapkan dapat meminimalkan data lost pada hasil prediksi dengan data asli. Arsitektur CNN terdiri dari empat bagian yaitu *convolution layer, pooling layer, fully-connected layer* dan *output layer*. Salah satu arsitektur yang ada dalam *Convolutional Neural Network* adalah MobileNet. MobileNet merupakan sebuah model arsitektur yang dirancang menggunakan konsep *depthwise separable convolutions* dengan tujuan agar mudah dalam mengimplementasian kebutuhan *mobile* dan *embedded applications*.

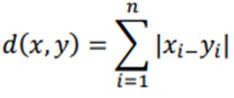
Konsep *depthwise separable convolutions* sendiri ditujukan untuk dapat mengurangi jumlah kompleksitas dari parameter dan komputasi yang digunakan saat pelatihan model. Dalam konsep ini terdapat beberapa pembagian konvolusi dengan dua tahapan, yaitu *Depthwise Convolution* dan *Pointwise Convolution*. *Depthwise Convolution* sendiri dalam perancangan diaplikasikan pada kernel atau filter dalam jumlah kecil yang masing-masing input dibuat secara terpisah. *Pointwise Convolution* 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 melalukan beberapa variasi model pada citra menjadi usulan dalam pemilihan konsep sistem. Pada gambar 3.1 merupakan arsitektur MobileNet dengan beberapa layer :

**Gambar 3.1** Arsitektur MobileNet

#### Metode *K-Nearest Neighbor* (KNN) dengan Ekstraksi Fitur *Gray Level*

#### *Co-occurrence Matrix* (GLCM)

*K-Nearest Neighbor* (KNN) adalah metode yang dapat mengklasifikasikan objek berdasarkan persamaannya dengan objek lain yang menjadi tetangga terdekatnya. Tetangga terdekat ini dilihat dari nilai k dengan syarat nilai k tidak boleh lebih besar dari jumlah data latih, harus ganjil, dan jumlahnya lebih dari satu. Jarak antara objek yang ingin diklasifikasikan dengan objek pada data latih ini akan dihitung dengan menggunakan metode *Manhattan Distance* yang dapat menghasilkan akurasi lebih baik dibandingkan dengan *Euclidean Distance* [18]. *Manhattan Distance* merupakan metode perhitungan jarak pada suatu ruang dengan menerapkan konsep selisih mutlak [19]. Persamaan *Manhattan Distance* dapat dilihat pada gambar dibawah ini

**Equation 1** Persamaan Manhattan Distance

Pada Equation 1 memiliki keterangan sebagai berikut:

d(x,y) = Jarak

x = koordinat lokasi 1

y = koordinat lokasi 2

Metode KNN yang digunakan pada opsi pilihan sistem ini akan dibantu dengan fitur ekstraksi gambar *Gray-Level Co-occurrence Matrix* (GLCM). GLCM merupakan metode untuk memperoleh nilai statistic orde kedua dengan menghitung probabilitas hubungan kedekatan antara dua buah piksel pada jarak (d) dan sudut (θ) tertentu [20]. GLCM berfungsi untuk mengubah gambar RGB menjadi gambar berskala keabuan, yang bertujuan untuk membantu algoritma KNN mendapatkan hasil akurasi yang lebih optimal.

**Gambar 3.2** Diagram Alir Pengolahan Citra

Secara garis besar, tahapan sistem klasifikasi menggunakan metode KNN dengan fitur ekstraksi GLCM digambarkan pada diagram alir yang terlihat pada gambar 3.2 sistem mengawali dengan melakukan pembacaan terhadap citra yang akan diklasifikasikan, kemudian dilanjutkan ke proses konversi citra ke format *grayscale*. Selanjutnya, akan dilakukan tahap segmentasi untuk memisahkan objek dengan *background* pada citra batik. Setelah segmentasi, citra akan masuk ke tahap ekstraksi fitur dengan GLCM lalu dideteksi pola mana yang serupa citra batik yang diuji dengan metode klasifikasi KNN. Ketika citra batik yang diuji berhasil diklasifikasi, sistem akan menampilkan informasi mengenai batik tersebut pada aplikasi BatiQu lalu bisa lanjut kembali ke menu awal apabila ingin melakukan klasifikasi lagi.

### **Analisis**

#### Kriteria Analisis

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

1. Ekonomi

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

1. Sistem

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

1. Ketersediaan komponen

Sistem yang akan pilih 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.

1. 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 sistem yang sederhana dan mudah dimengerti

#### Analisis Konsep

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

|  |  |
| --- | --- |
| Kriteria Ekonomi | |
| 1. Solusi Sistem pertama (Metode *Convolutional Neural Network* (CNN)) | Dari kriteria ekonomi, sistem yang akan dipilih adalah sistem yang memerlukan biaya produksi dan perawatan yang cukup murah sehingga dapat dengan mudah dijangkau dan digunakan oleh masyarakat umum. Biaya produksi dan perawatan yang murah juga akan memudahkan realisasi dari sistem yang akan dibuat. |
| 1. Solusi Sistem kedua (Metode *K-Nearest Neighbor* (KNN) dengan Ekstrasi Fitur *Gray Level Co-occurrence* *Matrix* (GLCM) |

(a)

|  |  |
| --- | --- |
| Kriteria Sistem | |
| 1. Solusi Sistem pertama (Metode *Convolutional Neural Network* (CNN)) | Pada kriteria sistem, sistem yang akan dipilih adalah sistem mudah dimodifikasi, sehingga sistem dapat dengan cepat diperbarui atau diperbaiki apabila ada kerusakan atau kekurangan. |
| 1. Solusi Sistem kedua (Metode *K-Nearest Neighbor* (KNN) dengan Ekstrasi Fitur *Gray Level Co-occurrence* *Matrix* (GLCM) |

(b)

|  |  |
| --- | --- |
| Kriteria Ketersediaan Komponen | |
| 1. Solusi Sistem pertama (Metode *Convolutional Neural Network* (CNN)) | Sistem yang akan pilih 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 |
| 1. Solusi Sistem kedua (Metode *K-Nearest Neighbor* (KNN) dengan Ekstrasi Fitur *Gray Level Co-occurrence* *Matrix* (GLCM) |

(c)

|  |  |
| --- | --- |
| Kriteria Penggunaan/Pengguna | |
| 1. Solusi Sistem pertama (Metode *Convolutional Neural Network* (CNN)) | 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 sistem yang sederhana dan mudah dimengerti. |
| 1. Solusi Sistem kedua (Metode *K-Nearest Neighbor* (KNN) dengan Ekstrasi Fitur *Gray Level Co-occurrence* *Matrix* (GLCM) |

(d)

**Gambar 3.3** (a)Kriteria Ekonomi (b)Kriteria Sistem (c)Kriteria ketersediaan komponen (d)Kriteria Penggunaan/Pengguna

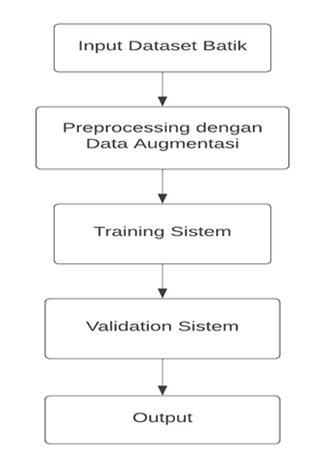
Pada tabel 3.1 berisikan konsep kriteria yang akan dibangun pada perancangan penelitian. Setiap konsep kriteria sudah dipilih berdasarkan kegunaan dari peracangan aplikasi. Dengan ada kriteria ini maka mempermudah dalam melakukan konsep perancangan.

### **Sistem yang akan Dikembangkan**

Dari kedua solusi sistem yang sudah dipaparkan, solusi yang akan dikembangkan yaitu sistem dengan metode *Convolutional Neural Network* (CNN). Sistem ini dipilih karena memiliki beberapa keunggulan dibanding solusi sistem lainnya. Contohnya sistem dengan metode CNN lebih efisien dan mudah untuk pengaplikasian ke dalam sistem yang akan dibuat. Selain itu, hasil dari pengenalan citra dalam metode CNN memiliki peluang hasil yang lebih relevan. Metode CNN sendiri dapat melakukan ekstraksi fitur tersendiri yang memudahkan proses pengklasifikasian citra batik. Sedangkan untuk sistem menggunakan metode *K-Nearest Neighbor* (KNN) dengan Ekstraksi Fitur *Gray Level Co-occurrence Matrix* (GLCM) kurang optimal untuk mengklasifikasikan data dengan jumlah besar. Kekurangan lainnya dari sistem menggunakan metode KNN dengan fitur ekstraksi GLCM *sensitive* terhadap jumlah tetangga dan jarak yang digunakan. Secara keseluruhan, perbandingan di atas memperkuat alasan untuk menggunakan solusi sistem dengan metode CNN.

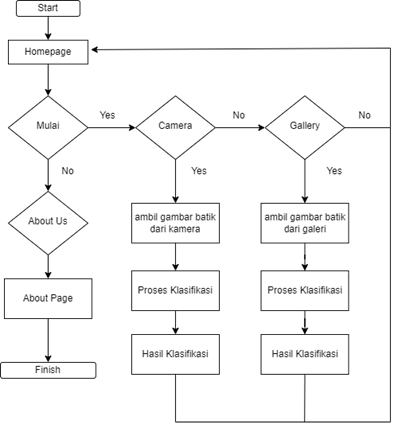
## Rencana Desain Sistem

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 pada kode program klasifikasi diawali dengan menyiapkan dataset yang didapat melalui *Kaggle* berupa citra batik yang terdiri dari enam kelas, yaitu Ceplok, Parang, Nitik, Megamendung, Kawung, dan Tambal. Kemudian setelah data terkumpul, dilakukan *pre-processing* berupa data augmentasi (meliputi *resize, crop, flip, rotate*) yang dapat meningkatkan variasi dari jumlah data yang ada. Selanjutnya data citra akan masuk ke dalam proses klasifikasi menggunakan metode CNN yang terdiri dari dua tahapan yaitu *training* dan *validation.* Perbedaan antara training dengan validation terletak pada tujuannya dimana *training* berfungsi untuk melatih sistem dalam mengenali kelas dari *dataset* yang digunakan dengan jumlah *dataset* yang banyak, Sedangkan *validation* berfungsi untuk menguji kemampuan sistem dalam mengklasifikasikan kelas dari *dataset* dengan jumlah data yang lebih sedikit dibandingkan saat training.

**Gambar 3.4** Flowchart Program CNN

Pada gambar 3.3 menjelaskan mengenai flowchart dari program CNN, dimana terlihat pada tahap awal di 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 selanjtunya ada pengambilan gambar menggunakan kamera handphone lalu setelah itu akan di proses klasifikasi menggunakan machine 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 lalu setelah itu akan di proses klasifikasi menggunakan machine learning. Selanjutnya, akan menampilkan 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 3.5** Flowchart Aplikasi BatiQu

## Pengujian Komponen (Kalibrasi)

Pengujian sistem akan dilakukan dengan melihat arsitektur yang menghasilkan persentase terbaik dan mampu meningkatkan performa sistem dalam mengklasifikasikan jenis batik. Pengujian dalam penelitian ini akan menggunakan arsitektur *Convolutional Neural Network* yang merupakan salah satu metode dari Machine Learning dan dirancang untuk mengolah sebuah data dari bentuk citra dua dimensi [21]. Dari beberapa penelitian terdahulu, klasifikasi menggunakan CNN telah mampu untuk mencapai nilai akurasi lebih tinggi dibandingkan klasifikasi menggunakan *k-Nearest Neighbor* (KNN).

Tujuan klasifikasi jenis batik menggunakan CNN yaitu pengujian dapat menghasilkan tingkat akurasi dalam rentang 90% dari data uji yang telah diinput. 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 *epochs* 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 yang 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 *Machine Learning* selaras dengan yang ada di aplikasi. Hasil uji dari aplikasi BatiQu dapat dilihat dari kecocokan gambar 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 19 cm untuk memberi jarak antara kamera *handphone* dengan kain batik. Alasan dibuatnya box adalah menyelaraskan sudut dan pencahayaan saat proses klasifikasi jenis batik.

## Jadwal Pengerjaan

**Gambar 3.6** Jadwal Pengerjaan Capstone Design

NOTE :

\*Warna Merah : Progress yang Telah Selesai

\*Warna Kuning : Progress yang Belum Selesai

## 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 metode *Convolutional Neural Network* (CNN) yang merupakan jenis paling umum untuk pengenalan citra dan pelatihan model jaringan syaraf tiruan. Metode CNN ini akan diaplikasikan dengan arsitektur MobileNet. Selanjutnya adalah penggunaan metode *K-Nearest Neighbor* (KNN) dengan ekstraksi fitur *Gray Level Co-occurrence Matrix* (GLCM). Metode ini mengklasifikasikan objek berdasarkan persamaan dengan objek lain terdekatnya. Kedua metode 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 metode *Convolutional Neural Network* (CNN). Sistem ini dipilih karena memiliki keunggulan salah satunya adalah mudah untuk diaplikasikan ke dalam sistem aplikasi yang akan dibuat. Untuk metode *K-Nearest Neighbor* (KNN) dengan ekstraksi fitur Gray Level Co-occurrence Matrix (GLCM) didapatkan hasil yang kurang optimal untuk mengklasifikasikan data dengan jumlah besar. Maka dengan itu, rencana desain sistem nantinya akan menautkan dua solusi antara skenario pada aplikasi BatiQu berbasis android dan skenario pada *machine learning* menggunakan CNN.

# IMPLEMENTASI

## Implementasi Sistem

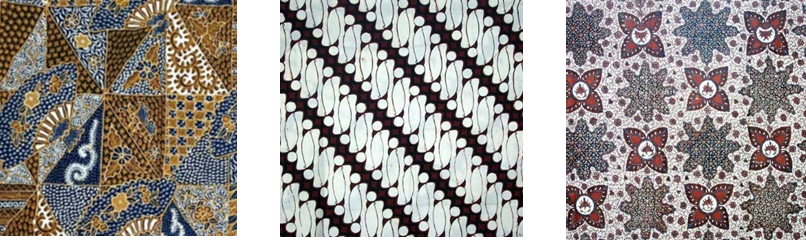
Implementasi sistem merupakan tahapan atau prosedur yang dilakukan untuk dapat menyelesaikan perancangan sistem yang telah dibuat dan akan dilakukan pengujian. Dimana dalam implementasi sistem juga disiapkan sistem yang akan dijalankan. Tujuan Implementasi Sistem ini agar peneliti dapat menyelesaikan 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

### **Machine Learning**

Berikut beberapa cara kerja dari Machine Learning :

#### 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 3 folder untuk dilakukan pelabelan disetiap datanya. Pada data training berjumlah 480 gambar, data validasi berjumlah 120 data dan untuk data testing berjumlah 60 gambar.



1. (b) (c)



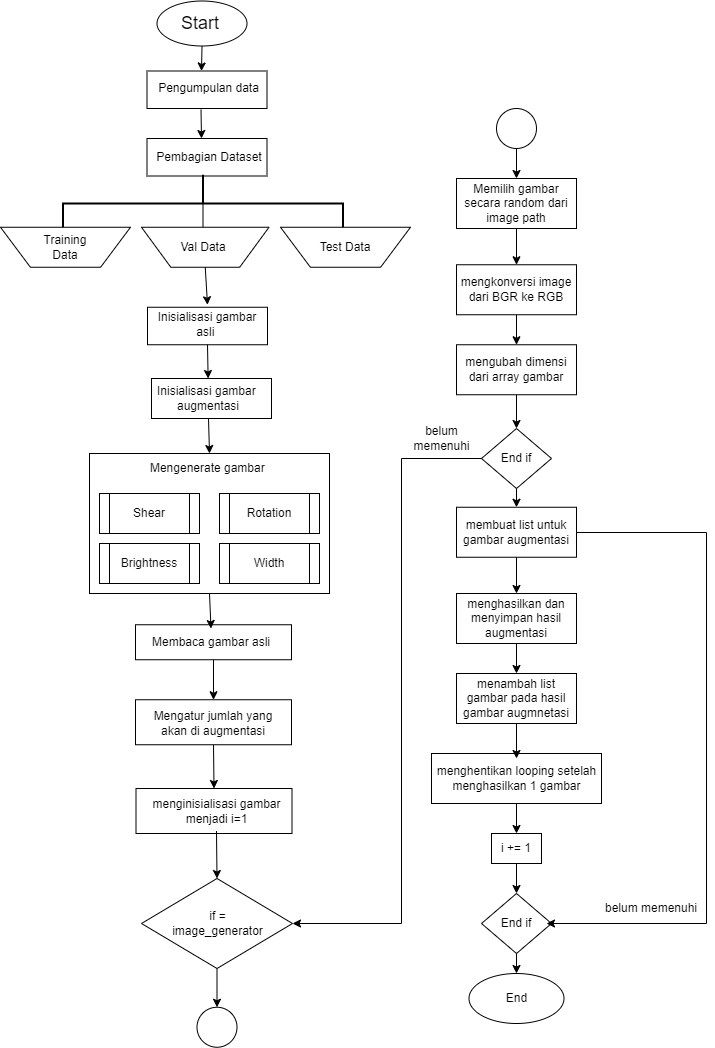
(d) (e) (f)

**Gambar 4.1** (a) Batik Tambal (a) Batik Parang (b) Batik Nitik (c) Batik Megamendung (d) Batik Kawung (e) Batik Ceplok (f)

Gambar 4.1 diatas 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.*

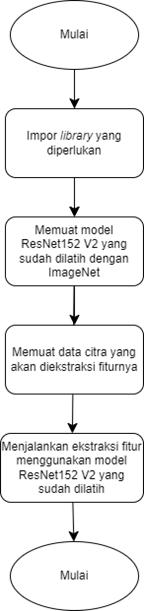
#### *Pre-processing*

Setelah proses dari pengumpulan data, tahapan selanjutnya adalah adanya *pre-processing.* *Pre-processing* adalah tahapan awal pengolahan data gambar pada Convolutional Neural Network (CNN). Dimana teknik *pre-processing* digunakan untuk mempersiapkan data input sebelum memasuki proses bagian *input* model 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 kedalam 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 diinputkan kedalam model. 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 pre-processing menggunakan augmentasi data yang disertakan bobot sepert featur

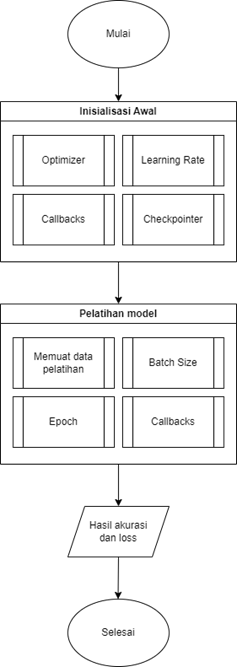
#### *Feature* *extraction*

*Feature extraction* atau ekstraksi fitur 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. 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 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 model *ResNet152 V2*.

#### *Classification*

Klasifikasi pada metode *Convolutional Neural Network* atau yang biasa disingkat menjadi 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, terdapat dua klasifikasi dari arsitektur yang berbeda untuk dibandingkan yaitu arsitektur *MobileNet* V1 dan *Resnet152* V2. Arsitektur *MobileNet* V1 dan *Resnet152* V2 memiliki beberapa kesamaan hyperparameter dan juga parameter yang akan digunakan seperti *Optimizer, Learning Rate, Callbacks,* *Batch Size, Epochs*. Nilai hyperparameter serta parameter dapat memengaruhi output model dari proses pelatihan programnya.

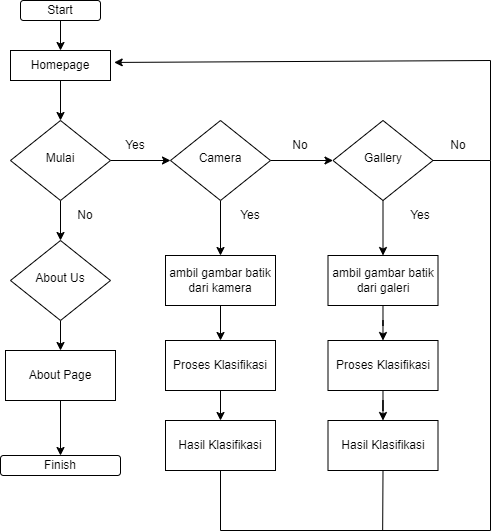
**Gambar 4.4** Flowchart Classification

Pada *flowchart* gambar 4.4 alur proses klasifikasi diawali dengan inisialisasi variabel yang diperlukan. Tahap awal dimulai dengan menambahkan *optimizer, learning rate, callbacks, dan checkpointer*. 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*. Sedangkan, *learning rate* merupakan hyperparameter untuk mengatur nilai perubahan dari bobot model selama proses pelatihan. Selanjutnya, terdapat parameter *callbacks* yang berfungsi untuk memanggil fungsi *checkpointer.* *Checkpointer* sendiri memiliki peran untuk menyimpan model dengan performa terbaik.

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* yang 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

### ***Mobile Application* *BatiQu***

*Aplikasi Mobile* berfungsi sebagai menampilkan hasil klasifikasi jenis batik kepada pengguna. Tampilan pada aplikasi sendiri terdiri dari beberapa elemen seperti tombol, gambar dan teks. Tampilan elemen yang digunakan ditulis dalam format XML (*eXtensible Markup Language*). Fungsionalitas dari setiap elemen menggunakan bahasa pemrograman Java yang dirancang *menggunakan Android Studio* IDE (*Integrated Development Environment*).

Setiap tombol pada halaman awal aplikasi memiliki fungsi masing masing, seperti contoh pada tombol Mulai yang berarti aplikasi tersebut dapat berjalan jika menekan tombol tersebut. Pada tampilan klasifikasi gambar, pengguna dapat mencoba klasifikasi jenis batik dengan tombol kamera yang berfungsi untuk memotret gambar batik yang dimiliki atau tombol galeri jika ingin upload gambar melalui galeri. Elemen – elemen dari pembuatan aplikasi ini dipersiapkan pada proses pembuatan UI (*User Interface*) dengan menggunakan aplikasi Figma dan Canva dan tiap elemen diekspor dengan format .png. Elemen yang telah di ekspor kemudian dimasukkan ke dalam *Android Studio* sebagai *resource* penampilan *Image View* pada tiap halaman aplikasi. Cara kerja *Mobile Application* seperti gambar di bawah ini.

**Gambar 4.5** Flowchart Mobile Application

Pada gambar 4.5 dapat dilihat, bahwa *flowchart* aplikasi dimuali dengan *homepage* kemudian dilanjutkan kedalam menu galeri ataupun kamera yang nantinya akan menampilkan hasil dari gambar yang sudah diklasifikasikan dengan dataset yang tertera.

## Analisis Pengerjaan Implementasi Sistem

Implementasi pada Klasifikasi Batik dilakukan untuk membuat aplikasi yang mampu melakukan prediksi citra gambar menggunakan model *Machine Learning* (ML) dan mengirimkan citra ke BatiQu. Adapun implementasi pada *pre-processing* sebagai berikut :

from tensorflow.keras.preprocessing.image import ImageDataGenerator

import os

import random

import cv2

# Definisikan path direktori gambar asli

images\_path = '/content/drive/MyDrive/REVISI BATIK MANUAL/Data Train/Tambal'

# Definisikan path direktori untuk menyimpan hasil augmentasi

augmented\_path = '/content/drive/MyDrive/SHEAR AUG BATIK/DATA TRAIN/TAMBAL'

# Inisialisasi objek ImageDataGenerator dengan pengaturan augmentasi yang diinginkan

datagen = ImageDataGenerator(

shear\_range= 20, # Rentang rotasi dalam derajat

)

**Gambar 4.6** Implementasi pre-processing

# Membaca daftar path gambar asli

image\_paths = [os.path.join(images\_path, im) for im in os.listdir(images\_path)]

# Mengatur jumlah gambar yang ingin di-generate

images\_to\_generate = 10

i = 1

while i <= images\_to\_generate:

# Memilih gambar asli secara acak

image\_path = random.choice(image\_paths)

# Membaca gambar asli

image = cv2.imread(image\_path)

image = cv2.cvtColor(image, cv2.COLOR\_BGR2RGB)

image = image.reshape((1,) + image.shape)

# Menghasilkan data augmentasi dengan mengaplikasikan transformasi pada gambar

augmented\_images = []

for batch in datagen.flow(image, batch\_size=1, save\_to\_dir= augmented\_path, save\_prefix= 'NonGMB\_Aug\_', save\_format='jpg'):

augmented\_images.append(batch[0])

if len(augmented\_images) == 1: # Menghentikan setelah menghasilkan 1 gambar augmentasi

break

i += 1

Pada tabel 4.6 menjelaskan mengenai implementasi dari pre-processing menggunakan ‘*ImageDataGenerator*’, proses ini digunakan untuk memisahkan data yang digunakan untuk training data ataupun validasi data. Proses untuk menguji test pada pre-processing sebagai berikut :

from tensorflow import keras

image\_path = '/content/drive/MyDrive/BRIGHTNESS AUG BATIK/DATA TRAIN/PARANG/NonGMB\_Aug\_\_0\_103.jpg'

img = keras.preprocessing.image.load\_img(image\_path, target\_size= (300, 300))

img\_tensor = keras.preprocessing.image.img\_to\_array(img)

img\_tensor = np.expand\_dims(img\_tensor, axis=0)

#Menggunakan image data generator untuk membalikkan gambar

datagen = ImageDataGenerator(horizontal\_flip=True, vertical\_flip= True)

#membuat batch pada satu gambar

pic = datagen.flow(img\_tensor, batch\_size= 64)

plt.figure(figsize=(10,5))

for i in range(1, 4):

plt.subplot(1, 3, i)

batch = pic.next()

image\_ = batch[0].astype('uint8')

plt.imshow(image\_)

plt.show()

**Gambar 4.7** Implementasi visualisasi pre-processing

Setelah hasil dari *pre-processing* keluar, dilanjutkan dengan melakukan visualisasi data dari augmentasi data seperti dijelaskan pada tabel 4.7 yang kemudian proses dilanjutkan ke tahap *Feature Extraction* dan juga Klasifikasi.

**Gambar 4.8** Implementasi Feature Extraction

model = tf.keras.Sequential([ ResNet152V2(

include\_top=False,

weights='imagenet',

input\_shape=(300, 300, 3)),

tf.keras.layers.Flatten(),

tf.keras.layers.Dense(128, activation='relu'),

tf.keras.layers.Dense(6, activation='softmax')

Tahapan ekstraksi fitur diimplementasikan dalam bentuk program seperti yang terlihat pada gambar 4.8 Pada program tersebut, digunakan ekstraksi fitur dengan model *ResNet152* V2 yang sudah dilatih dengan dataset imagenet dengan menambahkan tulisan ‘*weights=‘imagenet*”. Pada awal program, dituliskan ‘*model = tf.keras.Sequential*’ yang berfungsi untuk membuat objek model *sequential*  sehingga layer-layer dapat ditambahkan ke dalam model secara berurutan. Kemudian di dalam fungsi *ResNet152* V2 ditambahkan argumen *include\_top=False* karena penelitian ini berfokus pada bagian ekstraksi fitur dari model ini dan tidak memerlukan program untuk menyertakan layer output terakhir. Selanjutnya ditambahkan juga dimensi dari input citra yang digunakan dengan mencantumkan ‘input\_shape=(300, 300, 3)’, lalu disambung dengan menambahkan program untuk *layer.* Pada program *layer*, digunakan *layer flatten* untuk mengubah output dari model *ResNet152* V2 menjadi vektor satu dimensi, kemudian dilanjutkan dengan *‘layer dense’* dengan 128 unit dan aktivasi ‘*ReLu*’ agar representasi yang dihasilkan dari proses fiturnya menjadi lebih kompleks. Yang terakhir ditambahkan ‘*layer dense’* dengan 6 unit karena disesuaikan dengan jumlah kelas klasifikasinya dan disertai dengan fungsi aktivasi ‘*softmax*’ untuk menghasilkan probabilitas prediksi kelas sesuai dengan hasil ekstraksi fitur yang dimasukkan ke model. Tahapan terakhir untuk proses ini adalah Klasifikasi, berikut merupakan implementasi klasifikasi :

**Gambar 4.9** Implementasi klasifikasi untuk mengatur optimizer dan learning rate

#setting optimizer

filepath='model.h5'

opt  = tf.keras.optimizers.legacy.Adam(learning\_rate=0.0001 #learning rate biasanya antara 0.001 - 0.1. jika terlalu besar akan terjadi overshooting

model.compile(loss='categorical\_crossentropy', optimizer=opt, metrics=['acc'])

checkpointer = ModelCheckpoint(filepath,

monitor = 'val\_acc',

verbose=1,

save\_best\_only=True,

mode='max')

callbacks\_list = [checkpointer]

Pada gambar 4.9 merupakan tahapan klasifikasi untuk mendefinisikan bahwa ‘*optimizer*’ yang digunakan yaitu Adam dan learning rate yang diatur sebesar 0.0001. ‘*Optimizer*’ Adam sangat umum digunakan dalam metode CNN karena dapat memberikan hasil yang lebih baik. Untuk pemilihan nilai *learning rate* berada dalam rentang 0 sampai 1 [1]. *Leaning rate* dapat mempengaruhi kecepatan dari hasil pelatihan model.

**Gambar 4.10** Implementasi klasifikasi untuk proses pelatihan dan evaluasi model.

model\_history = [ ]

history = model.fit(

train\_generator,

validation\_data = validation\_generator,

batch\_size= 32,

epochs= 200,

callbacks= [checkpointer] )

# Evaluasi model pada data pelatihan

train\_score = model.evaluate(train\_generator)

print('Train Score: ', train\_score)

val\_score = model.evaluate(validation\_generator)

print('Val Score: ', val\_score)

test\_score = model.evaluate(test\_set)

print('Test Score: ', test\_score)

Setelah mengatur nilai pada bagian *optimizer* dan *learning rate,* pada gambar 4.10 merupakan program untuk melakukan pelatihan dan evaluasi model. Program tersebut telah memproses data *training* dan data *validation* yang akan dilatih dengan *‘train\_generator’* dan dievaluasi dengan *‘validation\_generator’*. Selanjutnya, terdapat *batch\_size* yang terdiri dari 32 sampel. Selain itu, parameter *epoch* sebesar 200 juga diperlukan untuk mencapai tingkat akurasi terbaik. Nilai *epoch* menentukan hasil pembelajaran model dalam memahami pola dari dataset yang telah dilatih. Apabila kedua *hyperparameter* tersebut telah disesuaikan, maka diperlukan evaluasi kinerja model pada data *training,* data *validation,* dan data *testing.* Evaluasi dari ketiga dataset tersebut akan ditampilkan dengan code *‘print’* untuk menghasilkan skor atau performa model. Hal ini mengacu pada data yang telah dilatih dapat memahami dan memprediksi dengan akurasi yang baik.

print(history.history.keys())

# summarize history for accuracy

plt.plot(history.history['accuracy'])

plt.plot(history.history['val\_accuracy'])

plt.title('model accuracy')

plt.ylabel('accuracy')

plt.xlabel('epoch')

plt.legend(['train', 'validation'], loc='upper left')

plt.show()

# summarize history for loss

plt.plot(history.history['loss'])

plt.plot(history.history['val\_loss'])

plt.title('model loss')

plt.ylabel('loss')

plt.xlabel('epoch')

plt.legend(['train', 'validation'], loc='upper left')

plt.show()

**Gambar 4.11** Implementasi Klasifikasi untuk visualisasi performa model dari proses training.

Setelah melakukan pelatihan dan evaluasi kinerja model, program pada gambar 4.11 dapat membantu menampilkan perubahan akurasi dan loss model selama proses training. Hasil visualisasi ini dapat dianalisis dan dipahami lebih dalam mengenai performa yang telah didapatkan.

## Hasil Akhir Sistem

Jelaskan secara detail bagaimana hasil akhir sistem yang telah dibuat

## Kesimpulan dan Ringkasan CD-4

# PENGUJIAN SISTEM

## Skema Pengujian Sistem

Pada perancangan sistem Klasifikasi Motif Batik menggunakan arsitektur ... dibuat dalam beberapa tahapan dalm bentuk diagram alir pada gambar 5.1 sebagai berikut :



**Gambar 5.1** Diagram Alir Pengujian sistem

Skema pengujian sistem ini akan dilakukan dengan bertahap disertakan pengujian pada box BatiQu yang telah dirancang sebelumnya. Dimana nanti batik akan dimasukkan kedalam box dan diuji dengan kecerahan yang dihasilkan oleh LED pada box.

## Proses Pengujian

Pengujian sistem dilakukan untuk dapat mengklasifikasikan citra batik yang digunakan sebagai tolak ukur dari penilaian dan mengevaluasi dari kinerja suatu sistem melalui beberapa sub-sistem yang telah dirancang sebelumnya. Untuk mengetahui kualitas sistem dapat dilihat dari pengujian performa yang akan dilakukan yaitu pada bagian *pre-processing, feature extraction* dan Klaisifkasi.

### *Pre-processing*

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

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.

1. Rotation :

Rotation merupakan parameter yang berguna untuk mengubah rotasi gambar secara acak. Tujuan dari rotation untuk membantu dalam meningkatkkan 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.

1. 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.

1. Shear :

### *Feature Extraction*

Jelaskan langkah-langkah dan hasil proses pengujian kedua. Banyaknya proses pengujian dapat disesuaikan jumlahnya sesuai dengan sistem yang dibuat

## Analisis Hasil Pengujian

Berisi analisis terkait hasil-hasil pengujian yang sudah dilakukan

### Analisis Hasil Pengujian 1

Analisa terkait hasil pengujian pertama

### Analisis Hasil Pengujian 2

Analisa terkait hasil pengujian kedua. Banyaknya analisa hasil pengujian dapat disesuaikan jumlahnya sesuai dengan sistem yang dibuat

## Kesimpulan dan Ringkasan CD-5

DAFTAR PUSTAKA

[1] V. S. Thomas, S. Darvesh, C. MacKnight, and K. Rockwood, “Estimating the prevalence of dementia in elderly people: a comparison of the Canadian Study of Health and Aging and National Population Health Survey approaches,” *Int Psychogeriatr*, vol. 13 Supp 1, no. SUPPL. 1, pp. 169–175, 2001, doi: 10.1017/S1041610202008116.

[2] M. M. Baig and H. Gholamhosseini, “Smart health monitoring systems: an overview of design and modeling,” *J Med Syst*, vol. 37, no. 2, Apr. 2013, doi: 10.1007/S10916-012-9898-Z.

[3] M. M. Alam, H. Malik, M. I. Khan, T. Pardy, A. Kuusik, and Y. le Moullec, “A survey on the roles of communication technologies in IoT-Based personalized healthcare applications,” *IEEE Access*, vol. 6, pp. 36611–36631, Jul. 2018, doi: 10.1109/ACCESS.2018.2853148.

[4] S. Li, L. da Xu, and X. Wang, “A continuous biomedical signal acquisition system based on compressed sensing in body sensor networks,” *IEEE Trans Industr Inform*, vol. 9, no. 3, pp. 1764–1771, 2013, doi: 10.1109/TII.2013.2245334.

[5] P. Rashidi and A. Mihailidis, “A survey on ambient-assisted living tools for older adults,” *IEEE J Biomed Health Inform*, vol. 17, no. 3, pp. 579–590, 2013, doi: 10.1109/JBHI.2012.2234129.

[6] A. Arcelus, R. Goubran, M. H. Jones, and F. Knoefel, “Integration of smart home technologies in a health monitoring system for the elderly,” *Proceedings - 21st International Conference on Advanced Information Networking and Applications Workshops/Symposia, AINAW’07*, vol. 1, pp. 820–825, 2007, doi: 10.1109/AINAW.2007.209.

[7] A. Pantelopoulos and N. G. Bourbakis, “A survey on wearable sensor-based systems for health monitoring and prognosis,” *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 40, no. 1, pp. 1–12, 2010, doi: 10.1109/TSMCC.2009.2032660.

[8] M. E. Garbelini *et al.*, “SweynTooth: Unleashing Mayhem over Bluetooth Low Energy”, Accessed: May 31, 2022. [Online]. Available: https://www.usenix.org/conference/atc20/presentation/garbelini

[9] A. S. Seferagi´c *et al.*, “Survey on Wireless Technology Trade-Offs for the Industrial Internet of Things,” *Sensors 2020, Vol. 20, Page 488*, vol. 20, no. 2, p. 488, Jan. 2020, doi: 10.3390/S20020488.

[10] V. S. Thomas, S. Darvesh, C. MacKnight, and K. Rockwood, “Estimating the Prevalence of Dementia in Elderly People: A Comparison of the Canadian Study of Health and Aging and National Population Health Survey Approaches,” *Int Psychogeriatr*, vol. 13, no. S1, pp. 169–175, 2001, doi: 10.1017/S1041610202008116.

[11] S. Majumder, T. Mondal, and M. J. Deen, “Wearable Sensors for Remote Health Monitoring,” *Sensors (Basel)*, vol. 17, no. 1, Jan. 2017, doi: 10.3390/S17010130.

LAMPIRAN CD-1

LAMPIRAN CD-2

LAMPIRAN CD-3

LAMPIRAN CD-4

LAMPIRAN CD-5