**PENGKLASIFIKASIAN JENIS PERMATA DENGAN COLOR HISTOGRAM MENGGUNAKAN METODE THRESHOLDING DAN PENGUKURAN JARAK L1-METRIC**

**SKRIPSI**

Sebagai Salah Satu Syarat Untuk Meraih

Gelar Sarjana Komputer

Oleh:

Clen Hartanto

32170043



Fakultas Teknologi Dan Desain

Program Studi Teknik Informatika

Universitas Bunda Mulia

Jakarta

2021

**ABSTRAK**

Bumi ini memiliki banyak kekayaan alam, salah satunya adalah batu permata. Permata ini pun ada banyak jenis berdasarkan dari bentuk dan warna nya. Maka dari itu perlu adanya suatu perantara untuk mengklasifikasikan berbagai jenis permata agar umat awam bisa mengetahui jenis permata yang ingin diketahui.

Pada penelitian ini, aplikasi menggunakan metode Histogram untuk pengambilan fitur intensitas warna dan *L1-Metric* untuk pengukuran jarak perbandingan fitur. Histogram merupakan metode pengambilan fitur warna berupa grafik yang menggambarkan penyebaran pada nilai - nilai intensitas piksel suatu citra. Metode tersebut dilakukan untuk mengekstrak ciri foto batu permata yang didapatkan dalam pengumpulan data dan dibandingkan dengan data testing menggunakan *L1-Metric* untuk mengenali jenis batu permata.

Hasil penelitian yang diharapkan pada program ini adalah program dapat mengenali jenis jenis Permata berdasarkan data yang disimpan dalam *database* serta dapat mengukur tingkat keakurasian metode tersebut dalam mengenali jenis jenis *Permata.*

Kesimpulan yang bisa didapatkan dari penelitian ini berupa tingkat keakurasian penggunaan algoritma dalam mengidentifikasi jenis batu permata.

Kata Kunci: Permata, Computer Vision, Histogram, L1-Metric

**DAFTAR ISI**

[**ABSTRAK** ii](#_Toc67748656)

[**DAFTAR ISI** iii](#_Toc67748657)

[**BAB I PENDAHULUAN** 1](#_Toc67748658)

[**1.1 Latar Belakang** 1](#_Toc67748659)

[**1.2 Rumusan Masalah** 2](#_Toc67748660)

[**1.3** **Tujuan dan Manfaat** 2](#_Toc67748661)

[1.3.1 **Tujuan** 3](#_Toc67748662)

[1.3.2 **Manfaat** 3](#_Toc67748663)

[**1.4** **Ruang Lingkup** 3](#_Toc67748664)

[**1.5** **Metode Pengumpulan Data** 3](#_Toc67748665)

[**1.6 Sistematika Penulisan** 4](#_Toc67748666)

[**BAB II LANDASAN TEORI** 6](#_Toc67748667)

[**1.1** **Permata** 6](#_Toc67748668)

[**1.2** **Histogram** 6](#_Toc67748669)

[***1.3*** ***L1-Metric*** 6](#_Toc67748670)

[**1.4** **Pascal** 7](#_Toc67748671)

[**1.5** **PHP** 7](#_Toc67748672)

[***1.6*** ***MySQL*** 8](#_Toc67748673)

[**2.6 *Flowchart*** 8](#_Toc67748674)

[**2.7 *Gantt Chart*** 11](#_Toc67748675)

[**2.8 UML** 11](#_Toc67748676)

[**2.9 Penelitian Terdahulu** 11](#_Toc67748677)

[**BAB III ANALISIS DAN PERANCANGAN** 12](#_Toc67748678)

[**3.1 Analisis Kebutuhan Fungsional dan Non-Fungsional** 12](#_Toc67748679)

[**3.1.1 Kebutuhan Fungsional** 12](#_Toc67748680)

[**3.1.2 Kebutuhan Non-Fungsional** 12](#_Toc67748681)

[**3.2 Pemilihan Algoritma / Teknik / Metode** 13](#_Toc67748682)

[**3.2.1 Metode Perancangan Sistem** 13](#_Toc67748683)

[**3.2.2 Pemilhan Algoritma / Metode** 14](#_Toc67748684)

[**3.3 Perancangan Proses** 15](#_Toc67748685)

[**3.3.1 *Flowchart* Program** 15](#_Toc67748686)

[**3.3.2 *Flowchart* Algoritma** 16](#_Toc67748687)

[**3.3.3 *Use* *Case* *Diagram*** 16](#_Toc67748688)

[**3.3.4 *Activity Diagram*** 17](#_Toc67748689)

[**3.3.5 *Class Diagram*** 18](#_Toc67748690)

[**3.3.6 *Sequence* *Diagram*** 18](#_Toc67748691)

[**3.4 Perancangan Basis Data** 20](#_Toc67748692)

[**3.4.1 Normalisasi** 20](#_Toc67748693)

[**3.5 *Gantt* *Chart*** 21](#_Toc67748694)

[**3.6 Perancangan Tampilan** 21](#_Toc67748695)

[**DAFTAR PUSTAKA** 22](#_Toc67748696)

**BAB I  
PENDAHULUAN**

**1.1 Latar Belakang**

Bumi ini memiliki banyak kekayaan alam, salah satunya adalah batu-batuan. Batu-batuan ini memiliki banyak manfaat, salah satunya adalah menjadi perhiasan. Salah satu jenis batuan yang bisa dijadikan perhiasan adalah batu permata. Karena dijadikan perhiasan, batu permata ini mempunyai nilai harga jual yang tinggi. Batu permata ini pun ada banyak jenis berdasarkan dari bentuk dan warna nya dengan harga jual yang berbeda pula. Maka dari itu perlu adanya suatu perantara untuk mengklasifikasikan berbagai jenis permata agar umat awam bisa mengetahui jenis permata yang ingin diketahui.

*Computer Vision* merupakan sebuah cabang keilmuan bagian dari *artificial intelligence* atau kecerdasan buatan yang dapat melihat seperti manusia dan memberikan informasi kepada komputer untuk di proses sesuai dengan tujuannya. Dalam prosesnya *Computer Vision* terdiri dari berberapa tahap mulai dari mengambil gambar, *Pre-processing*, mengekstrak fitur, dan mengenali pola.

Histogram citra merupakan diagram distribusi fitur warna berupa grafik yang menggambarkan penyebaran pada nilai - nilai intensitas piksel suatu citra.

Penggunaan metode *L1-Metric* dalam pengenalan pola berfungsi untuk mengenali bentuk objek permata berdasarkan jarak terdekat suatu nilai fitur pada objek citra dengan data training yang ada dalam database.

Pada penelitian ini, penulis hendak menerapkan histogram untuk mendapatkan distribusi fitur intensitas warna dan menggunakan thresholding untuk memisahkan objek dengan latar belakang foto juga menggunakan metode *L1-metric* dalam pengukuran jarak untuk mengenali pola distribusi fitur dengan harapan dapat mengklasifikasikan jenis-jenis Permata. Maka dari itu, penelitian ini akan mengambil judul pengklasifikasian jenis permata dengan histogram dengan menggunakan metode thresholding dan pengukuran jarak L1-metric

**1.2 Rumusan Masalah**

Berdasarkan dari latar belakang, maka rumusan masalah yang akan dibahas pada skripsi ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan *thresholding* dalam pengklasifikasian jenis permata dengan histogram dengan menggunakan metode *thresholding* dan pengukuran jarak *L1-metric*?
2. Bagaimana tingkat keakurasian klasifikasi permata dengan *L1-metric* dalam program pengklasifikasian jenis permata dengan histogram dengan menggunakan metode *thresholding* dan pengukuran jarak *L1-metric*?
   1. **Tujuan dan Manfaat**

Tujuan dan manfaat adalah untuk menjelaskan apa yang akan menjadi tujuan dalam pembuatan aplikasi rekomendasi ini dan juga menjelaskan manfaat – manfaat yang akan di dapatkan dari program ini.

### 1.3.1 **Tujuan**

Sesuai dengan rumusan masalah, tujuan yang hendak dicapai pada penelitian kami adalah sebagai berikut:

1. Dapat mengimplementasikan *thresholding* dalam pengklasifikasian jenis permata dengan histogram dengan menggunakan metode *thresholding* dan pengukuran jarak *L1-metric.*
2. Dapat mengukur tingkat keakurasian klasifikasi permata dengan *L1-metric* dalam program pengklasifikasian jenis permata dengan histogram dengan menggunakan metode *thresholding* dan pengukuran jarak *L1-metric.*

### 1.3.2 **Manfaat**

Manfaat yang hendak dicapai dalam penelitian ini adalah dapat dimanfaatkan sebagai alat pencarian dan pengenalan jenis-jenis batu permata.

* 1. **Ruang Lingkup**

Untuk memastikan agar penelitian yang dilakukan tidak menyimpang dari pokok pembahasan, maka kami telah menetapkan batasan-batasan dalam penelitian ini, yaitu:

1. Data *training* yang diperoleh merupakan foto permata yang diambil dari internet dengan mengambil 20 jenis permata dengan background putih.
2. Data testing yang diperoleh merupakan foto permata yang diambil dari internet dengan background putih atau warna polos lainnya.
   1. **Metode Pengumpulan Data**

Metode yang digunakan untuk mengumpulkan data ialah dengan melakukan riset secara *online* terhadap jurnal-jurnal yang sudah terpublikasi dan jurnal international untuk mengetahui apakah metode thresholding dan pengukuran jarak *L1-metric* dapat digunakan pada program pengklasifikasian jenis permata dengan histogram dengan menggunakan metode *thresholding* dan pengukuran jarak *L1-metric.*

1. **Analisis Pengumpulan Data**

Metode yang digunakan untuk mengumpulkan data ialah dengan melakukan riset secara online terhadap jurnal jurnal untuk mengetahui apakah metode thresholding dan pengukuran jarak *L1-metric* dapat digunakan pada program pengklasifikasian jenis permata dengan histogram dengan menggunakan metode *thresholding* dan pengukuran jarak *L1-metric.*

1. **Algoritma perancangan aplikasi**

Perancangan program yuntuk memisahkan objek dan ang dibuat dengan menggunakan *thresholding*  yang selanjutnya akan digunakan *L1-metric* untuk pengukuran jarak perbandingan nya.

1. **Pemodelan sistem**

Pemodelan sistem pada program ini menggunakan bahasa pemrograman *pascal* dan menggunakan database MySQL.

**1.6 Sistematika Penulisan**

1. **BAB 1 PENDAHULUAN**

Bab 1 berisi Latar Belakang masalah, Rumusan Masalah, Tujuan dan Manfaat, Ruang Lingkup, Metodologi Penelitian, dan Sistematika Penulisan

1. **BAB 2 LANDASAN TEORI**

Bab 2 berisi teori yang akan menjadi dasar untuk digunakan dalam menyelesaikan permasalahan yang sudah dijelaskan pada bab sebelumnya

1. **BAB 3 ANALISIS DAN PERANCANGAN**

Bab 3 berisi tentang analisis kebutuhan yang diperlukan dalam menyelesaikan permasalahan yang sudah dijelaskan pada bab sebelumnya

1. **BAB 4 IMPLEMENTASI**

Bab 4 berisi penerapan teori ke dalam bahasa pemrograman dan pengujian terhadap program

1. **BAB 5 SIMPULAN DAN SARAN**

Bab 5 berisi kesimpulan dan saran berdasarkan hasil penelitian dan pembahasan.

**BAB II  
LANDASAN TEORI**

* 1. **Permata**

Batu permata merupakan batuan yang terbentuk dari pendinginan magma atau lava yg bersuhu diatas 1.000 derajat *Celsius* yang berasal dari perut bumi yg keluar melalui proses erupsi gunung berapi, rentang saat yang dibutuhkan pun mencapai ratusan hingga milyaran tahun yang dimana unsurnya terdiri menurut satu atau beberapa komponen kimia, perlu melalui beberapa proses buat merubah bahan batu permata menjadi batu permata dan bisa digunakan menjadi perhiasan, seperti proses pemotongan, proses pembentukan, & penggosokan.

Batu permata dikenal lantaran keindahan, nilai, dan daya tahannya, yang memungkinkan untuk diteruskan menurut satu generasi ke generasi berikutnya. Sepanjang waktu, batu permata sudah memikat dan mempesona orang-orang berdasarkan seluruh dunia. Batu permata yang berwarna sangat digunakan dan dihargai dengan banyak sekali cara. Misalnya, banyaknya batu permata di "Permata Mahkota" mewakili kekuatan & kekayaan untuk keluarga Kerajaan. Sebuah tradisi yang dihormati dari waktu ke waktu yang telah diwariskan merupakan batu kelahiran. Batu kelahiran merupakan permata yg mewakili bulan Anda lahir. Batu kelahiran ini diberikan menjadi hibah untuk merayakan kehidupan seseorang, & sering dianggap sebagai cerminan menurut kepribadian mereka. Seperti manusia, semua batu permata tidak sama jadi ada sesuatu untuk setiap selera penggunanya.

**1.1.1 Klasifikasi Batu Permata**

Ada 2 klasifikasi batu permata: mulia dan semi-mulia. Jenis batu yang merupakan batu mulia adalah jenis *Diamond, Pearl, Jade, Safir, Ruby, dan Zamrud*. Permata - permata ini adalah salah satu dari banyak batu permata yang bisa disebut berharga. Jenis batu yang merupakan Batu Semi Mulia adalah *Amethyst, Aquamarine, Citrine, Garnet, Onyx, Opal, Peridot, Tanzanite, dan Topaz*. Semua permata, baik itu mulia atau semi mulia, semua sangat berharga. Nilainya bergantung dalam *poly* elemen termasuk warna, ukuran, kualitas, dan kelangkaan.

* + 1. **Kriteria Batu Permata**

Kriteria batu permata dibagi menjadi empat, yaitu warna, kejernihan, potongan, dan bobot.

1. Warna

Warna pada batu permata bergantung pada *hue*, yaitu warna dasar yang terdapat dalam batu permata dan juga *tone*, adalah seberapa banyak sisi gelap dan terang pada batu permata.

2. Kejernihan

Kejernihan disini mewakili kebersihan dan kemurnian dari sebuah batu permata. Kebanyakan batu permata mempunyai kotoran atau cacat.

3. Potongan

Jenis potongan pada batu permata dibagi menjadi 3 macam, yaitu:

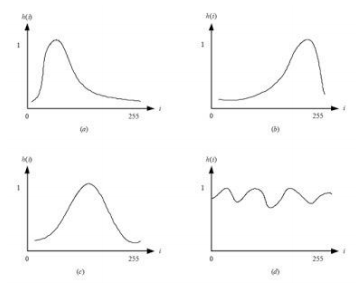
* 1. *Cabochon*, tidak mempunyai facet sama sekali, berbentuk polos seperti bentuk setengah telur.
  2. *Faceted*, bentuk potongan dengan asahan sudut kecil seperti potongan berlian.
  3. *Mixed* *cut*, perpaduan dari *cabochon* dan *faceted*, biasanya permukaan atas bagian kepalanya polos dan bagian bawahnya di facet

4. Bobot

Suatu batu permata dijual menurut ukuran bobot nya. Makin besar bobot batu permata, maka semakin mahal harga nya.

* 1. **Histogram**

Pengertian histogram dalam pengolahan citra adalah representasi grafis untuk distribusi warna dari citra digital atau menggambarkan penyebaran nilai-nilai intensitas pixel dari suatu citra atau bagian tertentu di dalam citra[1]. Dari data histogram citra, kita dapat mengetahui frekuensi kemunculan intensitas warna pada citra, kecerahan citra, serta kontras dari sebuah citra.



**Gambar 2.1 Kurva Histogram [1]**

* 1. ***L1-Metric***

*L1-metric* melakukan pengukuran jarak antara fitur-fitur yang dimiliki dua buah citra dimana jarak kedua buah citra ini yang nantinya akan dipertimbangkan sebagai kemiripan antara dua buah citra [2]. Jika nilai jarak yang dihasilkan semakin kecil, maka kedua citra akan dianggap semakin mirip. Sebaliknya, jika nilai jarak yang dihasilkan semakin besar, maka kedua citra akan dianggap semakin berbeda.

**Rumus 2.1 *L1-Metric* [2]**

Ket:  
l : pencacah fitur   
n : jumlah fitur   
I : himpunan fitur citra pada top stack / citra yang terakhir disimpan   
i : fitur citra pada top stack / citra yang terakhir disimpan   
H : himpunan fitur citra yang akan diuji   
h : fitur citra yang akan diuji   
d(I,H) : jarak citra I terhadap citra H

* 1. **Pascal**

Pascal adalah bahasa tingkat tinggi dimana bahasa pemrograman ini memperbolehkan pemrogram untuk menentukan instruksi pemrograman dalam Bahasa yang mirip dengan Bahasa Inggris, namun sangat terbatas. Untuk menghindari adanya kemenduaan arti dan untuk mempermudah penerjemahan program oleh pengompail kedalam perintah-perintah biner, sintaks bahasa yang bersangkutan mensyaratkan aturan yang ketat[3].

* 1. **PHP**

PHP (*Hypertext* *Preprocessor*) yaitu bahasa pemrograman *web* *server*-*side* yang bersifat *open* *source*, terintegrasi dengan HTML dan berada pada *server*. PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis[4]. Mekanisme ini membuat informasi yang diterima oleh client selalu merupakan informasi yang terbaru. Semua script PHP akan dieksekusi oleh *server* dimana *script* tersebut akan dijalankan.

* 1. ***MySQL***

*MySQL* adalah salah satu program yang dapat digunakan sebagai *database*, dan merupakan salah satu *software* untuk *database* *server* yang banyak digunakan [4]. *MySQL* sendiri menggunakan bahasa pemrograman *SQL*. *MySQL* mempunyai kelebihan, diantaranya:

* + - 1. Dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah.
      2. Memiliki kecepatan yang bagus dalam menangani *query* sederhana.

1. Memiliki operator dan fungsi secara penuh dan mendukung perintah *select* dan *where* dalam perintah *query*.
2. Memiliki keamanan yang bagus karena beberapa lapisan sekuritas seperti *level* *subnetmask*, nama host, dan izin akses *user*, dengan sistem perizinan yang mendetail serta sandi terenkripsi.
3. Mampu menangani basis data dalam skala besar, dengan jumlah rekaman lebih dari 50 juta dan 60 ribu tabel serta kurang lebih 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

**2.6 *Flowchart***

*Flowchart* atau bagan alir adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika [5]. *Flowchart* berguna untuk menjelaskan urutan proses yang pelaksanaannya memiliki banyak option pilihan atau percabangan. *flowchart* memerlukan simbol-simbol yang berbentuk seperti persegi, jajar genjang, maupun dengan bentuk lain yang kemudian dihubungkan dengan garis anak panah. Setiap simbol pada diagram bertujuan untuk menunjukkan langkah-langkah penyelesaian program, sedangkan bagi garis berpanah bertujuan untuk menunjukkan urutan-urutan kejadiansuatu program.

Tujuan dari *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian terhadap suatu masalah secara sederhana, terurai, rapi dan jelas dapat menggunakan simbol-simbol yang standar [5].

**Tabel 2.1 *Program Flowchart Symbols* [5].**

|  |  |
| --- | --- |
|  | ***Flow Direction Symbol*** Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan connecting line. |
|  | ***Terminator Symbol***Yaitu simbol untuk permulaan (*start*) atau akhir (*stop*) dari suatu kegiatan. |
|  | ***Connector Symbol*** Yaitu simbol untuk keluar - masuk atau penyambungan proses dalam lembar / halaman yang sama. |
|  | ***Connector Symbol*** Yaitu simbol untuk keluar - masuk atau penyambungan proses dalam lembar / halaman yang berbeda. |
|  | ***Processing Symbol*** Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer. |
|  | **Simbol *Manual Operation*** Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer. |
|  | **Simbol *Decision*** Simbol pemilihan proses berdasarkan kondisi yang ada. |
|  | **Simbol *Input - Output*** Simbol yang menyatakan proses *input* dan *output* tanpa tergantung dengan jenis peralatannya. |
|  | **Simbol *Manual Input*** Simbol untk pemasukkan data secara manual *on-line keyboard*. |
|  | **Simbol *Preparation*** Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam *storage*. |
|  | **Simbol *Predefine Proses*** Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedur. |
|  | **Simbol *Display*** Simbol yang menyatakan peralatan *output* yang digunakan yaitu layar, *plotter*, *printer*, dan sebagainya. |
|  | **Simbol *Disk and On-line Storage*** Simbol yang menyatakan *input* yang berasal dari *disk* atau disimpan ke *disk*. |
|  | **Simbol Magnetik *Tape Unit*** Simbol yang menyatakan *input* berasal dari pita magnetik atau *output* disimpan ke pita magnetik. |
|  | **Simbol *Punch Card*** Simbol yang menyatakan bahwa *input* berasal dari kartu atau *output* ditulis ke kartu. |
|  | **Simbol Dokumen** Simbol yang menyatakan *input* berasal dari dokumen dalam bentuk kertas atau *output* dicetak ke kertas. |

**2.7 *Gantt Chart***

Gantt Chart adalah sejenis grafik batang (Bar Chart) yang digunakan untuk menunjukan Tugas-tugas pada Proyek serta Jadwal dan waktu pelaksanaannya, seperti waktu dimulainya tugas tersebut dan juga batas waktu yang digunakan untuk menyelesaikan tugas yang bersangkutan [6].

Gantt Chart sendiri dikembangkan oleh Henry Laurence Gantt pada tahun 1910. pada dasarnya, *Gantt* *Chart* adalah suatu gambaran untuk menjelaskan perencanan, penjadwalan dan pemantauan setiap kegiatan atau aktivitas pada suatu proyek.

* 1. **UML**

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). UML sendiri pula memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep usaha proses, penulisan kelas-kelas pada bahasa program, skema database, dan komponen-komponen yg diperlukan pada sistem.

* + 1. ***Use Case Diagram***

*Use Case Diagram* merupakan pemodelan untuk mendeskripsikan tipikal interaksi antara user sebuah sistem dengan sistemnnya sendiri melalui sebuah cerita bagaimana sistem itu dipakai.

**Tabel 2.2 *Use Case Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
| 1 |  | *Actor* | Mendeskripsikan peran yang dimainkan oleh pengguna ketika berinteraksi dengan *use case*. |
| 2 |  | *Include* | Menspesifikasikan bahwa *use case* sumber secara eksplisit. |
| 3 |  | *Association* | Mendeskripsikan hubungan antara objek satu dengan objek lainnya. |
| 4 |  | *System* | Mendeskripsikan paket yang menampilkan sistem secara terbatas. |
| 5 |  | *Use Case* | Mendeskripsikan urutan aksi-aksi yg ditampilkan sistem yg menghasilkan suatu output yg terukur bagi suatu aktor. |

* + 1. ***Activity Diagram***

*Activity diagram* merupakan diagram yg mendeskripsikan *workflow* atau aktivitas menurut sebuah sistem yang ada pada perangkat lunak.

**Tabel 2.3 *Activity Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
| 1 |  | *Initial Node* | Bagaimana objek dibentuk atau diawali. |
| 2 |  | *Partition (Vertical)* | Pengelompokan aktifitas berdasarkan aktor dan sistem |
| 3 |  | *Action* | State dari sistem yang mencerminkan eksekusi dari suatu aksi |
| 4 |  | *Fork Node* | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran. |
| 5 |  | *Activity Final Node* | Bagaimana objek dibentuk  dan diakhiri. |
| 6 |  | *Decision Node & Merge Node* | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |

* + 1. ***Class Diagram***

*Class diagram* merupakan gambaran struktur sistem berdasarkan segi pendefinisian kelas-kelas yang akan dibuat untuk membentuk sistem. *Class diagram* terdiri dari atribut & operasi dengan tujuan pembuat program dapat menciptakan hubungan antara dokumentasi perancangan & perangkat lunak sesuai.

**Tabel 2.4 *Class Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
| 1 |  | *Class* | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
| 2 |  | *Association* | Apa yang menghubungkan  antara objek satu dengan objek lainnya. |
| 3 |  | *Composition* | relasi yang mempunyai tingkat ketergantungan yang kuat |
| 4 |  | *Dependency* | Hubungan antara satu objek yang bergantung dengan objek lain. |
| 5 |  | *Generalization* | Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada diatasnya objek  induk (*ancestor*). |
| 6 |  | *Multiplicity* | Jumlah banyaknya obyek sebuah *class* yang berelasi dengan sebuah obyek lain pada *class* lain yang berasosiasi dengan *class* tersebut. |

* + 1. ***Sequence Diagram***

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Gambaran sequence diagram dibentuk minimal sebanyak pendefinisan *use case* yg memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan hubungan jalannya pesan sudah dicakup pada *sequence diagram.*

**Tabel 2.5 *Sequence Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
| 1 |  | *LifeLine (Actor)* | Objek *actor*, antarmuka yang saling berinteraksi. |
| 2 |  | *LifeLine (entity)* | Objek *entity*, antarmuka yang saling berinteraksi. |
| 3 |  | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi. |
| 4 |  | *Execution Occurance* | Menunjukkan focus control objek pada suatu waktu. |

* 1. **Penelitian Terdahulu**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Penulis** | **Judul** | **Hasil** |
| 1 | Ratri Enggar Pawening, Wali Ja’far Shudiq, Wahyuni | Klasifikasi Kualitas Jeruk Lokal Berdasarkan Tekstur Dan Bentuk Menggunakan Metode k-Nearest Neighbor (K-NN)[7] | Metode k-Nearest Neighbour berhasil mengidentifikasi jeruk baik dan buruk dengan tepat berdasarkan bentuk dan teksturnya dengan Akurasi tertinggi didapatkan sebesar 93,33%, sedangkan tingkat akurasi terendah didapatkan sebesar 86,20% |
| 2 | Joan Angelina Widians, Herman Santoso Pakpahan, Edy Budiman, Haviluddin, Maratus Soleha | Klasifikasi Jenis Bawang Menggunakan Metode K-Nearest Neighbor Berdasarkan Ekstraksi Fitur Bentuk Dan Tekstur[8] | Penggunaan metode K-Nearest Neighbor (KNN) berdasarkan ekstraksi fitur bentuk dan tekstur dengan parameter contrast dapat diterapkan dalam klasifikasi jenis bawang dengan akurasi sebesar 83.56%. |
| 3 | Taftyani Yusuf Prahudaya, Agus Harjoko | Metode Klasifikasi Mutu Jambu Biji Menggunakan Knn Berdasarkan Fitur Warna Dan Tekstur[9] | Metode klasifikasi KNN dapat melakukan klasifikasi jambu biji dengan akurasi 91.25%. Fitur yang diekstraks untuk klasifikasi jambu biji adalah r,g,b, luas cacat, energy, homogeinity, dan contrast. |

**BAB III  
ANALISIS DAN PERANCANGAN**

**3.1 Analisis Kebutuhan Fungsional dan Non-Fungsional**

**3.1.1 Kebutuhan Fungsional**

Kebutuhan Fungsional merupakan rangkaian kebutuhan yang secara langsung memiliki keterkaitan dengan sistem. Kebutuhan fungsional dari aplikasi yang dirancang ini adalah sebagai berikut:

1. Sistem dapat mengklasifikasikan jenis permata berdasarkan dari foto yang diinput pengguna.
2. Sistem dapat mengeluarkan output berupa jenis permata.
3. Sistem dapat mengekstraksi dan menginput data foto permata beserta nama jenis nya ke dalam *database*.

**3.1.2 Kebutuhan Non-Fungsional**

Kebutuhan Non Fungsional merupakan rangkaian kebutuhan yang secara tidak langsung memiliki keterkaitan dengan sistem.

1. Perangkat Keras (*Hardware*)

Perangkat Keras (*hardware*) yang digunakan dalam perancangan aplikasi ini adalah sebagai berikut:

1. *Processor* min. *AMD A10-9620P RADEON R5*
2. Memori : 8GB
3. Sistem Operasi: *Microsoft Windows 10*
4. Perangkat Lunak *(Software)*

Perangkat lunak (*software*) yang digunakan dalam perancangan aplikasi ini adalah:

* + - 1. Lazarus IDE versi 2.0.10
      2. XAMPP versi 7.2.5
      3. *Pascal*

1. *Brainware*

*Brainware* merupakan setiap orang yang memiliki keterlibatan dalam perancangan maupun penggunaan aplikasi. Untuk penggunaan penggunaan aplikasi, *brainware* tidak memerlukan keahlian khusus namun setidaknya memiliki kemampuan dasar untuk mengoperasikan komputer dan memiliki pengetahuan dalam menggunakan aplikasi komputer.

**3.2 Pemilihan Algoritma / Teknik / Metode**

**3.2.1 Metode Perancangan Sistem**

Model perancangan sistem yang digunakan dalam membangun atau merancang aplikasi ini menggunakan model *waterfall*. Tujuan digunakan model *waterfall* ini adalah karena proses yang dilakukan secara bertahap, sehingga setiap proses teratur dan tidak tumpang tindih dalam pelaksanaannya.

Tahapan-tahapan yang dilakukan pada perancangan sistem dengan model *waterfall* ini adalah sebagai berikut:

1. Analisis kebutuhan

Untuk membangun aplikasi ini, perlu mengidentifikasi data berupa *input* data fitur warna gambar permata.

1. Desain sistem

Dalam tahap mendesain sistem, dirancang bagaimana tampilan *interface* dari program agar mudah digunakan oleh *user*. Tidak hanya desain *interface* yang dirancang, diagram-diagram yang menggambarkan bagaimana sistem bekerja dan berinteraksi dengan *user* juga dirancang pada tahapan desain sistem berupa *use case diagram* dan *activity diagram*.

1. Penulisan kode program

Bahasa pemrograman yang digunakan dalam merancang aplikasi adalah bahasa pemrograman *Pascal* dan didukung oleh server MySQL. Aplikasi yang digunakan untuk menulis kode program adalah Lazarus IDE*.*

**3.2.2 Pemilhan Algoritma / Metode**

Pada penelitian kali ini, program akan menggunakan dua metode yaitu *thresholding* dan *L1*-*metric.* alasan peneliti menggunakan metode tersebut dikarenakan thresholding dianggap dapat memisahkan antara objek dengan latar belakang gambar. Sedangkan L1-metric dianggap dapat membandingkan kesamaan dua fitur warna dalam mengklasifikasikan jenis permata.

1. *Thresholding*

*Thresholding* digunakan untuk memisahkan antara background dengan objek permata agar program bisa berfokus pada objek permata yang akan kita ambil data histogram nya. Dengan memberi batas warna 127 maka objek akan dinyatakan dengan nilai warna 0 dan background dinyatakan dengan nilai 255. Lalu ditentukan ujung atas, bawah, kiri, dan kanan objek, lalu dipisahkan secara segiempat antara objek dengan background.

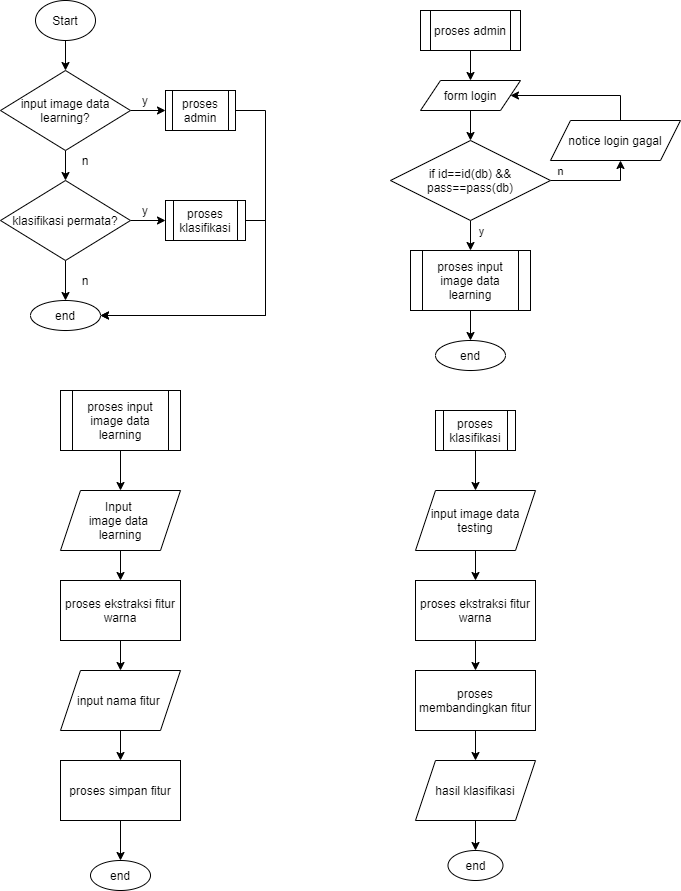
2. *L1-Metric*

*L1-Metric* digunakan untuk membandingkan jarak kesamaan histogram antara dua citra. Nilai jarak batas kesamaan yang akan saya gunakan adalah 300. Semakin kecil nilai jarak yang dihasilkan maka kedua citra akan dianggap semakin mirip.

**3.3 Perancangan Proses**

**3.3.1 *Flowchart* Program**

Perancangan *flowchart* program bertujuan tuntuk menggambarkan proses yang terjadi didalam sistem. *flowchart* program ditunjukkan pada gambar dibawah berikut.

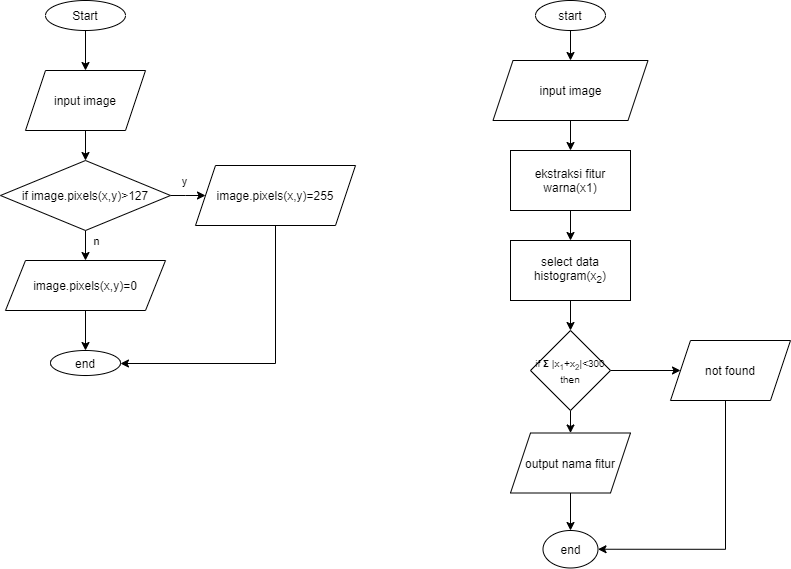
****

Gambar *Flowchart* Program

**3.3.2 *Flowchart* Algoritma**

*Flowchart* algoritma bertujuan untuk menjelaskan bagaimana aplikasi akan memisahkan objek dengan latar belakangnya dan juga melakukan pengukuran jarak kesamaan fitur objek pada citra. *Flowchart* algoritma pada program ini digambarkan pada gambar dibawah berikut.

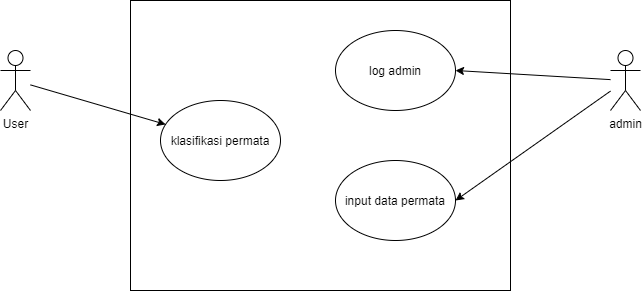
1. (b)



Gambar *Flowchart* Algoritma (a)*Thresholding* (b)*L1*-*metric*

**3.3.3 *Use* *Case* *Diagram***

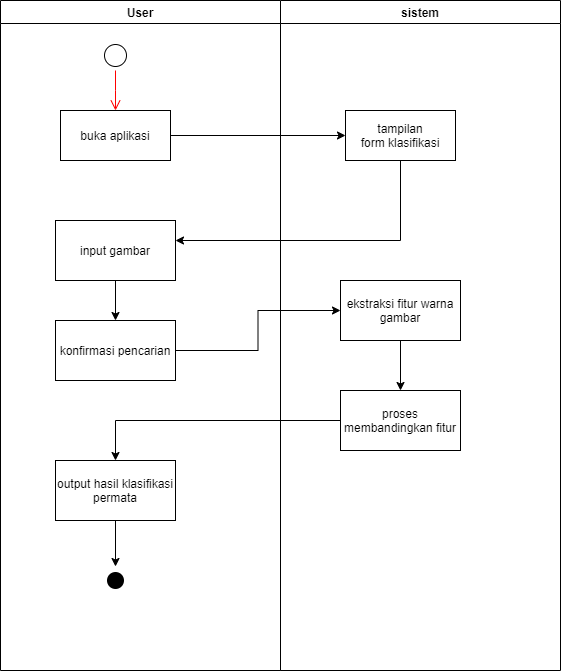
*Use case diagram* bertujuan untuk menggambarkan interaksi antar *user* dengan sistem pengklasifikasian permata ini. *Use case program* penelitian ini dapat digambarkan seperti dibawah berikut.



Gambar *Use Case Diagram*

**3.3.4 *Activity Diagram***

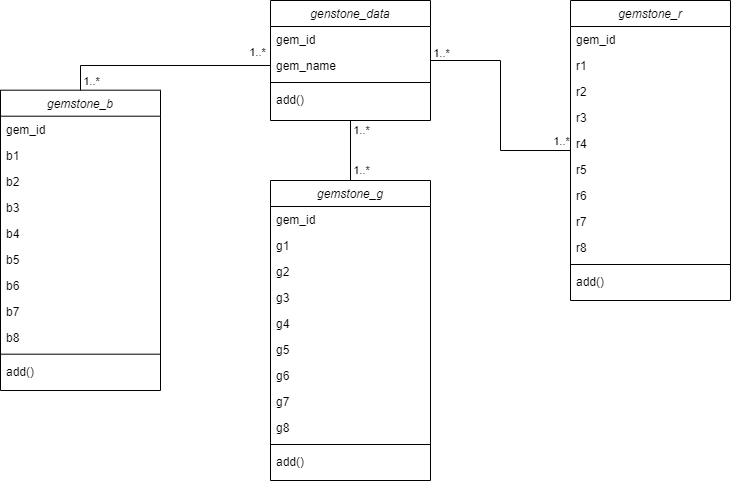
*Activity* *diagram* bertujuan untuk menjelaskan tentang alur kegiatan yang terjadi dalam program dari awal sampai akhir. *Activity* *diagram* pada program dapat dilihat pada gambar dibawah



Gambar *Activity* *Diagram*

**3.3.5 *Class Diagram***

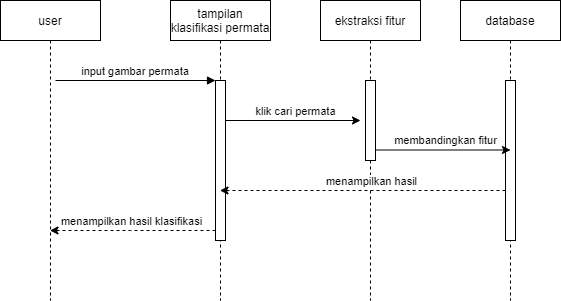
*Class diagram* berguna untuk menjelaskan hubungan *table* - *table* pada *database*.



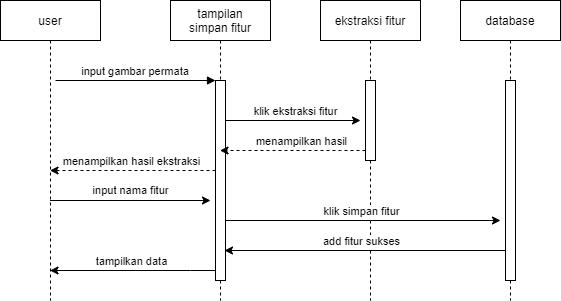
Gambar *Class Diagram*

**3.3.6 *Sequence* *Diagram***

*Sequence diagram* bertujuan untuk menjabarkan kegiatan objek pada *use case* dengan mendeskripsikan waktu hidup objek serta pesan yang dikirimkan dan diterima antar objek.



Gambar *Sequence Diagram* *form* klasifikasi



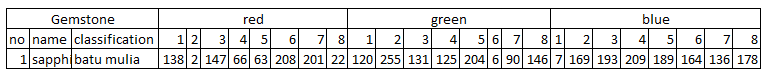
Gambar *Sequence Diagram* *form* simpan fitur

**3.4 Perancangan Basis Data**

**3.4.1 Normalisasi**

Normalisasi Data bertujuan untuk menghilangkan redudansi data didalam sistem yang kami buat dan untuk memastikan dependensi data. Bentuk normalisasi data yang kami buat adalah sebagai berikut.

Un-normalized



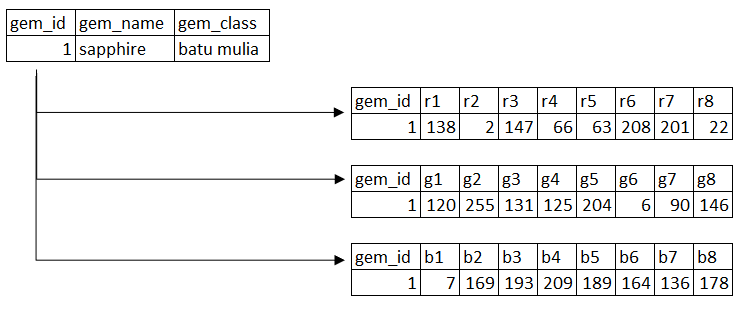
Gambar un-normalized

1NF



Gambar 1NF

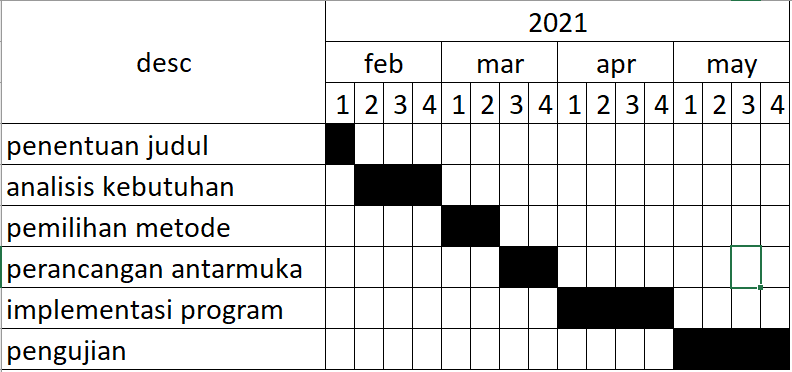
2NF



Gambar 2NF

**3.5 *Gantt* *Chart***

Penjadwalan dalam melakukan segala aktivitas sangatlah menjadi perhatian penting. Jika penjadwalan tidak dilakukan dengan baik, maka hasil yang akan didapatkan tidak akan sesuai dengan harapan. Oleh karena itu, peneliti membuat penjadwalan dalam bentuk *Gantt Chart* untuk perancangan aplikasi seperti terlihat pada Gambar dibawah berikut.

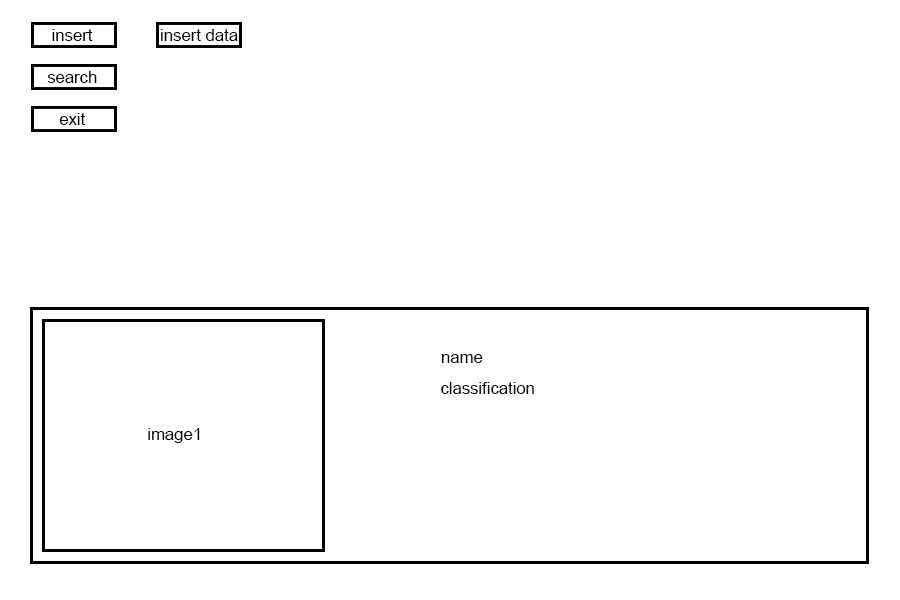


Gambar Gantt Chart

**3.6 Perancangan Tampilan**

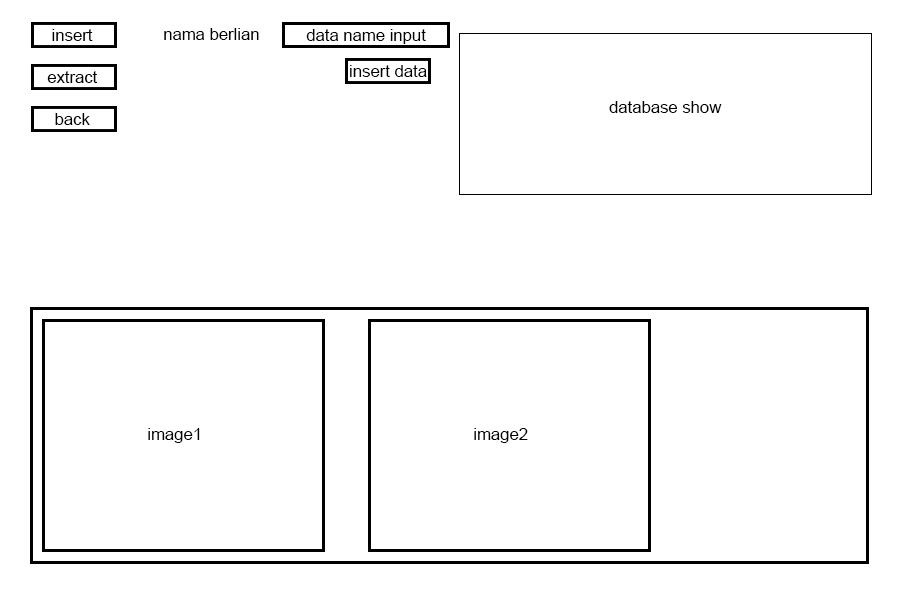
Perancangan tampilan merupakan tahapan untuk mendefinisikan informasi dari desain sehinggam memudahkan kita dalam perencanaan layout program yang akan dikerjakan

**3.6.1 Tampilan Halaman Klasifikasi**

****

Rancangan tampilan pada gambar diatas merupakan tampilan halaman awal pada saat aplikasi dibuka. *Form* ini berfungsi untuk melakukan klasifikasi foto berlian. Terdapat 4 button yang akan digunakan *user*, yaitu *insert* untuk memasukkan foto, *search* untuk melakukan klasifikasi foto berlian, *exit* untuk keluar program, dan *insert* *data* untuk mengarah ke form input data learning foto berlian.

* + 1. **Tampilan Halaman Input Data**

****

Rancangan tampilan pada gambar diatas tampilan ketika admin ingin memasukkan data learning foto berlian ke dalam *database*. Terdapat 2 kolom *image* untuk menampilkan foto sebelum dan sesudah ekstraksi dan satu kolom di pojok atas kanan untuk menampilkan data. Dan juga terdapat *form* input nama berlian yang akan disimpan ke dalam database. Terdapat 4 button yang akan digunakan user, yaitu *insert* untuk memasukkan foto, *extract* untuk melakukan ekstraksi warna foto berlian, *back* untuk Kembali ke form klasifikasi data, dan *insert* *data* untuk memasukkan data learning foto berlian.

**DAFTAR PUSTAKA**

[1] E. Maria, Yulianto, Y. P. Arinda, Jumiaty, and P. Nobel, “Segmentasi Citra Digital Bentuk Daun Pada Tanaman Di Politani Samarinda Menggunakan Metode Thresholding,” *Jurti*, vol. 2, no. 1, pp. 37–46, 2018.

[2] A. Setiawan and K. M. Suryaningrum, “Optical Character Recognition Jepang Menggunakan Matriks Populasi Piksel Dan L1-Metric,” *J. Ilm. Teknol. Inf. Terap.*, vol. IV, no. 1, pp. 62–72, 2017.

[3] S. Bayu Adhi, “PENGEMBANGAN PROGRAM APLIKASI PERMAINAN BAHASA TEKA-TEKI SILANG KOMPUTER MENGGUNAKAN TURBO PASCAL 5.5,” *AKRAB JUARA*, vol. 4, no. 3, p. 30, 2019.

[4] E. Orlando, “Aplikasi Pengajuan Cuti Pada Human Resource Management Menggunakan PHP dan MYSQL (Studi Kasus Pada PT. INTILOKA),” *J. Ilm. KOMPUTASI*, vol. 16, no. 3, pp. 275–284, 2017, [Online]. Available: http://ejournal.jak-stik.ac.id/index.php/komputasi/article/viewArticle/2336.

[5] S. Arief, I. Safi’I, and N. Laela, “Mekanisme Pembuatan Flowchart Penerimaan Mekanisme Pembuatan Flowchart Penerimaan Pinjaman ( Angsuran ) Pada ( Bumdes ) Di Desa,” *J. Abdikarya J. Karya Pengabdi. Dosen dan Mhs.*, vol. 03, no. 03, 2019.

[6] T. A. Gani, “Room Charting Berdasarkan Pemetaan Gantt Chart Pada Penjadwalan Kuliah Pengganti/Tambahan Di Fakultas Teknik Universitas Syiah Kuala,” *J. Karya Ilm. Tek. Elektro*, vol. 4, no. 4, pp. 25–32, 2015.

[7] R. Enggar Pawening, W. Ja, and far Shudiq, “KLASIFIKASI KUALITAS JERUK LOKAL BERDASARKAN TEKSTUR DAN BENTUK MENGGUNAKAN METODE k-NEAREST NEIGHBOR (k-NN),” *Ejournal.Unuja.Ac.Id*, vol. 1, no. 1, pp. 10–17, 2020, [Online]. Available: http://ejournal.unuja.ac.id/index.php/core.

[8] J. A. Widians, H. S. Pakpahan, E. Budiman, H. Haviluddin, and M. Soleha, “Klasifikasi Jenis Bawang Menggunakan Metode K-Nearest Neighbor Berdasarkan Ekstraksi Fitur Bentuk dan Tekstur,” *J. Rekayasa Teknol. Inf.*, vol. 3, no. 2, p. 139, 2019, doi: 10.30872/jurti.v3i2.3213.

[9] T. Y. Prahudaya and A. Harjoko, “Metode Klasifikasi Mutu Jambu Biji Menggunakan Knn Berdasarkan Fitur Warna Dan Tekstur,” *J. Teknosains*, vol. 6, no. 2, p. 113, 2017, doi: 10.22146/teknosains.26972.