

**KLASIFIKASI JENIS BUAH JAMBU BERDASARKAN DAUN  
MENGUNAKAN METODE *PRINCIPAL*  
*COMPONENT ANALYSIS***

**SKRIPSI**

**ATIKAH KHAIRIYAH HASIBUAN  
0701163078**



**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA  
MEDAN  
2020**

**KLASIFIKASI JENIS BUAH JAMBU BERDASARKAN DAUN  
MENGUNAKAN METODE *PRINCIPAL  
COMPONENT ANALYSIS***

**SKRIPSI**

*Diajukan Untuk Memenuhi Syarat Mencapai Gelar Sarjana Komputer*

**ATIKAH KHAIRIYAH HASIBUAN  
0701163078**



**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA  
MEDAN  
2020**

## PERSETUJUAN SKRIPSI

Hal : Surat Persetujuan Skripsi

Lamp : -

Kepada Yth,  
Dekan Fakultas Sains dan Teknologi  
UIN Sumatera Utara Medan

*Assalamu 'alaikum Wr.Wb*

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengatakan perbaikan, maka kami selaku pembimbing berpendapat bahwa skripsi saudara,

Nama : Atikah Khairiyah Hasibuan

Nomor Induk Mahasiswa : 0701163078

Program Studi : Ilmu Komputer

Judul : Klasifikasi Jenis Jambu Berdasarkan daun

Menggunakan Metode *Principal*

*Component Analysis*

Dapat disetujui untuk segera *dimunaqasyahkan*. Atas perhatiannya kami ucapkan terimakasih.

Medan, 23 Maret 2021

4 Syakban 1442

Komisi Pembimbing,

Pembimbing Skripsi I

Pembimbing Skripsi II

Dr. Mhd Furqan, S.Si.,M.Comp.Sc

NIP. 198008062006041003

Muhammad Ikhsan, ST.,M.Kom

NIP. 198304152011011008

## SURAT PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Atikah Khairiyah Hasibuan  
Nomor Induk Mahasiswa : 0701163078  
Program Studi : Ilmu Komputer  
Judul : Klasifikasi Jenis Jambu Beradasarkan daun  
Menggunakan Metode *Principal Component Analysis*

Dengan ini menyatakan bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya. Apabila dikemudian hari ditemukan plagiat dalam skripsi ini maka saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi lainnya sesuai dengan peraturan yang berlaku.

Medan, 23 Maret 2021



Atikah Khairiyah Hasibuan  
0701163078



**KEMENTERIAN AGAMA REPUBLIK INDONESIA  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA MEDAN  
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. IAIN No. 1 Medan 20235  
Telp. (061) 6615683-6622925, Fax. (061) 6615683  
Url: <http://saintek.uinsu.ac.id>, E-mail: [saintek@uinsu.ac.id](mailto:saintek@uinsu.ac.id)

---

**PENGESAHAN SKRIPSI**

Nomor : B.149/ST/ST.V.2/PP.01.1/08/2021

Judul : Klasifikasi Jenis Buah Jambu Berdasarkan Daun  
Menggunakan Metode *Principal Component Analysis*  
Nama : Atikah Khairiyah Hasibuan  
Nomor Induk Mahasiswa : 0701163078  
Program Studi : Ilmu Komputer  
Fakultas : Sains dan Teknologi

Telah dipertahankan di hadapan Dewan Penguji Skripsi Program Studi Ilmu Komputer  
Fakultas Sains dan Teknologi UIN Sumatera Utara Medan dan dinyatakan **LULUS**.

Pada hari/tanggal : Selasa, 23 Maret 2021  
Tempat : Via Zoom Meeting

Tim Ujian Munaqasyah,  
Ketua,

Ilka Zufria, M.Kom  
NIP. 198506042015031006

Dewan Penguji,

Penguji I,

Penguji II,

Dr. Mhd. Furqan, S.Si., M.Comp.Sc.  
NIP. 198008062006041003

Muhammad Ikhsan, S.T. M.Kom.  
NIP. 198304152011011008

Penguji III,

Penguji IV,

Armansyah, M.Kom  
NIB. 1100000074

Sriani, M.Kom.  
NIB. 1100000108

Mengesahkan,  
Dekan Fakultas Sains dan Teknologi  
UIN Sumatera Utara Medan,

Dr. Mhd.Syahnan, M.A  
NIP: 196609051991031002

## ABSTRAK

Perkembangan teknologi untuk teknik pengolahan citra berkembang pesat. Berbagai teknik dikembangkan untuk pekerjaan manusia, baik sebagai pengolahan citra, analisis citra maupun berbagai tujuan dan keperluan. Salah satunya adalah morfologi tumbuhan yang mempelajari bentuk dan susunan tubuh tumbuhan. *Feature Extraction* atau ekstraksi ciri merupakan teknik yang bertujuan untuk mengklasifikasikan citra yang telah diolah sebelumnya berdasarkan kesamaan atau kemiripan ciri yang dimilikinya. Dengan banyaknya jenis dan bentuk tumbuhan maka dapat diklasifikasikan berdasarkan kesamaan cirinya. Klasifikasi berdasarkan bentuk daun memiliki peranan penting dalam pengenalan jenis-jenis tumbuhan dikarenakan keanekaragaman tumbuhan yang beraneka ragam. Salah satu contoh tumbuhan daun yang beraneka ragam jenisnya adalah daun jambu. Dalam penelitian ini dibuat Klasifikasi Jenis Jambu Berdasarkan Bentuk Daun. Metode yang digunakan pada penelitian ini adalah *Principle Component Analysis* (PCA) dan *K-Nearest Neighbor* (K-NN). Proses PCA digunakan untuk proses ekstraksi fitur yaitu untuk mendapatkan ciri daun tersebut, sedangkan K-NN digunakan untuk proses klasifikasi citra daun. Masing-masing nilai bobot citra latih dan citra uji akan dibandingkan dengan meminimalkan nilai *Euclidean Distance*. Pada penelitian ini menggunakan 4 jenis buah jambu. Berdasarkan dari hasil uji akurasi, didapatkan nilai akurasi sebesar 81,25% untuk proses klasifikasi jenis jambu berdasarkan citra daun jambu sebanyak 16 data.

**Kata Kunci** : Daun, *Principal Component Analysis* (PCA), *K-Nearest Neighbor* (K-NN).

## ABSTRACT

Technological developments for image processing techniques are growing fast. Various techniques are developed for human work, both as image processing, image analysis and various other purposes. One of them is plant morphology which studies the shape and composition of the plant body. Feature Extraction or feature extraction is a technique that aims to classify previously processed images based on the similarity of the features they have. With so many types and forms of plants, it can be classified based on their similarities. Classification based on leaf shape has an important role in the introduction of plant types due to the diversity of plants. One example of various types of leaf plants is guava leaves. In this study, the Classification of Guava Types Based on the Shape of the Leaves was made. The method used in this research is Principle Component Analysis (PCA) and K-Nearest Neighbor (K-NN). The PCA process is used for the feature extraction process, namely to obtain the leaf characteristics, while the K-NN is used for the leaf image classification process. Each weight value of the training image and test image will be compared by minimizing the value of Euclidean Distance. This study used 4 types of guava fruit with 5 images in each type. Based on the result of the accuracy test, an accuracy value of 81,25% was obtained for the classification process of the guava types based on 16 guava leaf images.

**Keyword** : *Leaf, Principal Component Analysis (PCA), K-Nearest Neighbor (K-NN).*

## KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah, puji syukur kehadiran Allah SWT, yang telah melimpahkan rahmat, hidayah serta karunia-Nya dan menganugerahkan petunjuk untuk memudahkan dan melancarkan sehingga penulis dapat menyelesaikan skripsi ini dengan judul **“Klasifikasi Jenis Buah Jambu Berdasarkan Daun Dengan Menggunakan Metode *Principal Componets Analysis*”**. Tak lupa juga sholawat serta salam semoga senantiasa tercurahkan kepada uswatun hasanah kita, Nabi Muhammad SAW beserta keluarga, para sahabat dan pengikutnya termasuk kita semua yang senantiasa menantikan syafa'atnya kelak di hari akhir.

Demi kelancaran dalam penyelesaian skripsi ini tidak terlepas dari bantuan pihak terutama kepada Ibu *Suliyanti* yang telah memberikan bantuan moril maupun materil, semangat dan do'a yang begitu besar kepada penulis.

Penulis menyadari bahwa tersusunnya skripsi ini atas do'a, perhatian, bantuan, bimbingan, motivasi serta dukungan dari berbagai pihak sehingga dengan keikhlasan dan kerendahan hati pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Prof. Dr. Syahrin Harahap, MA selaku Rektor Universitas Islam Negeri Sumatera Utara.
2. Bapak Dr. Mhd. Syahnan, M.A, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara.
3. Bapak Ilka Zufria, M.Kom selaku Ketua Jurusan Ilmu Komputer.
4. Bapak Dr. Mhd Furqan, S,Si., M.Comp.Sc. selaku dosen pembimbing skripsi I yang telah berkontribusi membantu penulis dalam memberikan ide, saran, kritik, dan bimbingannya kepada penulis selama penulis mengerjakan skripsi ini.
5. Bapak Muhammad Ikhsan, ST.,M.Kom selaku dosen pembimbing skripsi II yang telah berkontribusi membantu penulis dalam memberikan ide, saran, kritik dan bimbingannya kepada penulis selama penulis mengerjakan skripsi ini.



6. Bapak Rakhmat Kurniawan, M.Kom selaku dosen pembimbing Akademik yang telah berkontribusi membantu penulis dalam memberikan ide, saran, kritik dan bimbingannya kepada penulis selama penulis mengerjakan skripsi ini.
7. Seluruh tenaga pengajar dan pegawai program studi S1 Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara.
8. Teman-teman kelas Ilmu komputer 3 yang selalu memberikan dukungan serta arahan kepada penulis.
9. Kepada adik kandung penulis, *Rosi Khairunnisa* terima kasih untuk dukungan, doa dan semangatnya yang selalu diberikan untuk penulis.
10. Dan semua pihak yang telah membantu penulis namun tidak dapat disebutkan satu persatu.

Penulis menyadari sepenuhnya bahwa dalam pembuatan skripsi ini masih jauh dari kesempurnaan, untuk karena itu kritik dan saran yang bersifat membangun sangat penulis harapkan dari pembaca. Semoga hasil skripsi ini menjadi ibadah bagi penulis dan bermanfaat bagi para pembaca. Aamin Ya Rabbal'alam.

Medan, 18 Maret 2021

Hormat saya,

Atikah Khairiyah Hasibuan

## DAFTAR ISI

<b>ABSTRAK</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>KATA PENGANTAR</b> .....	<b>iii</b>
<b>DAFTAR ISI</b> .....	<b>v</b>
<b>DAFTAR GAMBAR</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>ix</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>x</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	4
1.5 Manfaat Penelitian .....	4
<b>BAB II TINJAUAN PUSTAKA</b> .....	<b>5</b>
2.1 Daun Jambu .....	5
2.2 Citra Digital.....	7
2.3 Pengolahan Citra .....	12
2.4 Ekstrasi Fitur .....	11
2.5 <i>Principal Component Analysis</i> .....	13
2.6 Klasifikasi .....	18
2.7 <i>K-Nearest Neighbor</i> .....	18
2.8 <i>Euclidean Distance</i> .....	20
2.9 Akurasi.....	21
2.10 Penelitian Terkait.....	21
<b>BAB III METODOLOGI PENELITIAN</b> .....	<b>24</b>
3.1 Tempat dan Waktu Penelitian .....	24
3.1.1 Tempat Penelitian .....	24
3.1.2 Waktu Penelitian.....	24

3.2 Bahan dan Alat Penelitian.....	25
3.2.1 Perangkat Keras.....	25
3.2.2 Perangkat Lunak.....	25
3.3 Prosedur Kerja.....	25
3.3.1 Teknik Pengumpulan Data.....	26
3.3.2 Analisa Kebutuhan .....	27
3.3.3 Perancangan .....	28
3.3.4 Pengujian.....	32
3.3.5 Penerapan / Penggunaan .....	32
<b>BAB IV PEMBAHASAN DAN HASIL.....</b>	<b>33</b>
4.1 Pembahasan.....	33
4.1.1 Analisis Data.....	33
4.1.2 Representasi Data.....	36
4.1.3 Hasil Analisis.....	43
4.1.4 Perancangan Sistem .....	47
4.2 Hasil.....	56
4.2.1 Pengujian .....	56
4.2.2 Penerapan .....	72
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>73</b>
5.1 Kesimpulan .....	73
5.2 Saran .....	73
<b>DAFTAR PUSTAKA.....</b>	<b>74</b>
<b>LAMPIRAN-LAMPIRAN</b>	

## DAFTAR GAMBAR

Gambar	Judul Gambar	Halaman
2.1	Daun Jambu Biji .....	5
2.2	Daun Jambu Mete .....	6
2.3	Daun Jambu Bol .....	6
2.4	Daun Jambu Air.....	7
2.5	Citra Biner .....	8
2.6	Citra <i>Grayscale</i> .....	8
2.7	Skema Kubus Warna RGB .....	9
2.8	Skema Kubus Warna RGB .....	10
2.9	Citra Berwarna .....	10
3.1	Diagram Prosedur Kerja .....	25
3.2	Diagram Blok Model Sistem.....	29
3.3	<i>Flowchart</i> Proses Metode PCA.....	30
3.4	<i>Flowchart</i> Proses Metode KNN.....	31
4.1	Daun Jambu Mete Dan Biji.....	33
4.2	Nilai <i>Pixel</i> 3x3 Citra Jambu A .....	34
4.3	Nilai <i>Pixel</i> 3x3 Citra Jambu B.....	35
4.4	Nilai <i>Pixel</i> Citra Jambu Uji.....	44
4.5	Rancangan Menu Utama.....	47
4.6	Rancangan Klasifikasi .....	47
4.7	Rancangan Menu Info Penulis .....	48
4.8	<i>Flowchart</i> Menu Utama.....	51
4.9	<i>Flowchart</i> Klasifikasi .....	52
4.10	<i>Flowchart</i> Info Penulis .....	53
4.11	Tampilan Menu Utama .....	54
4.12	Tampilan Menu Klasifikasi.....	54
4.13	Tampilan Menu Info Penulis.....	55
4.14	Citra Daun Jambu Latih .....	56

4.15	Citra Daun Jambu Uji .....	56
4.16	Menu Proses Klasifikasi .....	57
4.17	Menu Proses Pencarian Citra .....	57
4.18	Citra Daun Jambu Dipilih .....	58
4.19	Citra Daun Jambu 1 Diklasifikasi .....	58
4.20	Citra Daun Jambu 2 Diklasifikasi .....	59
4.21	Citra Daun Jambu 3 Diklasifikasi .....	59
4.22	Citra Daun Jambu 4 Diklasifikasi .....	60
4.23	Citra Daun Jambu 1 Diklasifikasi .....	61
4.24	Citra Daun Jambu 2 Diklasifikasi .....	61
4.25	Citra Daun Jambu 3 Diklasifikasi .....	62
4.26	Citra Daun Jambu 4 Diklasifikasi .....	62
4.27	Citra Daun Jambu 1 Diklasifikasi .....	63
4.28	Citra Daun Jambu 2 Diklasifikasi .....	64
4.29	Citra Daun Jambu 3 Diklasifikasi .....	64
4.30	Citra Daun Jambu 4 Diklasifikasi .....	65
4.31	Citra Daun Jambu 1 Diklasifikasi .....	66
4.32	Citra Daun Jambu 2 Diklasifikasi .....	66
4.33	Citra Daun Jambu 3 Diklasifikasi .....	67
4.34	Citra Daun Jambu 4 Diklasifikasi .....	67

## DAFTAR TABEL

<b>Tabel</b>	<b>Judul Tabel</b>	<b>Halaman</b>
3.1	Waktu Penelitian.....	24
4.1	Hasil <i>Pixel 3x3 Grayscale Sample A</i> .....	34
4.2	Hasil <i>Pixel 3x3 Grayscale Sample B</i> .....	35
4.3	Nilai Variabel A Dan B.....	36
4.4	Rata-Rata Nilai Variabel A Dan B .....	37
4.5	Nilai Matriks Selisih Variabel A Dan B .....	38
4.6	Nilai Keseluruhan Varian Var A Dan B .....	39
4.7	Niali Rata-Rata Varian Var A Dan B .....	40
4.8	Dataset PCA .....	44
4.9	Hasil <i>Pixel 3x3 Grayscale Sample Uji</i> .....	45
4.10	Nilai Citra Data Uji.....	46
4.11	Nilai Keseluruhan Data Klasifikasi .....	46
4.12	Data Hasil Perhitungan Jarak Terdekat.....	47
4.13	Informasi Citra Latih Dan Uji .....	58
4.14	Hasil Pengujian.....	71

## DAFTAR LAMPIRAN

Lampiran	Judul Lampiran
1.	Hasil uji citra daun jambu pada “Klasifikasi Jenis Jambu Berdasarkan Daun Menggunakan Metode <i>Principal Component Analysis</i> ”
2.	<i>Tools</i> Matlab untuk “Klasifikasi Jenis Jambu Berdasarkan Bentuk Daun Menggunakan Metode <i>Principal Component Analysis</i> ”
3.	Kode Program Matlab untuk “Klasifikasi Jenis Jambu Berdasarkan Bentuk Daun Menggunakan Metode <i>Principal Component Analysis</i> ”
4.	Kartu Bimbingan Skripsi
5.	Riwayat Hidup

# BAB 1

## PENDAHULUAN

### 1.1 LATAR BELAKANG

Allah SWT menggambarkan peristiwa yang dialami manusia setiap hari dalam rangka memahami ilmu, kekuasaan dan hikmah Allah. Allah menjelaskan, Dialah yang membawa hujan dari langit dan memberi kehidupan pada berbagai jenis tanaman dengan bentuk yang berbeda jenis dan rasa yang berbeda-beda. Salah satunya ada dalam Al-Qur'an (Al-An'am : 99).

وَهُوَ الَّذِي أَنْزَلَ مِنَ السَّمَاءِ مَاءً فَأَخْرَجْنَا بِهِ نَبَاتَ كُلِّ شَيْءٍ فَأَخْرَجْنَا مِنْهُ خَضِرًا نُخْرِجُ مِنْهُ حَبًّا مُتَرَاكِبًا وَمِنَ النَّخْلِ مِنْ طَلْعِهَا قِنْوَانٌ دَانِيَةٌ وَجَنَّاتٍ مِّنْ أَعْنَابٍ وَالزَّيْتُونَ وَالرُّمَّانَ مُشْتَبِهًا وَغَيْرَ مُتَشَبِهٍ انظُرُوا إِلَى ثَمَرِهِ إِذَا أَثْمَرَ وَيَنْعِهِ إِنَّ فِي ذَلِكَ لَآيَاتٍ لِّقَوْمٍ يُؤْمِنُونَ ٩٩

*Artinya : Dan Dialah yang menurunkan air hujan dari langit, lalu Kami tumbuhkan dengan air itu segala macam tumbuh-tumbuhan maka Kami keluarkan dari tumbuh-tumbuhan itu tanaman yang menghijau. Kami keluarkan dari tanaman yang menghijau itu butir yang banyak; dan dari mayang korma mengurai tangkai-tangkai yang menjulai, dan kebun-kebun anggur, dan (Kami keluarkan pula) zaitun dan delima yang serupa dan yang tidak serupa. Perhatikanlah buahnya di waktu pohonnya berbuah dan (perhatikan pulalah) kematangannya. Sesungguhnya pada yang demikian itu ada tanda-tanda (kekuasaan Allah) bagi orang-orang yang beriman. (QS An An'am:99).*

Dalam hal ini, Sang Pencipta Allah SWT telah menyediakan berbagai macam tumbuhan yang dapat sangat bermanfaat bagi manusia. Tuhan mengirimkan air hujan dari langit, dengan air hidup dari semua jenis tumbuhan dari hijau tanaman, menarik banyak buah dari satu biji.

Perkembangan teknologi pengolahan citra semakin pesat. Baik sebagai pengolahan citra, analisis citra, maupun pengguna citra untuk maksud dan tujuan yang berbeda, berbagai teknik telah dikembangkan untuk memudahkan pekerjaan manusia. Salah satunya adalah morfologi tumbuhan, yang mempelajari bentuk



dan struktur tumbuhan. Klasifikasi berdasarkan bentuk daun memegang peranan penting dalam pengenalan jenis tumbuhan karena keanekaragaman tumbuhan. Contoh tumbuhan berdaun beraneka ragam adalah daun jambu biji. Daun jambu biji unik, mudah diamati, dan salah satu ciri tumbuhan yang paling khas. Daun jambu biji bervariasi dalam bentuk dan ukuran serta menunjukkan keragaman genetik yang cukup luas.

Pengenalan daun merupakan salah satu teknologi pengembangan untuk visi komputer dan pengenalan pola. Tujuan utamanya adalah untuk mengklasifikasikan citra berdasarkan kesamaan atau kemiripan karakteristik citra tersebut. Karena ada banyak jenis dan bentuk daun, mereka dapat diklasifikasikan berdasarkan kesamaan karakteristiknya. Tujuan dari penelitian ini adalah adalah klasifikasi buah jambu biji sesuai daunnya. Metode yang digunakan dalam penelitian ini adalah metode Principal Component Analysis dan K-Nearest Neighbor. Principal Component Analysis (PCA) digunakan dalam proses ekstraksi ciri, yaitu untuk mendapatkan karakteristik dari citra daun. Sebuah metode yang disebut K-NN digunakan untuk mengklasifikasikan citra daun. Hasilnya adalah sistem berbasis *desktop* yang dapat digunakan untuk mengidentifikasi buah-buahan menggunakan gambar buah itu sendiri.

Algoritma *Principal Component Analysis* (PCA) merupakan salah satu metode yang tepat untuk proses pengenalan karena dianggap mudah untuk mengekstrak informasi yang berkaitan dengan sekumpulan data. Metode PCA bekerja dengan cara mereduksi jumlah dimensi dari variabel input menjadi satu dimensi untuk komponen utama sebagai identitas. Pada penelitian ini akan digunakan metode *K-Nearest Neighbor* (K-NN) untuk mengklasifikasikan citra daun. Proses klasifikasi didasarkan pada karakteristik bentuk tepi daun. Metode K-NN mengklasifikasikan objek berdasarkan data latih dan objek terdekat. Tahap pelatihan adalah mengekstrak hasil koleksi berbagai daun, kemudian mengumpulkan dan menyimpannya dalam *database*. Hasil ekstraksi citra menggunakan algoritma PCA dikategorikan menggunakan metode K-NN dan dibandingkan dengan citra daun baru. Citra daun menentukan kemiripan objek

dan mengklasifikasikan jarak berdasarkan jarak. Objek yang berdekatan dengan jarak *Euclidean*.

Uraian diatas merupakan latar belakang penulis untuk mengangkat judul : **“Klasifikasi Jenis Buah Jambu Berdasarkan Daun Menggunakan Metode *Principal Analysis Components*”**.

## 1.2 RUMUSAN MASALAH

Dari penjelasan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana menerapkan metode *Principal Component Analysis* dalam hal ekstrasi ciri daun jambu berdasarkan bentuk daun?
2. Bagaimana menerapkan metode *K-Nearest Neighbor* untuk mengklasifikasikan jenis jambu biji berdasarkan bentuk daunnya?
3. Berapa tingkat keakuratan klasifikasi daun jambu dengan menggunakan *Principal Component Analysis* dan *K-Nearest Neighbor*?

## 1.3 BATASAN MASALAH

Agar pembahasan dalam penelitian ini tidak terlalu luas, penelitian ini dibatasi pada permasalahan sebagai berikut:

1. Program dibuat menggunakan Matlab R2012b.
2. Hanya 4 jenis daun jambu yang digunakan dalam penelitian.
3. Gambar yang diambil adalah daun jambu bagian permukaan.
4. Citra yang digunakan dengan format jpg.
5. Metode pengolahan citra yang akan digunakan adalah *Principal Component Analysis* (PCA) untuk mendapatkan ekstrasi ciri dan untuk klasifikasi menggunakan metode *K-Nearest Neighborhood* (K-NN)
6. Proses pencocokan citra menggunakan metode *Euclidean Distance*.

#### 1.4 TUJUAN PENELITIAN

Tujuan dilakukannya penelitian ini adalah sebagai berikut:

1. Untuk menerapkan metode *Principal Components Analysis* sebagai ekstraksi ciri citra daun jambu.
2. Untuk menerapkan metode *K-Nearest Neighbor* untuk mengklasifikasi jenis buah jambu berdasarkan daun.
3. Mengetahui tingkat keakuratan klasifikasi jenis buah jambu berdasarkan daun dengan menggunakan *Principal Components Analysis* dan *K-Nearest Neighbor*.

#### 1.5 MANFAAT PENELITIAN

Manfaat yang didapat dari penelitian ini adalah sebagai berikut:

1. Untuk mengetahui bagaimana proses penerapan metode *Principal Component Analysis* dan *K-Nearest Neighbor* dalam proses penerapan klasifikasi jambu berdasarkan bentuk daun.
2. Sebagai referensi tambahan dalam penelitian analisis pola daun.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Daun Jambu**

Daun tanaman merupakan salah satu biometric dan karakteristik tanaman. Hal ini dikarenakan bentuk daun dan tulang daun berbeda-beda tergantung dari jenis tanamannya. Struktur tulang daun merupakan salah satu ciri khas daun yang berperan penting dalam mengklasifikasikan jenis tumbuhan melalui serangkaian proses pencitraan (Wahyumianto, 2009).

##### **a. Daun Jambu Biji**

Tanaman jambu biji (*Psidium guajava*) berdaun tunggal dengan bentuk lonjong, tepi tumpul, bagian bawah membulat, dan tepi rata. Daun jambu biji memiliki panjang 6-14 cm dan lebar 3-6 cm. Daun ini memiliki warna hijau kekuning-kuningan dan memiliki daun berduri panjang. (Ide, 2011). Daunnya berbentuk lonjong, dengan tepi atas dan bawah melengkung ke atas. Buahnya bulat sampai lonjong, merah, hijau sampai kuning-hijau (Tanri, 2003). Buah pohon ini sangat lebat dan terdapat di tengah pohon, dengan biji yang banyak



**Gambar 2.1** Daun Jambu Biji

##### **b. Daun Jambu Mete**

Tanaman jambu mete (*Anacardium occidentale L*) merupakan tanaman asli Brazil. Daun jambu mete umumnya berbentuk lonjong, membulat pada ujungnya dan meruncing pada pangkalnya. Daun jambu biji

Jambu mete adalah daun yang tidak dilipat. Warna daun jambu mete bervariasi dari hijau tua hingga coklat kemerahan. Panjang daun umumnya mencapai 10-12 cm dan lebarnya 5-10 cm. (Suprati, 2004).



**Gambar 2.2** Daun Jambu Mete

c. Daun Jambu Bol

Jambu biji bol (*Syzygium malaccense*) juga dikenal sebagai jambu dersono. Jambu biji adalah daun sederhana, terletak di sisi berlawanan, dengan batang pendek 1-15 cm. Daun lonjong besar, menonjol dan tebal agak keras seperti kulit binatang. (Soeharso, 2011).



**Gambar 2.3** Daun Jambu Bol

d. Daun Jambu Air

Jambu air (*Syzygium Samarangense*) merupakan tanaman asli suku Jambu Biji di Indonesia. Daun tunggal berhadapan, batang pendek dan tebal,

panjang 3-5mm. Daunnya lonjong, berukuran 10-25 x 5-12 cm, bermata tipis, tembus cahaya, berbintik dan beraroma bila diremas (Handaya, 2013).



**Gambar 2.4** Daun Jambu Air

## **2.2 Citra Digital**

Citra merupakan hasil keseluruhan dari sistem perekaman data. Citra adalah gambaran, rupa, atau tiruan dari sesuatu. Gambar digital adalah representasi objek yang dipecah menjadi lokasi dan warna yang berbeda. Sebuah citra digital dapat direpresentasikan dengan matriks  $M$  baris dan  $N$  kolom. Suatu matriks memiliki baris dan kolom. Baris dan kolom berisi elemen gambar. Piksel menyimpan informasi dalam bentuk intensitas warna pada koordinat tersebut.

### **2.2.1 Jenis-Jenis Citra**

Berdasarkan ukuran pikselnya, citra dapat dibedakan menjadi beberapa jenis (Putra, 2010), yaitu :

#### **a. Citra Biner**

Gambar biner sering digunakan dalam program komputer untuk memproses gambar, seperti operasi morfologi, mengambang, dan segmentasi. Citra biner diwakili oleh dua tingkat abu-abu, hitam dan putih. Gambar biner hanya membutuhkan satu bit untuk mewakili setiap piksel. (McAndrew, 2004).



**Gambar 2.5** Citra Biner

**b. Citra *Grayscale* (Skala Keabuan)**

Citra *grayscale* adalah Gambar digital dengan satu saluran per piksel disebut skala abu-abu. Dengan kata lain, merah, hijau, dan sebagian biru. Nilai ini digunakan untuk menunjukkan tingkat intensitas. Warna yang mereka miliki adalah hitam, abu-abu, dan putih. Tingkat keabuan yang dimaksud adalah tingkat keabuan yang berbeda-beda, dari hitam hingga mendekati putih. Kedalaman warna gambar grayscale adalah 8 bit (256 kombinasi warna abu-abu) (McAndrew, 2004).



**Gambar 2.6** Citra *Grayscale*

Untuk menghitung nilai rata-rata piksel suatu citra RGB, nilai RGB setiap piksel diubah menjadi nilai skala keabuan dengan mencari nilai rata-rata dari masing-masing piksel.

$$f(x, y) = \frac{fR(x,y)+fG(x,y)+fB(x,y)}{3} \quad (1)$$

dengan:

$fR$  adalah nilai komponen *red*

$fG$  adalah nilai komponen *green*

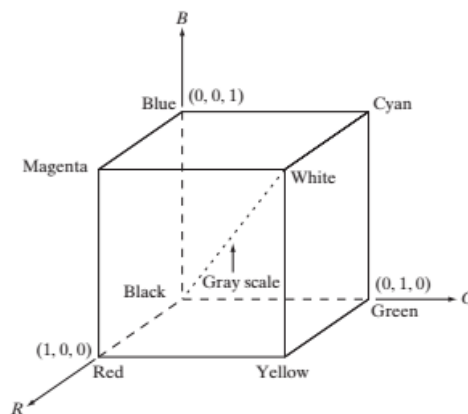
$fB$  adalah nilai komponen *blue*

Berdasarkan persamaan 1, maka dapat diambil contoh perhitungan nilai *grayscale*:

$$f(1,1) = (255, 240, 240) = (255 + 240 + 240) / 3 = 198$$

### c. Citra Berwarna (*True Color*)

Setiap gambar berwarna terdiri dari tiga lapisan warna yang terpisah, yaitu merah, hijau, dan biru. Setiap piksel warna diwakili oleh sejumlah bit misalnya gambar berwarna (24 bit), setiap piksel warna diwakili oleh 24 bit. Perubahan warna secara keseluruhan ini diyakini melebihi penglihatan manusia pada umumnya, yang diyakini hanya mampu membedakan sekitar 10 juta warna.



**Gambar 2.7 Skema Kubus Warna RGB**

Sumber (Gonzalez & Woods, 2008)





**Gambar 2.8 Skema Kubus warna RGB 24 bit**

Sumber (Gonzalez & Woods, 2008)

Dari Gambar 2.8 terlihat adanya warna lain selain warna tiga lapis. Warna-warna ini diperoleh dengan mengambil nilai elemen tiga lapisan dan menggabungkannya. Rentang setiap elemen adalah 0-255, dan kedalaman warna setiap elemen adalah 8 bit. Karena citra RGB memiliki 3 lapisan, maka jumlah kombinasi warna yang dapat dihasilkan pada citra RGB adalah  $256 \times 256 \times 256 = 16.777.216$ .



**Gambar 2.9 Citra Berwarna**

### **2.2.2 Operasi Dasar Citra Digital**

Karena citra digital merupakan matriks piksel, maka operasi yang terjadi pada citra digital pada dasarnya adalah operasi pada elemen matriks. Ada beberapa operasi dasar pada citra digital, salah satunya adalah dapat ditransformasikan secara geometris. Operasi geometri dalam

pemrosesan gambar mengubah koordinat piksel gambar dalam beberapa cara, dan nilai abu-abu dapat diubah seiring kemajuannya. Operasi geometri berkaitan dengan perubahan bentuk gambar, antara lain: *flip*, *rotate*, *crop*, dan *scaling*.

### **2.2.3 Citra Digital Berformat JPEG (Joint Photographic Expert Group)**

JPEG atau JPEG adalah standar universal untuk format file gambar. Kamera digital dan sistem pencitraan terkomputerisasi dapat membuat gambar dari format file DPOF. JPEG mendukung kedalaman warna 24-bit (gambar RGB memiliki 3 lapisan, setiap lapisan memiliki kedalaman warna 8 bit). Format file JPEG lebih kecil dari format file TIFF, tetapi algoritma kompresi lossy menghasilkan kualitas gambar yang buruk. File yang dikompresi dengan kompresi lossy lebih kecil dari aslinya. (McAndrew, 2004).

## **2.3 Pengolahan Citra (*Image Processing*)**

Proses pengolahan citra adalah untuk menghasilkan hasil citra yang diinginkan dengan menggunakan prosedur dan algoritma. Pengolahan citra biasanya dilakukan untuk perbaikan citra, kompresi citra, analisis citra, pengelompokan citra, dan lain-lain. :

### **a. Peningkatan Gambar**

Tujuan dari operasi ini adalah untuk meningkatkan tampilan gambar yang diambil oleh kamera, dengan membuatnya lebih jelas. Operasi ini biasanya menyoroti fitur gambar.

### **b. Kompresi Gambar**

Kompresi gambar dirancang untuk mengurangi jumlah data dalam gambar sehingga dapat ditampilkan tanpa menggunakan semua memori di komputer.

c. Segmentasi citra

Segmentasi citra bertujuan untuk membagi citra menjadi beberapa segmen sesuai dengan kriteria tertentu.

d. Analisis Citra

Jenis operasi ini ditujukan untuk menghitung besaran kuantitatif dari citra untuk menggambarannya. Teknik analisis citra mencari karakteristik khusus yang membantu mengidentifikasi objek.

## 2.4 Ekstraksi Fitur

Ekstraksi fitur adalah tindakan mengambil fitur dari suatu bentuk yang nantinya nilai yang diperoleh akan dianalisis menggunakan proses selanjutnya. Ekstraksi fitur dilakukan dengan menghitung jumlah titik atau piksel yang terdeteksi oleh setiap pemeriksaan. Di sini, analisis gambar digital terjadi dalam arah yang berbeda, mengikuti koordinat Cartesian gambar. Fitur adalah karakteristik unik dari suatu objek. Fungsi dapat dibagi menjadi dua. "Alami" adalah salah satu atribut gambar. Selain itu, fungsi "buatan" adalah fungsi yang diperoleh dengan operasi tertentu pada gambar, seperti histogram tingkat abu-abu. Ekstraksi fitur adalah proses untuk mendapatkan fitur yang membedakan suatu objek dengan objek lainnya. (Putra, 2010).

a. Ciri Warna

Model warna adalah model matematika standar yang dapat digunakan untuk menggambarkan berbagai komponen warna. Warna dapat dipecah menjadi beberapa warna. Warna yang paling umum adalah warna model warna RGB.

b. Ciri Bentuk

Bentuk menggambarkan posisi, orientasi, dan ukuran objek. Ciri morfologi yang umum biasanya tiga dimensi dan ciri morfologi yang sering dibentuk oleh mata adalah gambar dua dimensi (Muslim, 2013). Bentuk dalam gambar dapat membantu mendeteksi batas objek dan wilayah.

### c. Ciri Tekstur

Tekstur adalah fitur visual yang dapat menentukan sifat-sifat suatu gambar. Hal ini dikarenakan tekstur mengandung informasi penting tentang komposisi struktur permukaan gambar.

## 2.5 *Principal Component Analysis (PCA)*

*Principal Component Analysis* merupakan salah satu metode yang digunakan untuk mengidentifikasi pola pada data. PCA adalah algoritma untuk menemukan komponen yang berguna dari data yang lebih kompleks, sering digunakan untuk klasifikasi dan kompresi data. Metode PCA bekerja dengan mengubah variabel independen awal menjadi variabel independen baru yang tidak memiliki korelasi sama sekali untuk menghilangkan korelasi antar variabel independen. Variabel baru ini berisi nilai variabel komponen utama (PC). Program PCA dirancang untuk menyederhanakannya dengan mengurangi dimensi variabel yang diamati. (Abdi, 2010). Salah satu kelebihan metode PCA adalah dapat mengurangi jumlah waktu yang dibutuhkan untuk memproses data. Metode PCA melibatkan nilai eigen dari matriks kovarians dalam perhitungannya. Analisis komponen utama juga dikenal sebagai transformasi *Karhunen-Love*. PCA adalah metode statistik yang digunakan untuk pengenalan citra.

Analisis komponen utama atau prinsipal komponen teknik statistik mengubah sebagian besar variabel asli yang digunakan dan saling terkait menjadi satu set variabel baru yang lebih kecil dan tidak berkorelasi. (Nugraheny, 2013). Tujuan dari komponen utama adalah untuk mereduksi dimensi suatu variabel yang saling terkait, dan variabelnya banyak, sehingga data lebih mudah untuk diinterpretasikan. Algoritma PCA dalam perhitungan *zeromean* adalah mencari nilai rata-rata citra dengan cara mengurangkan target mencari nilai rata-rata semua citra kemudian mengurangkannya dan mencari nilai kovarians citra latih. Setelah matriks kovarians diperoleh, dilakukan reduksi, kemudian citra latih diubah menjadi dimensi yang lebih rendah. Selanjutnya memasuki tahap sertifikasi, akan ada proses perhitungan pada tahap sertifikasi, dan didapatkan hasil berupa akurasi (Priyawati, 2011).

Berikut adalah tahapan-tahapan dari metode *Principal Component Analysis* :

**a. Reduksi dimensi PCA**

Salah satu pendekatan yang berhubungan dengan dimensi tinggi adalah dengan mereduksi dimensi data. Pengurangan dimensi adalah pengurangan dimensi kumpulan data di bawah kondisi bahwa informasi penting dipertahankan. Pengurangan dimensi dapat digunakan untuk menyederhanakan data besar secara matematis.

**b. Nilai rata-rata (Mean)**

*Mean* adalah rata-rata dari data yang dikumpulkan. Untuk mendapatkan mean, semua data dibagi dengan jumlah data pengamatan (Indrian, 2008). Nilai rata-rata dari variabel  $x$  dapat dirumuskan :

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad (2.1)$$

Dimana  $n$  = data observasi

$x_i$  = nilai ke  $i$  dari variable  $x$

**c. Kovarian**

Kovarians mewakili hubungan antara distribusi data dari dua atau lebih variabel yang berbeda. Oleh karena itu, jika kita memiliki kumpulan data tiga dimensi ( $x, y, z$ ), maka kita dapat mengukur kovarians antara dimensi  $x$  dan  $y$ , dimensi  $x$  dan  $z$ , dan dimensi  $y$  dan  $z$ . Mengukur kovarians antara  $x$  dan  $x$ , atau  $y$  dan  $y$ , atau  $z$  dan  $z$  akan memberikan varians pada setiap dimensi  $x$ ,  $y$  dan  $z$  (Maisa Hana, 2014). Analisis komponen utama dapat mengubah data dari data besar menjadi data kecil. Vektor eigen dihasilkan dari matriks kovarians. Matriks kovarians sebelumnya mengalami transformasi PCA berdasarkan ortogonalitas untuk menyajikan distribusi data yang terbaik. Tugas PCA sendiri adalah untuk menghasilkan atau menangkap perubahan total gambar pelatihan dalam *database*. Setelah itu, kurangi untuk mendapatkan lebih sedikit variabel. Pengurangan dapat mengurangi waktu perhitungan dan kompleksitas gambar yang tidak dapat dihilangkan.

#### d. *Eigen vector dan Eigen value*

Eigen berasal dari bahasa Jerman yang artinya ciri. Nilai eigen didefinisikan sebagai nilai eigen matriks. Vektor eigen dari suatu matriks yang selalu bersesuaian dengan nilai eigen disebut vektor eigen. Vektor eigen dari matriks persegi adalah vektor yang bukan nol. Ketika dikalikan dengan matriks, nilai suatu vektor masih sebanding dengan vektor aslinya. Nilai eigen yang tepat dari vektor eigen adalah faktor yang mengubah vektor eigen saat mengalikan matriks (Maisa, 2014).

#### **Algoritma *Principal Component Analysis (PCA)***

Langkah-langkah untuk proses PCA adalah sebagai berikut (Turk, 1991).

##### 1. Membuat matriks data set daun S

Langkah pertama dalam metode PCA adalah membuat satu set semua gambar pelatihan dan menyiapkan datanya. Setiap citra diubah menjadi vektor berukuran  $n$  yang disebut sebagai dataset citra vektor (data latih) dan dimasukkan ke dalam himpunan S. Dimana N adalah jumlah citra. Representasi matriks S dapat digambarkan sebagai matriks berikut:

$$S = \begin{bmatrix} \Gamma_1 & \Gamma_{11} & \Gamma_{12} & \Gamma_{13} & \dots & \Gamma_{1n} \\ \Gamma_2 & \Gamma_{21} & \Gamma_{22} & \Gamma_{23} & \dots & \Gamma_{2n} \\ \Gamma_3 & \Gamma_{31} & \Gamma_{32} & \Gamma_{33} & \dots & \Gamma_{3n} \\ \vdots & & & \vdots & & \\ \Gamma_n & \Gamma_{n1} & \Gamma_{n2} & \Gamma_{n3} & \dots & \Gamma_{Nn} \end{bmatrix}$$

$$S = \{\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_N\}$$

##### 2. Menghitung matriks rata-rata (*mean*) $\Psi$

Rumus matriks rata-rata adalah :

$$\Psi = \frac{1}{M} \sum_n^M = 1 \Gamma_n \quad (2.2)$$

Dari rumus diatas maka akan didapatkan hasil  $\Psi = [\Psi_1 \Psi_2 \Psi_3 \dots \Psi_n]$

$\Psi$  = matriks rata-rata (*mean*)

M = banyaknya data di data set

$\Gamma_n =$  data ke-n

### 3. Membuat matriks selisih $\Phi$

Matriks selisih  $\Phi$  dibuat dengan mengurangkan setiap elemen matriks  $\Gamma$  pada kolom ke- $i$  dengan  $\Psi$ . Representasi matriks  $\Phi$  dapat dilihat pada rumus dibawah :

$$\Phi = \Gamma_i - \Psi \quad (2.3)$$

Dari persamaan di atas, dapat dinyatakan dalam matriks perbedaan berikut.:

$$\Phi = \begin{bmatrix} \Gamma_{11} & -\psi_1 & \Gamma_{12} & -\Psi_2 & \Gamma_{13} & -\Psi_3 & \dots & \Gamma_{1n} & -\Psi_n \\ \Gamma_{21} & -\Psi_1 & \Gamma_{22} & -\Psi_2 & \Gamma_{23} & -\Psi_3 & \dots & \Gamma_{2n} & -\Psi_n \\ \Gamma_{31} & -\Psi_1 & \Gamma_{32} & -\Psi_2 & \Gamma_{33} & -\Psi_3 & \dots & \Gamma_{3n} & -\Psi_n \\ \vdots & & & & & & & & \\ \Gamma_{N1} & -\Psi_1 & \Gamma_{N2} & -\Psi_2 & \Gamma_{N3} & -\Psi_3 & \dots & \Gamma_{Nn} & -\Psi_n \end{bmatrix}$$

$\Phi =$  Matriks selisih

$\Gamma_i =$  data ke- $i$

$\Psi =$  data rata-rata (mean)

### 4. Membuat matriks kovarian C

Rumus untuk menghitung matriks kovarians C adalah :

$$C = \frac{1}{M} \sum_n^M = \frac{1}{M} (\Phi_n) (\Phi_n)^T = AA^T \text{ atau } L = A^T A \quad (2.4)$$

$$A = \{\Phi_1, \Phi_2, \Phi_3, \dots, \Phi_n\}$$

C = Matriks kovarian berdimensi  $n \times n$

$\Gamma =$  berisi matriks selisih

$\Phi =$  matriks selisih

M = Jumlah data set

### 5. Menghitung *eigenvector* ( $v$ ) dan *eigenvalue* ( $\lambda$ ) dan matriks kovarian ( $C$ )

Rumus untuk menghitung vektor eigen dan nilai eigen dari matriks kovarians adalah:

$$C v = \lambda v \quad (2.5)$$

Dari rumus diatas, maka didapatkan sebuah persamaan :

$$\begin{aligned} (C v - \lambda v) &= \mathbf{0} \\ (C - \lambda I) &= \mathbf{0} \text{ atau } \det (\lambda I - C) = \mathbf{0} \end{aligned} \quad (2.6)$$

Gunakan persamaan di atas untuk mencari nilai eigen ( $\lambda$ ) dan vektor eigen ( $v$ ) secara bersamaan, kemudian kita substitusikan nilai nilai eigen ( $\lambda$ ) ke dalam persamaan:

$$(\lambda I - C)v = \mathbf{0} \quad (2.7)$$

Keterangan :

$\lambda$  = *eigenvalue* atau nilai eigen

$v$  = *eigenvector* atau vektor eigen

$C$  = matriks kovarian

$I$  = matriks identitas

### 6. Mencari nilai *Principal Component* (PC)

Pada proses 5, *eigenvectors* dan *eigenvalues* akan dihasilkan. Dari *eigenvector* tersebut diambil  $m$  *eigenvector*  $v$  yang memiliki nilai eigen value terbesar, karena ini merupakan satu-satunya *eigenvector* yang memiliki dimensi yang sama dengan gambar aslinya, maka *eigenvector* ini juga dapat disebut sebagai *principal component* (PC). Nilai PC juga dapat dicari dengan rumus berikut:

$$PC = A^T * V \quad (2.8)$$

Keterangan :

PC = *Principal Component*

$A$  = *Matriks mean corrected data*

$V$  = *Eigen vector*



7. Lakukan transformasi data untuk menghasilkan data PCA. PCA dapat dihitung menggunakan persamaan berikut::

$$PCA = A * PC \quad (2.9)$$

Keterangan :

PCA = *Principal Component Analysis*

PC = *Principal Component*

A = *Matriks Mean Corrected Data*

## 2.6 Klasifikasi

Klasifikasi adalah metode yang digunakan untuk mengelompokkan suatu item ke dalam kategori tertentu. Klasifikasi adalah proses membuat model yang memisahkan kelompok data berdasarkan kategorinya. (Han, 2006).

Klasifikasi data terdiri dari 2 langkah. Tahap pertama adalah membuat algoritma untuk data latih, yang kemudian disajikan dalam bentuk aturan klasifikasi. Proses kedua adalah klasifikasi, yang menggunakan data uji untuk menghitung keakuratan aturan klasifikasi. (Han, 2006).

Bentuk daun merupakan faktor penentu dalam mengklasifikasikan tumbuhan. Langkah pengenalan pola daun seperti ciri struktur daun ini dapat dilakukan.

## 2.7 K-Nearest Neighbor

Algoritma *K-Nearest Neighbor* merupakan salah satu contoh metode klasifikasi dimana data diklasifikasikan dengan konsistensi yang kuat dengan menghitung kedekatan antara kasus baru dan kasus lama. (Lutfhi & Kusri, 2009). KNN berisi algoritme pembelajaran terawasi yang mengklasifikasikan hasil instans kueri baru berdasarkan sebagian besar kategori KNN. Kategori yang paling banyak dilihat adalah kategori yang dihasilkan.

Salah satu metode yang digunakan untuk menyelesaikan masalah klasifikasi adalah algoritma k-nearest neighbor. KNN mengklasifikasikan data baru yang kelasnya masih belum diketahui, berdasarkan jarak data baru tersebut ke beberapa

tetangga terdekat. Tetangga terdekat adalah data dengan jarak terkecil atau terbesar dari data lama.

Secara umum, semakin tinggi nilai  $k$ , semakin banyak dampak kebisingan pada klasifikasi akan berkurang, tetapi batas antar kelas akan kabur. Dalam kasus khusus memprediksi klasifikasi berdasarkan data sampel terdekat, yaitu Tetangga terdekat adalah vektor dengan panjang 1.

Pendekatan sederhana untuk menentukan nilai  $k$  yaitu :

$k\sqrt{n}$ , dimana  $n$  adalah jumlah dari sampel data yang ada.

Misalkan terdapat 30 sampel data, gunakan rumus untuk menentukan nilai  $k$

$$K = \sqrt{n} = \sqrt{30} = 5,47, \text{ berarti nilai } k = 5$$

$$K = \sqrt{n} = \sqrt{20} = 4,47, \text{ berarti nilai } k = 4$$

Nilai  $k$  biasanya ditentukan dengan bilangan ganjil (3, 5, 7) sehingga proses klasifikasi tidak menunjukkan jumlah jarak yang sama. Jika ada dua atau lebih kelas yang terlihat sama, nilai  $k$  akan menjadi  $k-1$  (jarak antara satu kelas tidak terlalu dekat). Jika kelas yang sama ada lagi, nilai  $k$  akan menjadi  $k-2$ , dan seterusnya. Kelas dapat ditemukan dengan nomor yang sama.

Prinsip kerja metode *K-Nearest Neighbor* adalah mencari jarak terpendek antar data yang akan dievaluasi dengan metode *Nearest Neighbor* pada data latih. (*K-Nearest Neighbor*) tidak hanya menghasilkan jarak terpendek, tetapi juga menghasilkan  $k$  jarak terpendek (Kusumadewi, 2009). Jumlah maksimum kelas dengan jarak terpendek adalah kelas dari data yang dievaluasi. Tetangga dekat atau tetangga jauh (*neighbors*) biasanya dihitung berdasarkan jarak *Euclidean*.

Prosedur perhitungan algoritma K-NN :

1. Tentukan nilai  $k$ .
2. Menghitung kuadrat jarak *euclidean* setiap objek untuk data pelatihan yang ditentukan.
3. Urutkan obyek-obyek tersebut ke dalam kelompok yang mempunyai *Euclidian Distance* terkecil.

4. Mengumpulkan label kelas Y (klasifikasi *Nearest Neighbor*).

Dengan menggunakan kategori *Nearest Neighbor* yang paling mayoritas maka dapat diprediksikan nilai *query instance* yang telah dihitung.

## 2.8 Euclidean Distance

Jarak *Euclidean* adalah jarak antara dua titik dalam ruang. Jarak antara dua titik adalah garis terpendek yang menghubungkannya. Hasil ekstraksi ciri digunakan sebagai teknik identifikasi dengan cara pengukuran kemiripan. Ada beberapa metode pengukuran kemiripan citra yaitu *Mahalanobis*, jarak *Euclidean*, *Bayes*, sudut antar vektor fitur (Muntasa, 2010).

Jarak *Euclidean* adalah pengukuran dengan menghitung kesamaan dua vektor dan nilai minimum sebagai nilai yang paling mirip.

Ini adalah persamaan umumnya:

$$d(x,y) = \|x - y\| = \sqrt{\sum(\omega_j - \omega_i)^2} \quad (2.10)$$

Keterangan :

$d(x,y)$  = jarak *euclidean*

$\omega_j$  = bobot citra uji

$\omega_i$  = bobot citra latih

Dalam penelitian ini, metode *Euclidean distance* digunakan sebagai metode jarak karena merupakan metode *similarity* yang paling sederhana untuk meminimalkan proses perhitungan. *Euclidean distance* merupakan ukuran jarak yang dapat digunakan untuk menentukan kemiripan objek dan mengklasifikasikan jarak berdasarkan kedekatan objek tetangga. Nilai minimum dalam kumpulan data ini adalah 0, yang paling mirip dengan objek.

Jarak hitung yang dihasilkan oleh jarak *Euclidean* semakin kecil. Pada penelitian ini, jarak *Euclidean* didasarkan pada jarak antara citra latih dan data uji untuk mengetahui tingkat kemiripan. Dalam penelitian ini, jarak *Euclidean* diterapkan menggunakan nilai bobot citra latih dan nilai bobot data uji.

## 2.9 Akurasi Pengujian

Hasil penelitian ini diukur dengan menghitung tingkat akurasi hasil klasifikasi jenis daun jambu biji dengan menggunakan data uji. Tingkat akurasi ini dihitung dengan menggunakan rumus berikut :

$$\text{Akurasi} = \frac{\text{Jumlah klasifikasi yang benar}}{\text{Jumlah semua data}} \times 100\% \quad (2.10)$$

## 2.10 Penelitian Terkait

Penelitian Dessie Maya Sari (2009) berjudul “Penelitian Penggunaan Analisis Komponen Utama dan Metode Jarak *Euclidean* dalam Pengenalan Sistem Mutu Buah Tomat”. Dalam penelitian ini, PCA digunakan untuk memperkecil ukuran. Proses mengidentifikasi tomat yang paling mirip dengan objeknya. Bobot masing-masing citra latih dan citra uji akan dibandingkan dengan menggunakan nilai jarak *Euclidean*. Sistem dapat membaca 90% tomat dengan benar, tetapi tidak dapat membaca salah satunya.

Penelitian yang dilakukan oleh Herfina (2013) berjudul “Menggunakan Analisis Komponen Utama dan K-NN untuk Mengenalkan Pola Bunga”. Tujuan dari penelitian ini adalah untuk mengetahui bagaimana bentuk bunga bervariasi dari satu gambar ke gambar berikutnya. Metode yang digunakan dalam penelitian ini adalah analisis komponen utama dan K-nearest neighbor. PCA digunakan dalam proses ekstraksi ciri dan K-NN digunakan dalam proses klasifikasi citra bunga. Penggunaan K-Nearest Neighbors berdasarkan ekstraksi fitur *Principal Component Analysis* untuk mengevaluasi pengenalan pola bentuk citra bunga diperoleh hasil 97,70%.

Selain itu, penelitian yang dilakukan oleh Salamah Nur Aqidah (2019) dalam makalah yang berjudul “Analisis Komponen Utama dan Pengolahan Citra Untuk Pengenalan Wajah Manusia Menggunakan *Euclidean Distance*”, meskipun mampu mengidentifikasi wajah, namun rata-rata akurasinya adalah 95%.

Aksesoris yang digunakan berupa kacamata dengan berbagai ekspresi. Penelitian ini dirancang untuk mendeteksi wajah pada citra menggunakan metode deteksi kulit berdasarkan model warna YcbCr. Penelitian ini dimulai dengan mendeteksi wajah dan bukan wajah pada citra menggunakan deteksi kulit. Setelah wajah terdeteksi, langkah selanjutnya adalah proses ekstraksi ciri menggunakan analisis komponen utama. Klasifikasi citra menggunakan metode jarak *Euclidean* dengan menghitung jarak antara dua titik. Metode PCA pada penelitian ini dapat mereduksi fitur wajah dari matriks 34 kali 40.000 menjadi matriks 34 kali 200. Matriks ini merupakan fitur pilihan karena memiliki dampak yang signifikan pada database.

Penelitian yang dilakukan oleh Resty Wulannigrum (2014) dalam makalah yang berjudul “Analisis Komponen Utama dan Penggunaan Jarak *Euclidean* untuk Mengidentifikasi Gambar Tanda Tangan”. Penelitian ini bertujuan untuk mengidentifikasi tanda tangan dari berbagai proses. Pengujian ini menunjukkan tingkat akurasi tertinggi dengan ambang batas 50-219, dengan nilai akurasi 95%. Menggunakan dimensi yang berbeda antara gambar pelatihan dan gambar uji memberikan akurasi 60%. Pengujian dengan warna tinta yang berbeda telah menunjukkan akurasi hingga 100%. Oleh karena itu, tanda tangan dapat diidentifikasi menggunakan metode jarak PCA dan *Euclidean*.

Fadhilillah et al, (2015), Analisis dan Implementasi Klasifikasi *K-Nearest Neighbor* (K-NN) Dalam Sistem Biometrik Jejak Kaki Manusia, Dalam Studinya, Persepsi Individu Penting dalam Banyak Aspek Kehidupan Modern untuk Diperoleh Tidak jarang korban tidak terluka dalam kasus informasi atau identitas tertentu, seperti bencana alam, yang telah digambarkan sebagai bagian dari cerita. Hal ini membuat sulit untuk mengidentifikasi. Salah satu solusinya adalah mengidentifikasi telapak kaki, biometrik telapak kaki. Ini dapat digunakan untuk mengidentifikasi individu secara akurat menggunakan metode *K-Nearest Neighbor* (KNN). Biometrik pada telapak kaki memenuhi persyaratan pemilihan biometrik: universal, identifikasi, dan permanen. Berikut ini adalah klasifikasi nilai K. Sesuaikan untuk akurasi maksimum. Dalam proses identifikasi, sistem terlebih dahulu menangkap seluruh tampilan gambar kaki kemudian secara

otomatis memotong gambar ke bagian yang diinginkan dengan ukuran yang seragam. Metode ekstraksi fitur citra menggunakan wavelet Haar, kemudian menggunakan algoritma clustering untuk mengklasifikasi citra, dan didapatkan akurasi yang dihasilkan untuk hasil tes. Saat menggunakan metode K-NN untuk mengenali kaki manusia, hasil penggunaan metode *Euclidean distance* dan *cosinus distance* mencapai 98%.

Dinda Rizki Taningrum (2016) Sistem identifikasi pelat nomor kendaraan menggunakan analisis komponen utama untuk membedakan pelat dan klasifikasi K-NN untuk mengidentifikasi kelas pelat. Pada tugas akhir ini, saya akan membuat sebuah sistem yang dapat mengenali plat nomor kendaraan. Analisis komponen utama (PCA) digunakan untuk mengekstrak fitur dan K-NN digunakan untuk mengklasifikasikan karakter. Sistem ini mampu mengenali 44 pelat data uji dengan akurasi 97,78%. Akurasi segmentasi karakter adalah 99,1% atau diambil dari 334 karakter yang dipilih secara acak menjadi 331 karakter yang berhasil tersegmentasi. Data pelatihan berhasil digunakan untuk mengklasifikasikan 297 karakter dengan benar, dengan akurasi 88,92% dan akurasi keseluruhan 60%. Atau, 27 dari 45 data uji berhasil dideteksi dan diidentifikasi.

Pengembangan sistem deteksi katarak menggunakan *Principal Component Analysis* dan *K-Nearest Neighbor*. Pada penelitian ini, sistem dapat mendeteksi dan mengklasifikasikan katarak, katarak imatur, dan mata normal dengan akurasi 70,27%. Pengambilan gambar 37 mata, 10 gambar mata normal, 17 gambar katarak imatur dan 10 gambar katarak matur.



## BAB III METODOLOGI PENELITIAN

### 3.1 Tempat dan Waktu Penelitian

#### 3.1.1 Tempat Penelitian

Tempat penelitian ini diadakan di Laboratorium Multimedia Fakultas Sains dan Teknologi yang berada di Jalan IAIN No.1 Medan, Sumatera Utara.

#### 3.1.2 Waktu Penelitian

Penelitian ini dilaksanakan pada semester genap tahun ajaran 2019/2020 yaitu antara bulan Mei sampai dengan Agustus 2020 dengan tabel sebagai berikut :

**Tabel 3.1.2 Waktu Penelitian**

No	Kegiatan	Bulan 2020			
		Mei	Juni	Juli	Agustus
1.	Tahap Persiapan Penelitian				
	a. Penyusunan dan Pengajuan Judul				
	b. Pengajuan Proposal				
	c. Perijinan Penelitian				
2.	Tahap Pelaksanaan				
	a. Pengumpulan data				
	b. Analisis Data				
3.	Tahap Penyusunan Laporan				



## 3.2 Bahan dan Alat Penelitian

### 3.2.1 Perangkat Keras

Kebutuhan perangkat keras yang digunakan untuk membangun sistem ini sebagai berikut :

#### 1. Laptop

Merk	: Asus
Type	: A44H
Processor	: <i>Intel Core i3-2330M (2.20GHz)</i>
RAM	: 4.00 GB

#### 2. *Smartphone*

Merk	: Vivo
Type	: Y71
Kamera	: 13 MP
CPU	: <i>Snap dragon 425 Quad-core Cortex-A53</i>
RAM	: 3.00 GB
Internal Memory	: 32 GB

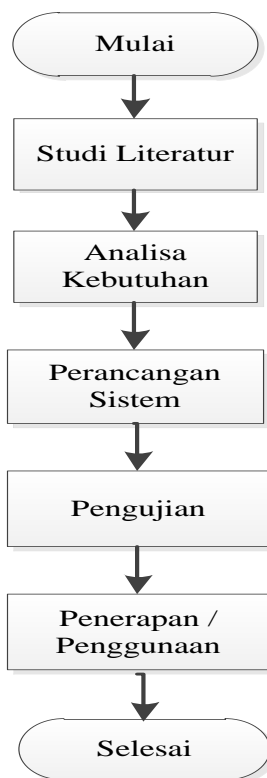
### 3.2.2 Perangkat Lunak

Adapun kebutuhan perangkat lunak yang digunakan untuk membangun sistem ini adalah sebagai berikut:

1. *Operating System Windows 7 32 bit.*
2. Matlab 2012b.

## 3.3 Prosedur Kerja Penelitian

Prosedur kerja ini dimaksudkan untuk menjelaskan semua tahapan kegiatan yang dilakukan selama penelitian agar sejalan dengan tujuan yang diberikan. Langkah-langkah yang dilakukan adalah sebagai berikut :



**Gambar 3.1** Diagram Prosedur Kerja

### 3.3.1 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini adalah :

1. Studi Literatural

Studi literatural adalah rangkaian kegiatan yang berkaitan dengan metode pengumpulan daftar pustaka, membaca dan mencatat, serta rangkaian kegiatan yang berkaitan dengan pengolahan bahan penelitian atau mencari referensi teori yang berkaitan dengan kasus atau masalah yang berkaitan dengan tugas akhir ini. Tujuannya adalah untuk memberikan ide-ide untuk mengembangkan kerangka konseptual metode penelitian berdasarkan tinjauan pustaka.

2. Observasi

Observasi yaitu pengumpulan data yang dilakukan dengan cara meninjau daun jambu langsung. Pengamatan dilakukan dengan meninjau daun jambu biji, yang dilakukan dengan mengamatinya secara

langsung. Cara pengambilan gambarnya adalah dengan memetik daun jambu biji dan meletakkannya di atas kertas putih. Setelah itu, smartphone Vivo Y71 digunakan untuk memotret daun jambu biji. Ambil gambar dari sudut atas yang menunjukkan permukaan daun.

### 3.3.2 Analisa Kebutuhan

#### A. Metode Analisa

Sistem klasifikasi buah jambu biji dirancang menggunakan analisis komponen utama dan metode k-nearest neighbor. Diagram proses sistem ini meliputi proses input dan proses output. Pada tahap ini, simbol digunakan untuk menggambarkan proses dari metode dan sistem, yang akan sangat membantu dalam proses pengkategorian jenis jambu biji ketika pengguna menggunakannya.

#### B. Hasil Analisa

Data yang diperoleh dari observasi selama proses penelitian dan data setelah proses analisis, meliputi kebutuhan proses, kebutuhan input dan kebutuhan output, yaitu:

##### 1. Analisa Kebutuhan Proses

Persyaratan proses untuk sistem klasifikasi jenis jambu biji adalah :

- a. Proses penginputan citra daun jambudengan format *Joint Photographic Expert Group* (JPG).
- b. Proses ekstrasi ciri citra daun jambu serta klasifikasi citra daun jambu.

##### 2. Analisa Kebutuhan *Input*

*Input* atau masukan dari sistem klasifikasi citra daun jambu ini, berupa parameter-parameter yang diperlukan dalam metode *Principal Component Analysis* (PCA) dan *K-Nearest Neighbor* (K-NN). yaitu:

- a. Kumpulan data yang digunakan untuk melakukan proses ekstraksi ciri adalah citra daun jambu biji dengan format *Joint Photographic Expert Group* (JPG).

b. B. Parameter yang diperlukan untuk perhitungan metode PCA yaitu: Membuat matriks dataset daun, menghitung matriks rata-rata (mean  $\psi$ ), Membuat matriks selisih ( $\Phi$ ), Membuat matriks kovarian C, Menghitung *eigenvector* ( $v$ ) dan *eigenvalue* ( $\lambda$ ) dan matriks kovarian (C), Mencari nilai *Principal Component* (PC). Setelah itu dilakukan klasifikasi dengan metode K-NN yaitu menentukan nilai k, lalu menghitung kuadrat *Euclidean Distance*, objek ke dalam kumpulan dengan jarak Euclidean terkecil, mengumpul label kelas Y (K-Nearest Neighbor) lalu dilakukan perhitungan jarak kemiripan dengan menggunakan *Euclidean Distance*.

### 3. Analisa Kebutuhan *Output*

Data keluaran dari proses sistem ini adalah citra yang diperoleh dari ekstraksi ciri dan klasifikasi yang secara otomatis ditampilkan pada layar.

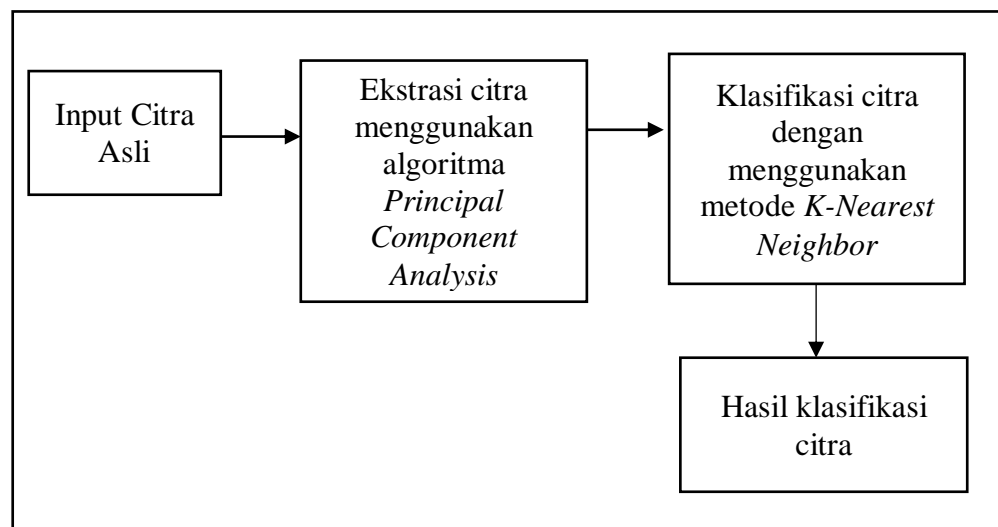
### C. Kebutuhan Antarmuka

Desain antarmuka terbaik untuk mengimplementasikan proses adalah dengan menggunakan MATLAB. Keuntungan lain dari antarmuka sistem adalah memiliki tampilan yang memudahkan pengguna dalam menggunakan sistem ini.

### 3.3.3. Perancangan

Perancangan desain adalah langkah pertama dalam fase desain pengembangan rekayasa produk atau sistem. Desain adalah proses penerapan berbagai teknik dan prinsip untuk tujuan mendefinisikan suatu objek.

Berikut adalah diagram perancangan sistem pengenalan jenis jambu berdasarkan daun.

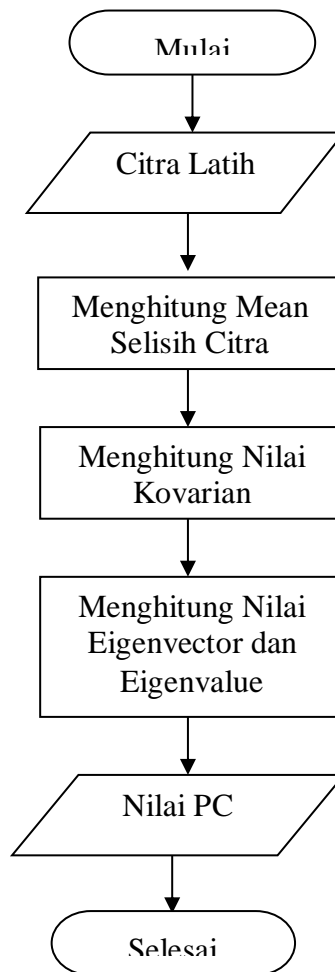


**Gambar 3.2** Diagram Sistem

Gambar di atas menunjukkan aliran sistem pengenalan jenis jambu biji berbasis daun. Tahap pengujian program ini adalah menyiapkan data latih dan uji berupa citra daun jambu biji yang terdiri dari 47 citra daun jambu biji sebagai citra latih dan 16 citra daun jambu biji sebagai citra uji. Ekstraksi fitur kemudian dilakukan dengan menggunakan algoritma *Principal Component Analysis* (PCA), dan tahap selanjutnya adalah klasifikasi menggunakan metode *K-Nearest Neighbor* (K-NN). Selanjutnya adalah tahap identifikasi dengan menghitung jarak Euclidean sebagai acuan. Menentukan klasifikasi daun jambu biji. Hasil ekstraksi citra menggunakan algoritma *Principal Component Analysis* (PCA) kemudian diperoleh, dan algoritma *K-Nearest Neighbor* (K-NN) digunakan dalam proses klasifikasi citra daun jambu biji.

#### ***Flowchart Metode Principal Components Analysis***

Pada tahap ini dilakukan perancangan sebuah program untuk klasifikasi jenis buah jambu berdasarkan daun dengan metode fitur ekstraksi menggunakan *Principal Component Analysis*.



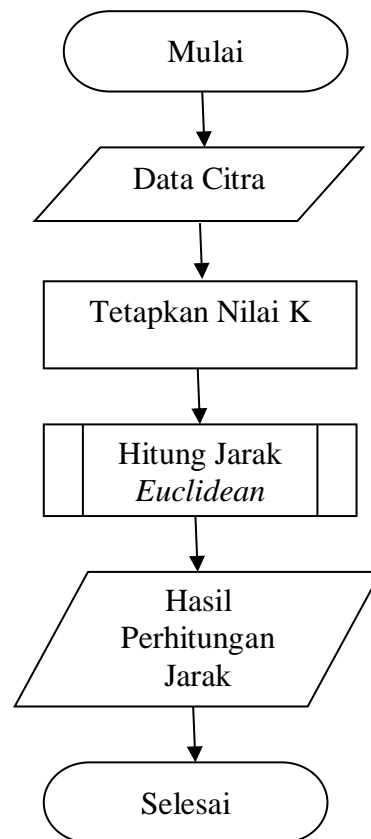
**Gambar 3.3** Flowchart Proses *Principal Component Analysis*.

Berikut ini menjelaskan diagram alur cara kerja Analisis Komponen Utama :

- a. Hitung *mean* dari matriks (mean). Matriks rata-rata dihitung dengan membagi jumlah semua vektor datar dengan jumlah vektor dalam *database*.
- b. Hitung vektor eigen dan nilai eigen dari matriks kovarians. Nilai eigen dan vektor nilai eigen diperoleh dari matriks kovarians, yang berisi fitur utama dari data. Nilai eigen adalah nilai karakteristik dari matriks. Vektor eigen adalah vektor kolom bukan nol, tetapi perkaliannya dengan matriks menghasilkan vektor lain yang merupakan kelipatan dari vektor eigen itu sendiri. Disebut juga nilai PC karena proses perhitungan untuk mencari nilai vektor eigen berhenti

- pada nilai maksimum.  
c. Selesai.

**Flowchart Metode *K-Nearest Neighbor* (*K-NN*)**



**Gambar 3.4** Flowchart Proses Klasifikasi *K-Nearest Neighbor*

Berikut penjelasan dari *flowchart* cara kerja *K-Nearest Neighbor* :

- Menentukan nilai  $k$  yang memiliki nilai kemiripan terbesar dengan jumlah tetangga terdekat dinyatakan dengan  $k$ .
- Menghitung nilai *euclidean distance* masing-masing objek terhadap data training yang diberikan.
- Objek diurutkan kepada kumpulan dengan jarak *Euclidean* terkecil.

### **3.3.4 Pengujian**

Penelitian ini perlu diuji untuk mengetahui apakah sistem berfungsi dengan baik. Pengujian penelitian ini adalah untuk melihat seberapa baik data terdistribusi dan kepada kelompok atau kelompok mana data tersebut didistribusikan. Folder data untuk pengenalan citra dibagi menjadi dua jenis data, yaitu data latih dan data uji. Pengujian dilakukan pada daun jambu biji dalam format JPEG.

### **3.3.5 Penerapan / Penggunaan**

Penerapan sistem ini adalah pengenalan daun, yaitu dapat mengidentifikasi tumbuhan dengan membedakan pola dan bentuk daun. Pengolahan citra digital dapat dilakukan pada citra seperti daun, sehingga mesin atau komputer dapat mengenali penglihatan manusia.



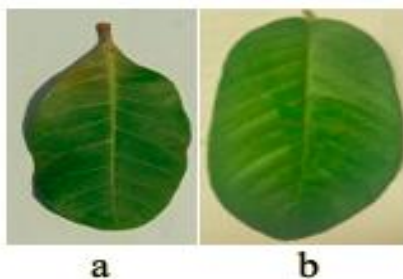
## BAB IV PEMBAHASAN DAN HASIL

### 4.1 Pembahasan

Pada bab ini, adapun dilakukan pembahasan yang meliputi dari analisis data citra daun jambu yang akan dilatih dan diuji secara manual menggunakan metode *Principal Component Analysis* dan *K-Nearest Neighbor*.

#### 4.1.1 Analisis Data

Data yang digunakan untuk keperluan hitungan manual adalah sampel citra daun jambu yang berukuran  $3 \times 3$  *pixel* dalam klasifikasi jenis jambunya berdasarkan citra daun yang di ekstraksi menggunakan metode PCA. Hasil ekstraksi nilai PCA kemudian diklasifikasi menggunakan metode KNN. Langkah pertama adalah menentukan *dataset* sebagai data latih yang diekstraksi menggunakan metode PCA. Adapun *dataset* yang digunakan adalah citra daun jambu sebanyak 2 citra yang dilklasifikasi dengan jenis daun jambu mete variabel (A) dan daun jambu biji variabel (B) yang digunakan sebagai variabel data latih dengan ukuran  $3 \times 3$  *pixel*. Berikut adalah *dataset* berupa citra daun jambu:

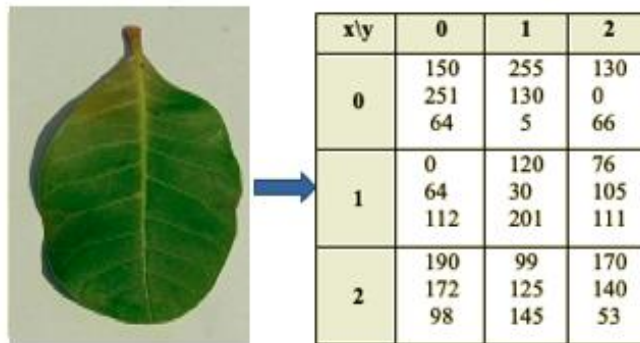


**Gambar 4.1** a. Daun Jambu Mete, b. Daun Jambu Biji

Berdasarkan pada gambar 4.1, diambil nilai pixel citra sebanyak  $3 \times 3$  pixel untuk diproses menjadi dataset menggunakan metode PCA. Sebelum diekstraksi dengan PCA terlebih dahulu nilai citra dirubah kedalam *grayscale* (abu-abu).

#### 1. Nilai Pixel $3 \times 3$ Citra Daun Jambu A

Adapun nilai pixel sampel  $3 \times 3$  dari citra daun jambu A adalah sebagai berikut :



x\y	0	1	2
0	150 251 64	255 130 5	130 0 66
1	0 64 112	120 30 201	76 105 111
2	190 172 98	99 125 145	170 140 53

**Gambar 4.2** Nilai *Pixel* 3x3 Citra Jambu A

Berdasarkan pada gambar 4.2, selanjutnya adalah melakukan *grayscale* dihitung dengan menggunakan rumus sebagai berikut:

$$(x, y) = (0.21 \times R) + (0.71 \times G) + (0.07 \times B)$$

Dari persamaan diatas, maka perhitungan untuk matriks (0,0) sampai dengan matriks (2,2) adalah:

$$(0,0) = (0.21 \times 150) + (0.71 \times 251) + (0.07 \times 64) = 214,19$$

$$(0,1) = (0.21 \times 0) + (0.71 \times 64) + (0.07 \times 112) = 53,28$$

$$(0,2) = (0.21 \times 190) + (0.71 \times 172) + (0.07 \times 98) = 168,88$$

$$(1,0) = (0.21 \times 255) + (0.71 \times 130) + (0.07 \times 5) = 146,2$$

$$(1,1) = (0.21 \times 120) + (0.71 \times 30) + (0.07 \times 201) = 60$$

$$(1,2) = (0.21 \times 99) + (0.71 \times 125) + (0.07 \times 145) = 119,69$$

$$(2,0) = (0.21 \times 130) + (0.71 \times 0) + (0.07 \times 66) = 31,92$$

$$(2,1) = (0.21 \times 76) + (0.71 \times 105) + (0.07 \times 111) = 98,28$$

$$(2,2) = (0.21 \times 170) + (0.71 \times 140) + (0.07 \times 53) = 138,81$$

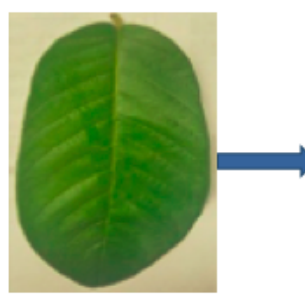
Hasil nilai piksel keabuan 3x3 telah dibulatkan pada tabel 4.1.

**Tabel 4.1** Hasil Nilai Pixel 3 x 3 *Grayscale* Sampel Daun Jambu A

x\y	0	1	2
0	214	146	32
1	53	60	98
2	169	120	139

## 2. Nilai *Pixel* 3x3 Citra Daun Jambu B

Adapun nilai *pixel* sampel 3x3 dari citra daun jambu B adalah sebagai berikut :



x\y	0	1	2
0	155 248 73	249 132 7	129 3 68
1	5 76 115	135 25 199	70 110 109
2	190 168 96	104 121 142	165 142 51

**Gambar 4.3** Nilai *Pixel* 3x3 Citra Jambu B

Berdasarkan Gambar 4.3, langkah selanjutnya adalah menghitung grayscale menggunakan persamaan berikut:

$$(x, y) = (0.21 \times R) + (0.71 \times G) + (0.07 \times B)$$

Dari persamaan di atas, perhitungan dari matriks (0,0) ke matriks (2,2) adalah:

$$(0,0) = (0.21 \times 155) + (0.71 \times 248) + (0.07 \times 73) = 213,74$$

$$(0,1) = (0.21 \times 5) + (0.71 \times 76) + (0.07 \times 115) = 63,06$$

$$(0,2) = (0.21 \times 190) + (0.71 \times 168) + (0.07 \times 96) = 165,2$$

$$(1,0) = (0.21 \times 249) + (0.71 \times 132) + (0.07 \times 7) = 146,5$$

$$(1,1) = (0.21 \times 135) + (0.71 \times 25) + (0.07 \times 199) = 60$$

$$(1,2) = (0.21 \times 104) + (0.71 \times 121) + (0.07 \times 142) = 117,69$$

$$(2,0) = (0.21 \times 129) + (0.71 \times 3) + (0.07 \times 68) = 33,981$$

$$(2,1) = (0.21 \times 70) + (0.71 \times 110) + (0.07 \times 109) = 100,43$$

$$(2,2) = (0.21 \times 165) + (0.71 \times 142) + (0.07 \times 51) = 139,04$$

Dari perhitungan sebelumnya, hasil nilai piksel tipe grayscale 3 x 3 telah dibulatkan seperti terlihat pada Tabel 4.2.

**Tabel 4.2** Hasil Nilai *Pixel* 3 x 3 *Grayscale* Sampel Daun Jambu B

x\y	0	1	2
0	214	147	34
1	63	60	100
2	165	118	139

#### 4.1.2 Representasi Data

Setelah mendapatkan nilai grayscale untuk setiap variabel citra pada dataset, langkah selanjutnya adalah menerapkan metode PCA.. PCA memiliki beberapa proses yaitu membuat matriks dataset daun, kemudian menghitung nilai rata-rata (*mean*) dataset variabel daun, menghitung matriks selisih, mencari nilai matriks kovarian, menghitung nilai *eigenvector* dan *eigenvalue* serta melakukan transformasi nilai untuk menghasilkan data PCA. Adapun proses tersebut dapat dilihat sebagai berikut:

##### 1. Membuat Matriks Dataset

Pada proses metode PCA setiap citra dimasukkan kedalam dataset matrik sesuai jumlah citranya. Penelitian ini hanya mencontohkan 2 variabel citra daun jambu A dan B yang kemudian dimasukkan nilai grayscalenya kedalam matriks sebagai berikut:

$$\begin{bmatrix} 214 & 214 \\ 53 & 63 \\ 169 & 165 \\ 146 & 147 \\ 60 & 60 \\ 120 & 118 \\ 32 & 34 \\ 98 & 100 \\ 139 & 139 \end{bmatrix}$$

Berdasarkan pada matriks variabel citra daun jambu A dan daun Jambu B, untuk mempermudah hitungan maka dimasukkan kedalam tabel berikut:

**Tabel 4.3** Nilai Variabel A dan B

Data	Nilai Variabel Daun Jambu A	Nilai Variabel Daun Jambu B
1	214	214
2	53	63
3	169	165
4	146	147
5	60	60
6	120	118

**Tabel Lanjutan 4.3** Nilai Variabel A dan B

7	32	34
8	98	100
9	139	139

## 2. Menghitung Nilai Rata-Rata Dataset Variabel

Selanjutnya menghitung nilai rata-rata (*mean*) dataset variabel yaitu dengan menjumlah keseluruhan dataset setiap variabel kemudian dibagikan dengan jumlah nilai data.

$$\begin{aligned} \text{Mean Jambu A} &= (214 + 53 + 169 + 146 + 60 + 120 + 32 + 98 + 139) / 9 \\ &= 114,55 \end{aligned}$$

$$\begin{aligned} \text{Mean Jambu B} &= (214 + 63 + 165 + 147 + 60 + 118 + 34 + 100 + 139) / 9 \\ &= 115,55 \end{aligned}$$

Hasilnya ditunjukkan pada tabel di bawah ini.

**Tabel 4.4** Rata-Rata Nilai Variabel A dan B

Data	Nilai Variabel Daun Jambu A	Nilai Variabel Daun Jambu B
1	214	214
2	53	63
3	169	165
4	146	147
5	60	60
6	120	118
7	32	34
8	98	100
9	139	139
<b>TOTAL</b>	<b>1031</b>	<b>1040</b>
<b>MEAN</b>	<b>114,55</b>	<b>115,55</b>

### 3. Membuat Matriks Selisih

Proses selanjutnya adalah menghitung matriks selisih dengan mengurangi nilai data setiap variabel daun jambu A dan B dengan nilai rata-rata masing-masing variabel. Konsep proses dapat dilihat pada tabel berikut.

**Tabel 4.5** Nilai Matrik Selisih Variabel Jambu A dan B

Data	Nilai Jambu A – Means A	Nilai Jambu B – Means B
1	$214 - 114,5 = 99,45$	$214 - 155,55 = 98,45$
2	$53 - 114,5 = -61,55$	$63 - 155,55 = -62,55$
3	$169 - 114,5 = 54,45$	$165 - 155,55 = 53,45$
4	$146 - 114,5 = 31,45$	$147 - 155,55 = 30,45$
5	$60 - 114,5 = -54,55$	$60 - 155,55 = -55,55$
6	$120 - 114,5 = 5,45$	$118 - 155,55 = 4,45$
7	$32 - 114,5 = -82,55$	$34 - 155,55 = -83,55$
8	$98 - 114,5 = -16,55$	$100 - 155,55 = -17,55$
9	$139 - 114,5 = 24,45$	$139 - 155,55 = 23,45$

Setelah dilakukanya pengurangan, didadapatilah selisih pada matriks berikut:

99,45	98,45
-61,55	-62,55
54,45	53,45
31,45	30,45
-54,55	-55,55
5,45	4,45
-82,55	-83,55
-16,55	-17,55
24,45	23,45

#### 4. Membuat Matriks Konvarian

Setelah didapatkan selisih dari matriks atribut, selanjutnya adalah membentuk matriks kovarian. Membentuk nilai matriks kovarian terlebih dahulu mencari nilai varian dengan rumus berikut ini:

$$\text{Var} = \sum_{i=1}^n (xi - xmean)^2$$

Berdasarkan pada rumus di atas, terdapat 2 atribut atau variabel yang digunakan yaitu variabel nilai jambu A dan variabel nilai jambu B, sehingga untuk membentuk nilai varian dapat dilakukan dengan cara sebagai berikut:

$$\begin{aligned} \text{Var Nilai Jambu A} &= (214 - 114,55)^2 \\ &= (99,45)^2 \\ &= 9890,30 \end{aligned}$$

Proses yang sama dilakukan hingga data terakhir pada variabel nilai jambu B sehingga menghasilkan data varian untuk pembentukan data kovarian Seperti terlihat pada tabel berikut ini:

**Tabel 4.6** Nilai Keseluruhan Varian Variabel Jambu A dan B

Data	Nilai Selisih Jambu A	Varian A (Ai-Amean) <sup>2</sup>	Nilai Selisih Jambu B	Varian B (Bi-Bmean) <sup>2</sup>
1	99,45	9890,30	98,45	9692,40
2	-61,55	3788,40	-62,55	3912,50
3	54,45	2964,80	53,45	2856,90
4	31,45	989,10	30,45	927,20
5	-54,55	2975,70	-55,55	3085,80
6	5,45	29,70	4,45	19,80
7	-82,55	6814,50	-83,55	6980,60
8	-16,55	273,90	-17,55	308,00
9	24,45	597,80	23,45	549,90
Total		<b>28324,22</b>		<b>28333,12</b>

Selanjutnya adalah melakukan perkalian hasil Varian A dan Varian B seperti tabel berikut:

**Tabel 4.7** Nilai Rata-Rata Varian Variabel Jambu A dan B

Data	Nilai Selisih Jambu A	Nilai Selisih Jambu B	(A <sub>i</sub> -A <sub>mean</sub> ) (B <sub>i</sub> -B <sub>mean</sub> )
1	99,45	98,45	9790,85
2	-61,55	-62,55	3849,95
3	54,45	53,45	2910,35
4	31,45	30,45	957,65
5	-54,55	-55,55	3030,25
6	5,45	4,45	24,25
7	-82,55	-83,55	6897,05
8	-16,55	-17,55	290,45
9	24,45	23,45	573,35
<b>Total</b>			<b>28324,17</b>

Selanjutnya adalah membentuk nilai matriks kovarian dengan rumus sebagai berikut :

$$\text{Cov}(A, B) = \sum_{i=1}^n (x_i - x_{\text{mean}})(y_i - y_{\text{mean}}) / (n - 1)$$

$$\begin{pmatrix} \text{cov}(A, A) & \text{cov}(A, B) \\ \text{cov}(B, A) & \text{cov}(B, B) \end{pmatrix}$$

Berdasarkan pada matriks kovarian di atas, A dan A serta B dan B memiliki diagonal yang sama sehingga dapat disederhanakan menjadi sebagai berikut:

$$\begin{pmatrix} \text{cov}(A) & \text{cov}(A, B) \\ \text{cov}(B, A) & \text{cov}(B) \end{pmatrix} = \begin{pmatrix} \frac{28324,22}{9-1} & \frac{28324,17}{9-1} \\ \frac{28324,17}{9-1} & \frac{28333,12}{9-1} \end{pmatrix}$$

Sehingga hasilnya adalah sebagai berikut :

$$\text{Matriks Kovarian} = \begin{pmatrix} 3540,52 & 3540,52 \\ 3540,52 & 3541,64 \end{pmatrix}$$

##### 5. Menghitung *Eigenvalue* ( $\lambda$ ) dan *Eigenvector*( $v$ )

Setelah didapatkan nilai matriks kovarian, selanjutnya adalah mencari nilai *Eigenvalue* dan nilai *Eigenvector*.

###### a. Menghitung *Eigenvalue*



Adapun rumus mencari nilai *Eigenvalue* adalah dengan proses determinan seperti pada rumus di bawah ini:  $\lambda$

$$|MCov - \lambda(\text{Matriks Identitas})| = 0$$

$$\left| \begin{pmatrix} 3540,52 & 3540,52 \\ 3540,52 & 3541,64 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

$$\left| \begin{pmatrix} 3540,52 & 3540,52 \\ 3540,52 & 3541,64 \end{pmatrix} - \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} \right| = 0$$

Selanjutnya adalah melakukan perhitungan determinan dengan matriks ordo 2x2 sebagai berikut:

$$((3540,52 - \lambda)(3541,64 - \lambda) - (3540,52)(3540,52)) = 0$$

$$(\lambda^2 - 7082,16\lambda + 3965,38) = 0$$

Selanjutnya adalah mencari nilai lambda 1 dan lambda 2 untuk nilai *eigenvalue* 1 dan *eigenvalue* 2 dengan hitungan sebagai berikut:

$$\frac{-(-7082,16) \pm \sqrt{(-7082,16)^2 - 4(1)(3965,38)}}{2(1)}$$

$$\frac{7082,16 \pm 7081,04}{2}$$

$$\text{eigenvalue 1} = \frac{7082,16 + 7081,04}{2} = 7081,6$$

$$\text{eigenvalue 2} = \frac{7082,16 - 7081,04}{2} = 0,55$$

$$\text{eigenvalue 1} = 7081,6$$

$$\text{eigenvalue 2} = 0,55$$

b. Menghitung *Eigenvektor*

Setelah diketahui *eigenvalue1* dan *eigenvalue2*, selanjutnya nilai *eigenvalue* digunakan untuk mencari nilai *eigenvector*. Adapun rumus mencari nilai *eigenvector* berdasarkan nilai *eigenvalue* 1 = 7081,6 adalah sebagai berikut:

$$MCov(v) = \lambda v$$

Berdasarkan rumus di atas, matriks konvarian dikali dengan *eigenvector* sama dengan *eigenvalue* dikali *eigenvector* seperti di bawah ini:

$$\begin{pmatrix} 3540,52 & 3540,52 \\ 3540,52 & 3541,64 \end{pmatrix} \begin{pmatrix} v1 \\ v2 \end{pmatrix} = \lambda \begin{pmatrix} v1 \\ v2 \end{pmatrix}$$

$$\begin{pmatrix} 3540,52 & 3540,52 \\ 3540,52 & 3541,64 \end{pmatrix} \begin{pmatrix} v1 \\ v2 \end{pmatrix} = 7081,6 \begin{pmatrix} v1 \\ v2 \end{pmatrix}$$

$$3540,52v_1 + 3540,52v_2 = 7081,6v_1$$

$$3540,52v_1 + 3541,64v_2 = 7081,6v_2$$

Selanjutnya disederhanakan sesama variabel dengan pengenaan sehingga seperti berikut ini:

$$(3540v_1 - 7082v_1) + 3540v_2 = 0$$

$$3540v_1 + (3542v_2 - 7082v_2) = 0$$

$$-3542v_1 + 3540v_2 = 0$$

$$3540v_1 - 3540v_2 = 0$$

---


$$-2v_1$$

$$v_1 = \frac{1}{-2} = -0,5$$

Masukan persamaannya sebagai berikut

$$-3542(-0,5) + 3540v_2$$

$$1771 + 3540v_2$$

$$3540v_2 = -1771$$

$$v_2 = \frac{-1771}{3540} = 0,528$$

Proses perhitungan mencari nilai *eigenvector* berhenti pada lamda 1 dikarenakan nilai lamda atau *eigenvalue*1 adalah nilai yang terbesar, sehingga mendapatkan hasil *eigenvector* seperti berikut:

$$v = \begin{pmatrix} -0,5 \\ -0,528 \end{pmatrix}$$

Nilai *eigenvector* juga disebut dengan nilai *Principal Component* (PC).

## 6. Tranformasi Data PCA

Setelah didapatkan nilai *Principal Component* (PC), selanjutnya adalah melakukan tranformasi untuk menghasilkan nilai *Principal Component Analysis* (PCA) dengan rumus sebagai berikut:

$$PCA = A * PC$$

$$PCA = \begin{pmatrix} 214 & 53 & 169 & 146 & 60 & 120 & 32 & 98 & 139 \\ 214 & 63 & 165 & 147 & 160 & 118 & 34 & 100 & 139 \end{pmatrix} * \begin{pmatrix} -0,5 \\ -0,528 \end{pmatrix}$$

PCA

$$= \begin{pmatrix} (214 * (-0,5) + 214 * (-0,528))(53 * (-0,5) + 53 * (-0,528)) \\ (169 * (-0,5) + 169 * (-0,528))(146 * (-0,5) + 146 * (-0,528)) \\ (60 * (-0,5) + 60 * (-0,528))(120 * (-0,5) + 120 * (-0,528)) \\ (32 * (-0,5) + 32 * (-0,528))(98 * (-0,5) + 98 * (-0,528)) \\ (139 * (-0,5) + 139 * (-0,528)) \\ (214 * (-0,5) + 214 * (-0,528))(63 * (-0,5) + 63 * (-0,528)) \\ (165 * (-0,5) + 165 * (-0,528))(147 * (-0,5) + 147 * (-0,528)) \\ (160 * (-0,5) + 160 * (-0,528))(118 * (-0,5) + 118 * (-0,528)) \\ (34 * (-0,5) + 34 * (-0,528))(100 * (-0,5) + 100 * (-0,528)) \\ (139 * (-0,5) + 139 * (-0,528)) \end{pmatrix}$$

Berdasarkan proses dan hasil perkalian matrik PCA di atas, didapatkan matriks dataset baru PCA sebagai berikut:

-219,99	-219,99
-54,48	-64,76
-173,73	-169,62
-150,09	-151,12
-61,68	-61,68
-123,36	-121,30
-32,90	-34,95
-100,74	-102,80
-142,89	-142,89

Berdasarkan dari hasil kesuruhan proses ekstraksi dengan metode PCA menggunakan nilai sampel citra daun jambu 3x3 sebanyak 2 dataset, maka didapatkan dataset baru hasil ekstraksi metode PCA yang akan disimpan sebagai data uji dan data latih untuk klasifikasi jenis jambu berdasarkan citra daun jambu. Proses klasifikasi nilai dataset dilakukan dengan metode KNN.

#### 4.1.3 Hasil Analisis

Setelah didapatkannya data ekstraksi citra daun jambu menggunakan metode PCA, selanjutnya adalah melakukan klasifikasi jenis jambu berdasarkan dataset yang telah didapat.

### 1. Dataset PCA

Adapun dataset disusun dalam bentuk nilai rata-rata seperti pada tabel di bawah ini:

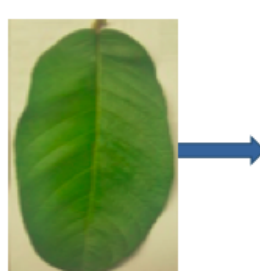
**Tabel 4.8** Dataset PCA

Data Daun	Nilai PCA Daun	Nilai Mean	Jenis Jambu
Daun A	-219,99	-117,76	Jambu Mete
	-54,48		
	-173,73		
	-150,09		
	-61,68		
	-123,36		
	-32,9		
	-100,74		
-142,89	-118,79	Jambu Biji	
-219,99			
-64,76			
-169,62			
-151,12			
-61,68			
-121,3			
-34,95			
-102,8			
-142,89			

Berdasarkan pada tabel 4. di atas, pada contoh hitungan manual ini, hanya dihitung 2 dataset yang akan dijadikan data klasifikasi.

### 2. Proses Klasifikasi *K-Nearest Neighbor* (KNN)

Diketahui sebuah citra daun jambu uji dengan nilai *pixel* sebagai berikut:



xy	0	1	2
0	160 241 72	230 130 13	110 13 72
1	9 79 113	141 23 187	76 114 121
2	187 162 91	111 117 151	190 160 40

**Gambar 4.4** Nilai *Pixel* 3x3 Citra Jambu Uji

Berdasarkan Gambar 4.4, maka langkah selanjutnya adalah melakukan perhitungan grayscale dengan menggunakan persamaan berikut:

$$(x, y) = (0.21 \times R) + (0.71 \times G) + (0.07 \times B)$$

Dari persamaan diatas, maka perhitungan untuk matriks (0,0) sampai dengan matriks (2,2) adalah:

$$(0,0) = (0.21 \times 160) + (0.71 \times 241) + (0.07 \times 72) = 209,75$$

$$(0,1) = (0.21 \times 9) + (0.71 \times 79) + (0.07 \times 113) = 65,89$$

$$(0,2) = (0.21 \times 187) + (0.71 \times 162) + (0.07 \times 91) = 166,96$$

$$(1,0) = (0.21 \times 230) + (0.71 \times 130) + (0.07 \times 13) = 141,5$$

$$(1,1) = (0.21 \times 141) + (0.71 \times 23) + (0.07 \times 187) = 59$$

$$(1,2) = (0.21 \times 111) + (0.71 \times 117) + (0.07 \times 151) = 116,95$$

$$(2,0) = (0.21 \times 110) + (0.71 \times 13) + (0.07 \times 72) = 37,7$$

$$(2,1) = (0.21 \times 76) + (0.71 \times 114) + (0.07 \times 121) = 105,37$$

$$(2,2) = (0.21 \times 190) + (0.71 \times 160) + (0.07 \times 40) = 136,72$$

Dari perhitungan sebelumnya maka didapat hasil nilai *pixel* 3 x 3 tipe *grayscale* yang telah dibulatkan seperti pada tabel 4.9.

**Tabel 4.9** Hasil Nilai *Pixel* 3 x 3 *Grayscale* Sampel Daun Jambu Uji

x\y	0	1	2
0	210	141	38
1	66	59	105
2	167	117	153

Berdasarkan hasil grayscale citra daun jambu uji, selanjutnya dilakukan perkalian dengan nilai *eigenvector* PCA sebagai berikut:

$$PCA = (210 \ 66 \ 167 \ 141 \ 59 \ 117 \ 38 \ 105 \ 137) * \begin{pmatrix} -0,5 \\ -0,528 \end{pmatrix}$$

$$PCA = \begin{pmatrix} (210 * (-0,5) + 210 * (-0,528)) & (66 * (-0,5) + 66 * (-0,528)) \\ (167 * (-0,5) + 167 * (-0,528)) & (141 * (-0,5) + 141 * (-0,528)) \\ (59 * (-0,5) + 59 * (-0,528)) & (117 * (-0,5) + 117 * (-0,528)) \\ (38 * (-0,5) + 38 * (-0,528)) & (105 * (-0,5) + 105 * (-0,528)) \\ & (153 * (-0,5) + 153 * (-0,528)) \end{pmatrix}$$

Berdasarkan perkalian di atas, didapatkan nilai citra PCA daun jambu sebagai data uji klasifikasi sebagai berikut:

**Tabel 4.10** Nilai Citra Data Uji

Data	Nilai PCA Uji	Mean PCA
Daun Jambu Uji	-215,88	-120,61
	-67,84	
	-171,67	
	-144,94	
	-60,65	
	-120,27	
	-39,06	
	-107,94	
	-157,28	

Berdasarkan dari hasil ekstraksi data uji dengan nilai *eigenvactor* PCA maka diketahui keseluruhan data yang disusun dalam bentuk nilai *mean* (rata-rata) sebagai berikut:

**Tabel 4.11** Nilai Keseluruhan Data Klasifikasi

Data	Nilai Means PCA	Jenis Jambu
Daun Jambu A	-117,76	Jambu Mete
Daun Jambu B	-118,79	Jambu Biji
Daun Jambu Uji	-120,61	???

Berdasarkan pada tabel di atas, selanjutnya dilakukan pencarian nilai *euclidean distance* untuk mendapatkan jarak nilai data uji dengan data jambu A dan B yang paling dekat. Adapun ditentukan  $K=1$ , yang artinya hanya diambil 1 contoh nilai paling dekat untuk mendapatkan jenis klasifikasi jambu. Adapun rumus untuk mendapatkan nilai *euclidean distance* dengan satu dimensi sebagai berikut:

$$dis(x_1, x_2) = \sqrt{\sum_{i=0}^n (x_{1i} - x_{2i})^2}$$

$$Dis \text{ Data } 1 = \sqrt{(-120,61 - (-117,76))^2} = 2,85$$

$$\text{Dis Data 2} = \sqrt{(-120,61 - (-118,79))^2} = 1,82$$

Berdasarkan hitungan *euclidean distance*, kemudian dirangkum dengan jarak paling dekat, maka menghasilkan klasifikasi seperti pada tabel di bawah ini:

**Tabel 4.12** Data Hasil Perhitungan Jarak Terdekat

Data	Nilai Means PCA	Jarak Dengan Data Baru	Jenis Jambu
Daun Jambu A	-117,76	2,85	Jambu Mete
Daun Jambu B	-118,79	1,82	Jambu Biji

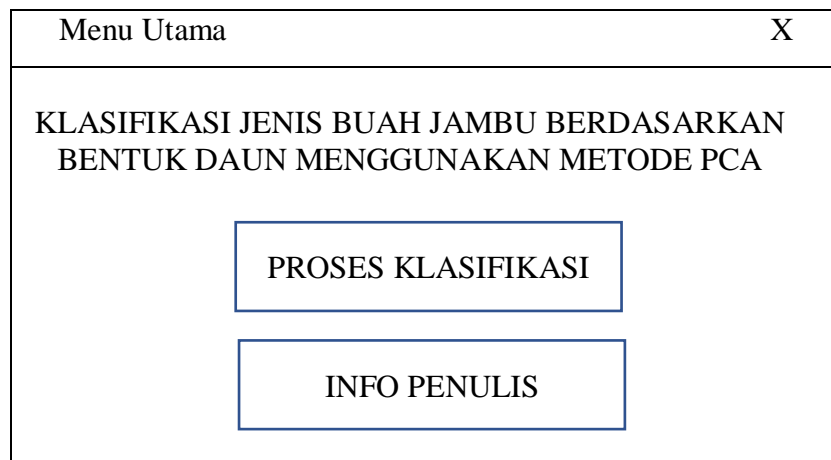
Berdasarkan dari hasil klasifikasi didapatkan K=1 atau 1 tetangga dengan jarak yang paling terdekat yaitu Daun Jambu B dengan klasifikasi Jambu Biji, sehingga data citra daun jambu uji yang diproses masuk kedalam klasifikasi jenis “Jambu Biji”

#### 4.1.4 Perancangan Sistem

Sebelum diimplementasi kedalam sebuah program aplikasi, terlebih dahulu merancang sistem klasifikasi jenis jambu berdasarkan bentuk daun jambu agar *system* dapat berjalan sebagaimana mestinya. Adapun perancangan sistem terdiri dari rancangan menu utama, rancangan menu info penulis dan rancangan menu klasifikasi. Adapun keseluruhan rancangan *interface* sistem adalah sebagai berikut:

##### 1. Rancangan Menu Utama

Menu utama adalah menu yang ditampilkan saat program pertama kali dijalankan. Adapun menu utama juga sebagai navigasi untuk kemenu-menu yang lain. Berikut adalah rancangan menu utama dapat dilihat pada gambar di bawah:



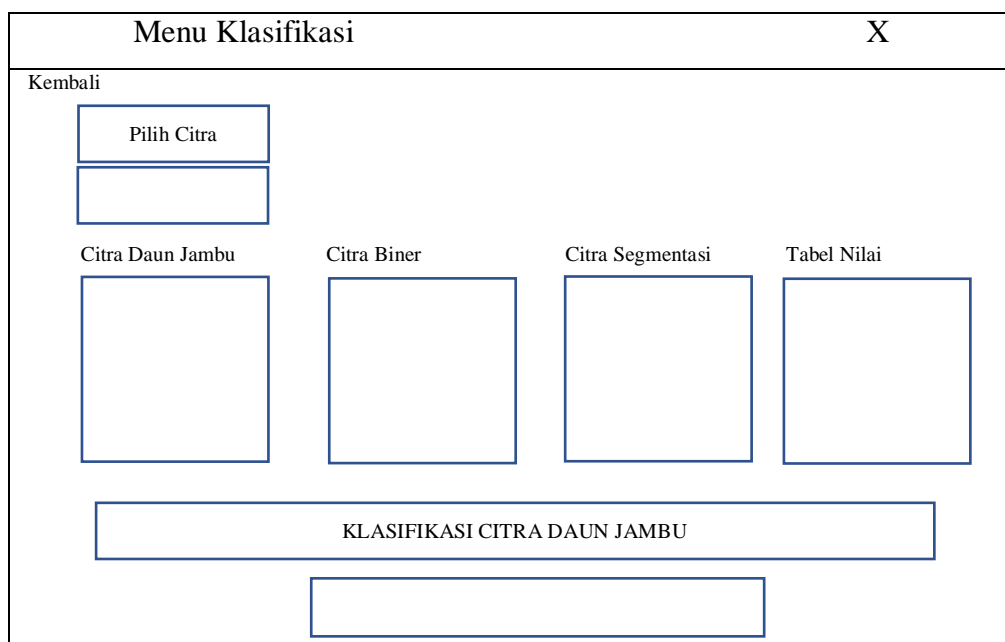
**Gambar 4.5** Rancangan Menu Utama

Berdasarkan pada gambar di atas, adapun dapat diberi keterangan sebagai berikut:

- a. *Button* yang berfungsi untuk menampilkan menu klasifikasi citra daun jambu.
- b. *Button* yang berfungsi untuk menampilkan menu informasi tentang penulis.

## 2. Rancangan Menu Klasifikasi

Menu klasifikasi adalah sebuah menu yang akan tampil ketika pengguna akan melakukan klasifikasi citra daun jambu. Adapun Berikut adalah rancangan klasifikasi dapat dilihat pada gambar di bawah:



**Gambar 4.6** Rancangan Klasifikasi



Berdasarkan gambar di atas, informasi berikut dapat diperoleh:

- a. *Button* yang berfungsi untuk memilih dan mencari citra daun jambu yang akan diuji.
  - b. *EditText* yang berfungsi menampilkan informasi nama dari citra daun jambu uji.
  - c. *Axes* yang berfungsi menampung data citra daun jambu yang telah dipilih.
  - d. *Axes* yang berfungsi menampung data citra daun jambu yang telah disegmentasi biner.
  - e. *Axes* yang berfungsi menampung data citra daun jambu yang telah disegmentasi akhir.
  - f. *Uitables* berfungsi untuk menampilkan nilai ekstraksi *PCA*.
  - g. *Button* yang berfungsi untuk memproses citra daun jambu yang akan diuji dengan metode *PCA* dan *KNN*.
  - h. *EditText* yang berfungsi menampilkan informasi nama dari citra daun jambu hasil klasifikasi.
  - i. *Menueditor* yang berfungsi untuk kembali pada menu utama.
3. Rancangan Menu Info Penulis

Menu info penulis adalah sebuah menu yang akan tampil ketika pengguna melihat info tentang penulis. Berikut adalah rancangan menu info penulis dapat dilihat pada gambar di bawah:

Menu Info	X
<p>NAMA MAHASIWA ATIKAH KHAIRIYAH HASIBUAN</p> <p>NIM 0701163078</p> <p>PROGRAM STUDI ILMU KOMPUTER</p> <div style="border: 1px solid black; width: fit-content; margin: 10px auto; padding: 5px 20px; text-align: center;">KEMBALI</div>	

**Gambar 4.7** Rancangan Menu Info Penulis

Berdasarkan pada gambar di atas, adapun dapat diberi keterangan sebagai berikut:

- a. *Button* yang berfungsi untuk kembali pada menu utama.

#### 4. Algoritma *Pseudocode*

*Pseudocode* adalah deskripsi tingkat tinggi informal dan ringkas dari algoritma pemrograman komputer yang menggunakan konvensi struktural bahasa pemrograman dan ditunjukkan untuk dibaca oleh manusia dan bukan oleh mesin..

```

program_gui_matlab
sistem_principal_component_analysis_dan_knearest_neighbor
_pada_citra_daun_jambu
begin
deklarasi :
    axes1, axes2, axes3 (gambar) : nilaipc : knn : akurasi (value);
fungsi nilaipc :
deklarasi :
    ciri_uji, coeff, PC1, PC2, score_uji:
deskripsi :
    ciri_ujiZ = zeros(jumlah_file, 16);
    for k = 1:jumlah_file
        ciri_ujiZ(k,:) = (ciri_uji(k,:) - muZ) ./ sigmaZ;
        score_uji = ciri_ujiZ * coeff;
        PC1 = score_uji(:, 1);
        PC2 = score_uji(:, 2);
    end;
deskripsi axes1, axes2, axes3;
    read = folder (.jpg);
    show axes1;
    rgb2gray(axes1) = show axes2;
    rgb = cat(3, r, g, b) = axes2
    show axes3

```

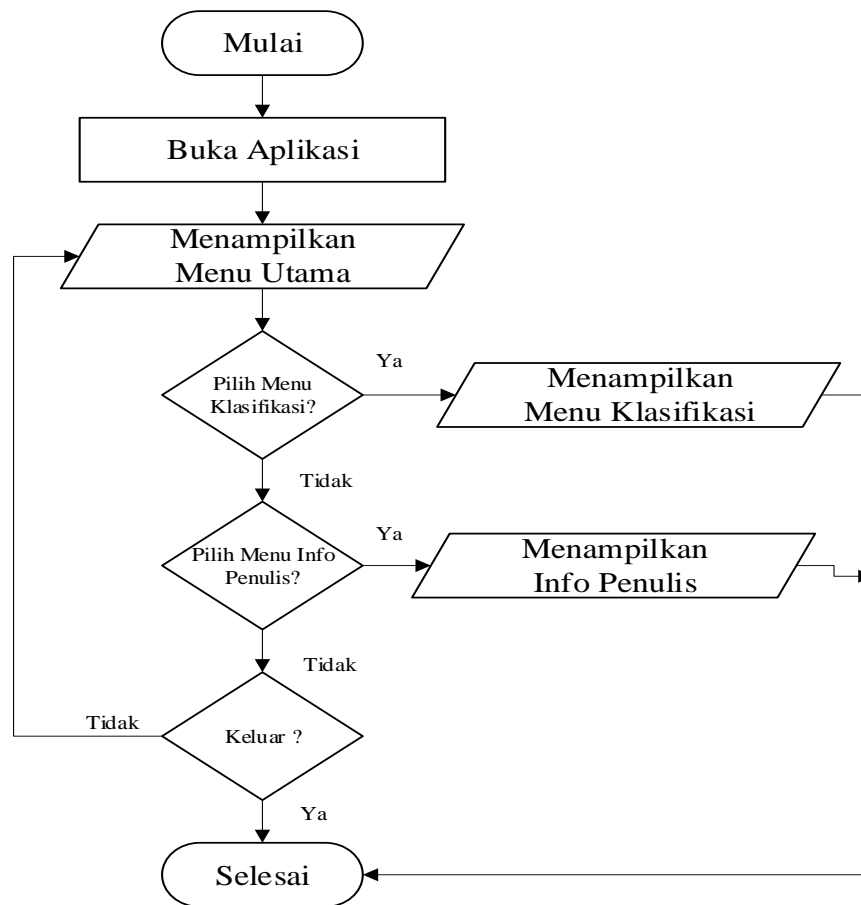
```
deskripsi knn :  
    fitcknn(input,target,[10 5],{'logsig','logsig'},'trainlm');  
  
deskripsi akurasi(value) :  
    akurasi = sum(m)/total_images*100;  
    end;
```

#### 4.1.4.1 Flowchart Sistem

Diagram alir sistem digunakan untuk menunjukkan alur proses dari sistem yang akan dibangun. Adapun *flowchart* sistem dibagi menjadi tiga bagian, yaitu *flowchart* menu utama, *flowchart* klasifikasi dan *flowchart* info penulis. Berikut adalah keseluruhan dari *flowchart* sistem yang akan dibangun:

##### 1. *Flowchart* Menu Utama

Diagram alir pada menu utama merupakan diagram dari alur proses. ketika pengguna berada pada menu utama. Berikut adalah rancangan dari *flowchart* menu utama:

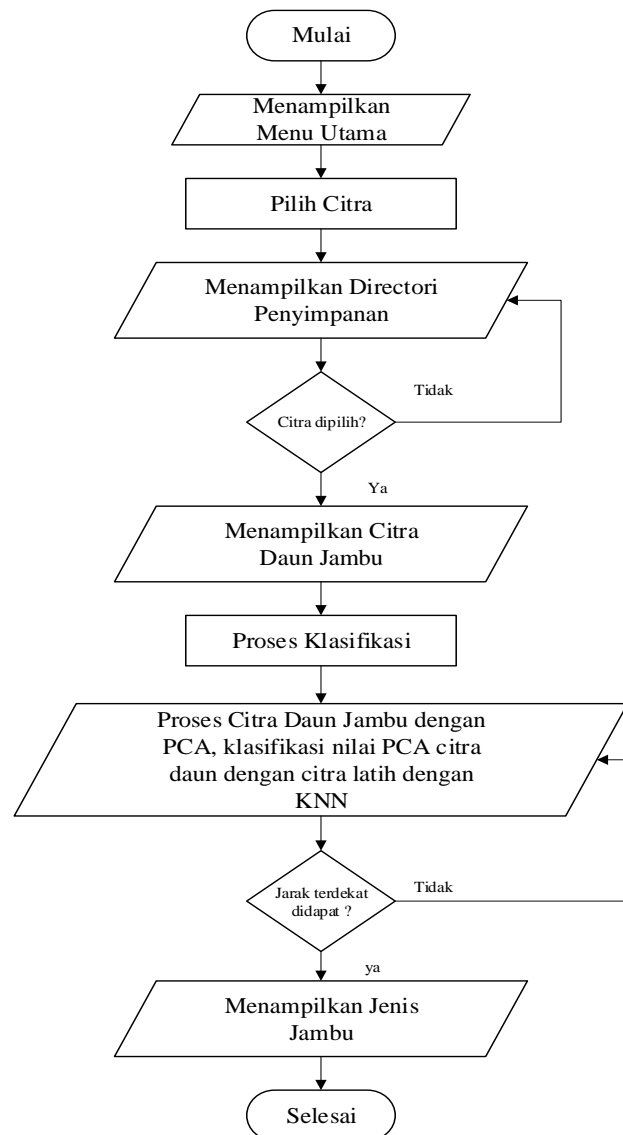


**Gambar 4.8** *Flowchart* Menu Utama

Berdasarkan pada gambar *flowchart* menu utama, dapat dijelaskan bahwa langkah pengguna yang membuka aplikasi dihadapkan dengan 2 menu yaitu menu klasifikasi yang digunakan untuk memproses klasifikasi citra daun jambu serta menu info penulis yang digunakan untuk menampilkan informasi tentang penulis skripsi ini.

## 2. *Flowchart* Klasifikasi

*Flowchart* klasifikasi adalah gambar alur proses ketika pengguna melakukan klasifikasi citra daun jambu. Berikut adalah rancangan dari *flowchart* klasifikasi:

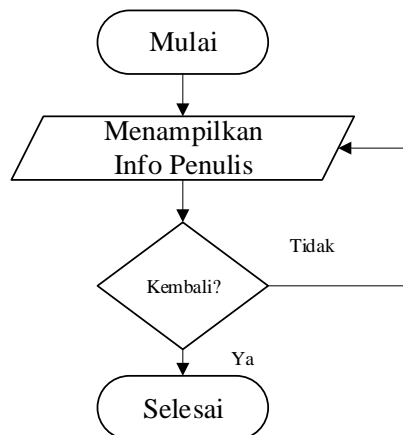


**Gambar 4.9** *Flowchart* Klasifikasi

Berdasarkan pada gambar *flowchart* citra klasifikasi, dapat dijelaskan bahwa langkah awal pengguna melakukan pemilihan citra daun jambu, kemudian diproses menggunakan metode PCA, hasil ekstraksi metode PCA diklasifikasi dengan jarak terdekat pada citra latih menggunakan metode KNN, sehingga menghasilkan jenis jambu yang sesuai dengan citra daun jambu inputan.

### 3. *Flowchart* Info Penulis

*Flowchart* penulis adalah gambar alur proses ketika pengguna membuka menu info penulis. Berikut adalah rancangan dari *flowchart* info penulis:



**Gambar 4.10** Flowchart Info Penulis

Berdasarkan pada gambar *flowchart* info penulis, dapat dijelaskan bahwa pengguna disajikan dengan informasi tentang penulis skripsi.

## 4.2 Hasil

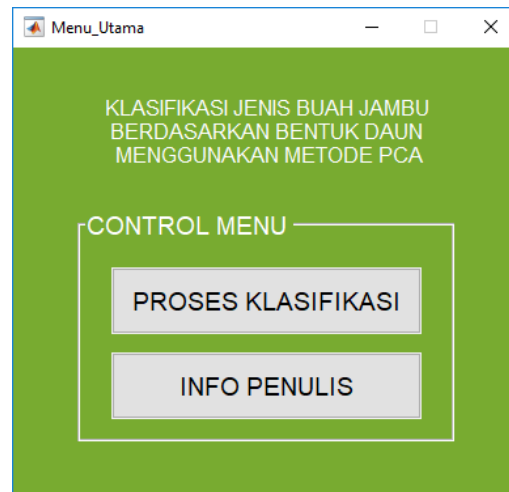
Berkenaan dengan hasil yang diperoleh dalam penelitian ini akan dibahas beberapa tahapan yaitu pengujian dan aplikasi, seperti pada gambar di bawah ini.

### 4.2.1 Pengujian

Berdasarkan dari proses tahapan hitungan manual dan rancangan program aplikasi, selanjutnya adalah melakukan tahapan pengujian sistem aplikasi yang telah dibangun menggunakan program Matlab. Berikut adalah tampilan program yang telah dibangun :

#### 1. Tampilan Program Menu Utama

Adapun tampilan program menu utama yang telah dibangun kedalam aplikasi menggunakan matlab dapat dilihat sebagai berikut ini:



**Gambar 4.11** Tampilan Menu Utama

## 2. Tampilan Program Menu Klasifikasi

Adapun tampilan program menu klasifikasi yang telah dibangun kedalam aplikasi menggunakan matlab dapat dilihat sebagai berikut ini:



**Gambar 4.12** Tampilan Menu Klasifikasi

## 3. Tampilan Program Menu Info Penulis

Adapun tampilan program menu info penulis yang telah dibangun kedalam aplikasi menggunakan matlab dapat dilihat sebagai berikut ini:



**Gambar 4.13** Tampilan Menu Info Penulis

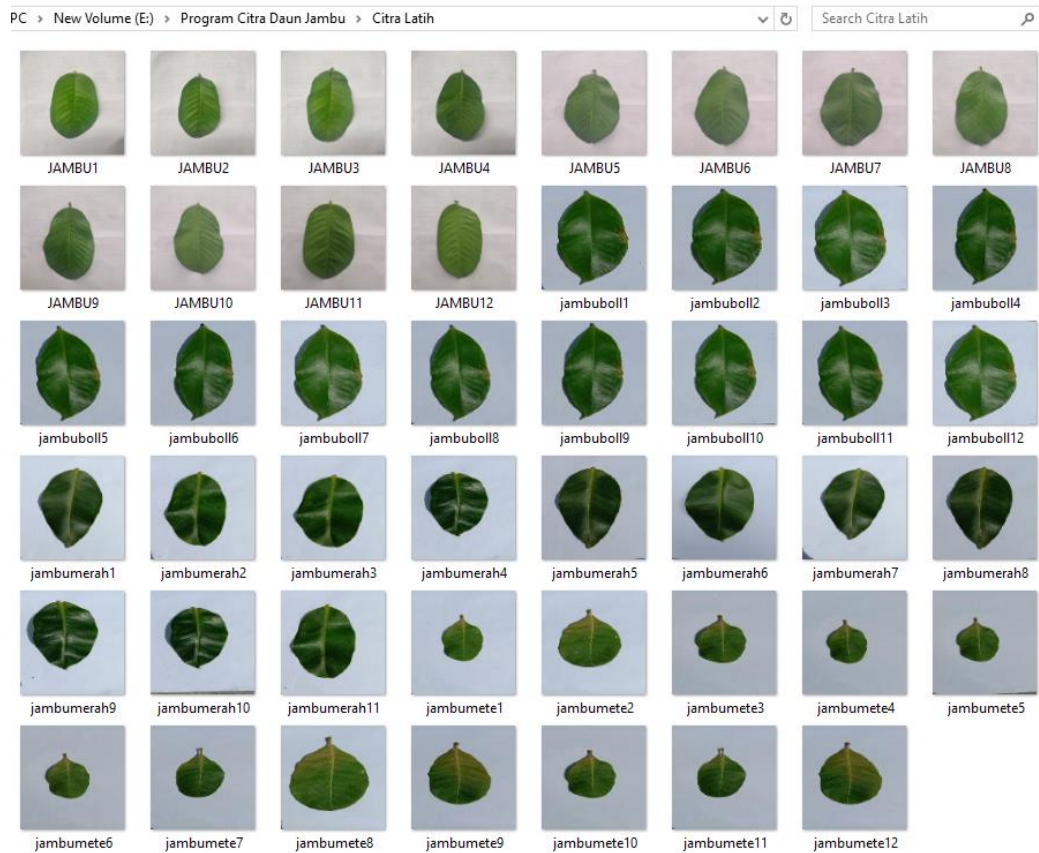
Adapun tahapan pada pengujian program ini adalah dengan menyiapkan data latih dan data uji berupa citra daun jambu yang terdiri dari 47 citra daun jambu sebagai data citra latih dan 16 citra daun jambu sebagai citra latih. Berikut adalah data lengkap dari citra daun jambu disajikan kedalam tabel di bawah ini:

**Tabel 4.13** Informasi Citra Latih dan Citra Uji

Data Citra	Jumlah	Jenis Jambu	Jumlah
Citra Latih	47 Citra	Jambu Biji	12
		Jambu Bol	12
		Jambu Merah	11
		Jambu Mete	12
Citra Uji	16 Citra	Jambu Biji	4
		Jambu Bol	4
		Jambu Merah	4
		Jambu Mete	4

Berdasarkan pada table informasi citra daun jambu, selanjutnya adalah menyiapkan data citra daun latih dan citra daun uji yang dimasukkan kedalam folder “Citra Latih” dan folder “Citra Uji” sebagai berikut:





**Gambar 4.14** Citra Daun Jambu Latih



**Gambar 4.15** Citra Daun Jambu Uji

Selanjutnya adalah melakukan klasifikasi jenis jambu berdasarkan citra daun uji yang telah disiapkan pada gambar 4.15.

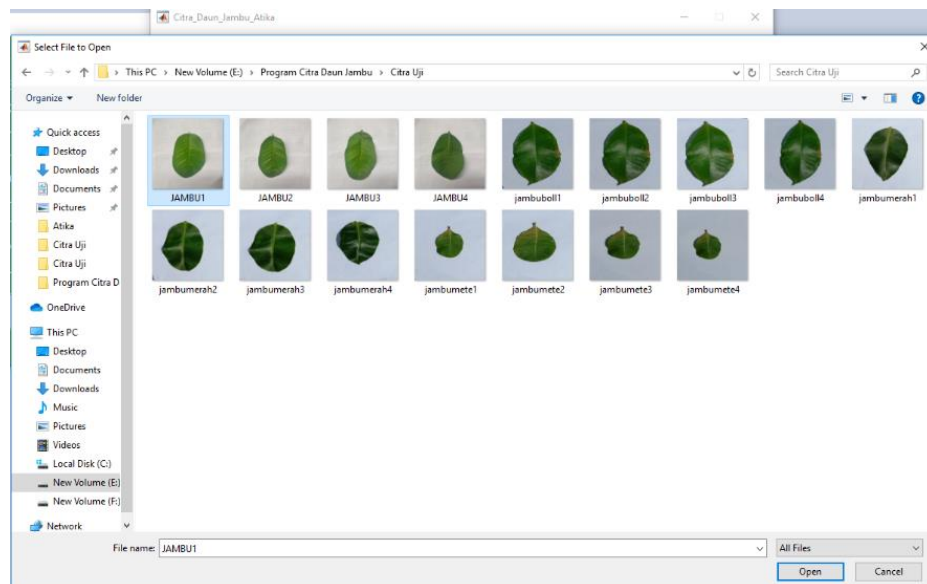
#### 1. Klasifikasi Citra Daun Jambu Biji

Adapun tahapannya pada sistem aplikasi adalah sebagai berikut:



**Gambar 4.16** Menu Proses Klasifikasi

Berdasarkan pada gambar di atas, untuk memulai proses klasifikasi jenis jambu berdasarkan bentuk citra daun jambu, pengguna dapat menekan *button* “Pilih Citra” sehingga menampilkan menu pencarian citra sebagai berikut ini:



**Gambar 4.17** Menu Pencarian Citra

Berdasarkan pada gambar di atas, pengguna memilih citra uji yang akan diklasifikasi, pada tahapan ini pengguna memilih citra “JAMBU1” atau citra jambu biji, kemudian pilih “*open*”. Adapun hasilnya sebagai berikut:



**Gambar 4.18** Citra Daun Jambu Dipilih

Berdasarkan pada gambar di atas, setelah dilakukan pemilihan citra daun jambu yang akan diklasifikasi, system dapat membaca informasi nama citra. Selanjutnya untuk memulai proses klasifikasi citra dapat dilakukan dengan menekan *button* “Klasifikasi Citra Daun Jambu” sehingga menghasilkan klasifikasi jenis jambu seperti pada gambar di bawah ini:



**Gambar 4.19** Citra Daun Jambu 1 Diklasifikasi Sebagai Jambu Biji

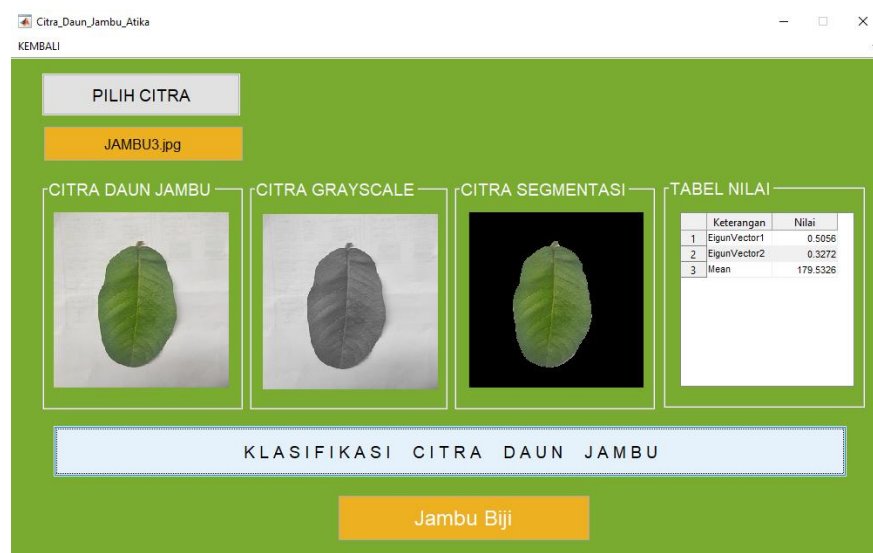
Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu biji berhasil diklasifikasi sebagai jenis “Jambu Biji”, sehingga klasifikasi sistem benar.

Berikut adalah klasifikasi untuk jenis daun jambu biji 2 dengan hasil sebagai berikut :



**Gambar 4.20** Citra Daun Jambu 2 Diklasifikasi Sebagai Jambu Biji

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu biji berhasil diklasifikasi sebagai jenis “Jambu Biji”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu biji 3 dengan hasil sebagai berikut :



**Gambar 4.21** Citra Daun Jambu 3 Diklasifikasi Sebagai Jambu Biji

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu biji berhasil diklasifikasi sebagai jenis “Jambu Biji”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu biji 4 dengan hasil sebagai berikut :

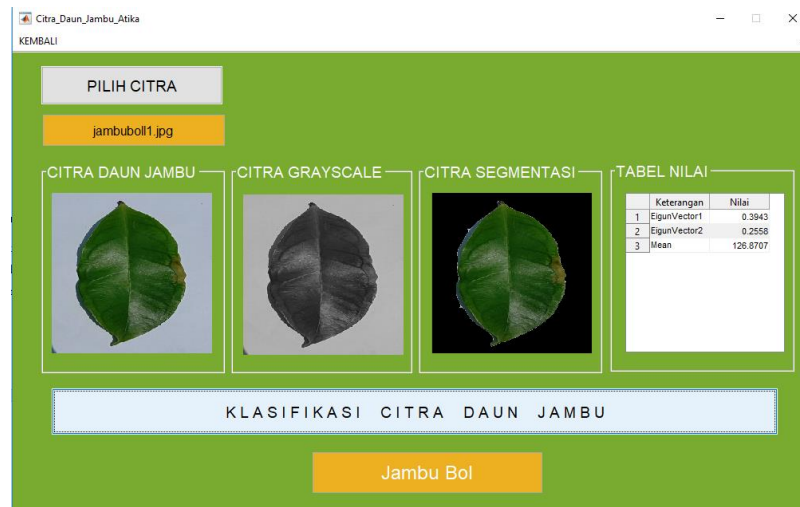


**Gambar 4.22** Citra Daun Jambu 4 Diklasifikasi Sebagai Jambu Biji

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu biji berhasil diklasifikasi sebagai jenis “Jambu Biji”, sehingga klasifikasi sistem benar.

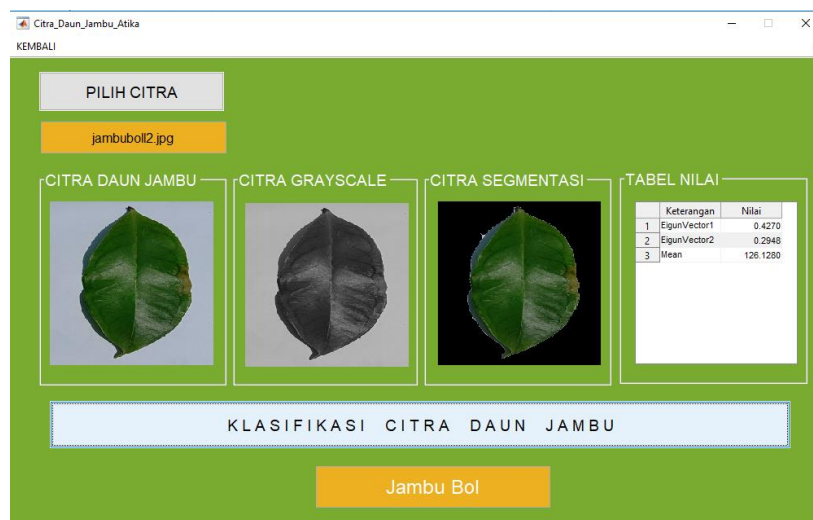
## 2. Klasifikasi Citra Daun Jambu Bol

Adapun tahapannya pada sistem aplikasi dilakukan dengan cara yang sama seperti pada proses sebelumnya, sehingga adapun hasil dari klasifikasi keseluruhan untuk citra daun jambu bol adalah sebagai berikut :



**Gambar 4.23** Citra Daun Jambu 1 Diklasifikasi Sebagai Jambu Bol

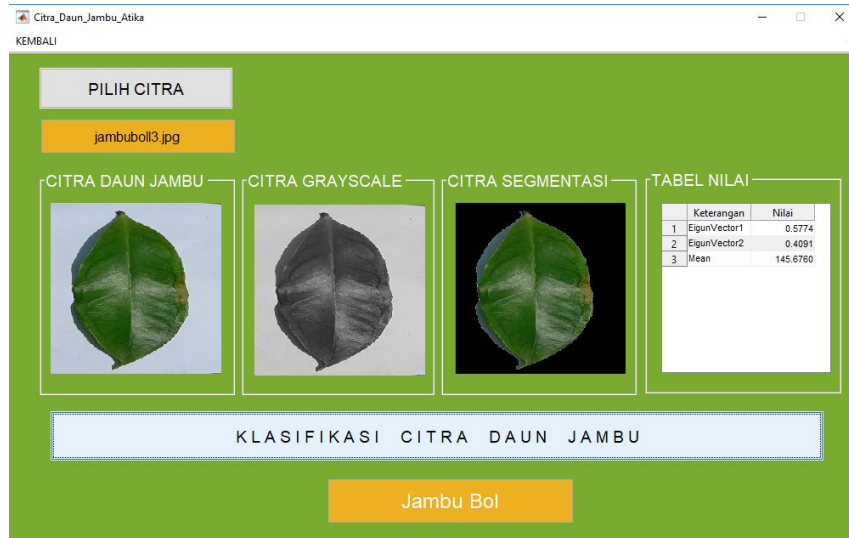
Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu bol berhasil diklasifikasi sebagai jenis “Jambu Bol”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu bol 2 dengan hasil sebagai berikut :



**Gambar 4.24** Citra Daun Jambu 2 Diklasifikasi Sebagai Jambu Bol

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu bol berhasil diklasifikasi sebagai jenis “Jambu Bol”, sehingga klasifikasi sistem benar.

Berikut adalah klasifikasi untuk jenis daun jambu bol 3 dengan hasil sebagai berikut :



**Gambar 4.25** Citra Daun Jambu 3 Diklasifikasi Sebagai Jambu Bol

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu bol berhasil diklasifikasi sebagai jenis “Jambu Bol”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu bol 4 dengan hasil sebagai berikut :



**Gambar 4.26** Citra Daun Jambu 4 Diklasifikasi Sebagai Jambu Bol

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu bol berhasil diklasifikasi sebagai jenis “Jambu Bol”, sehingga klasifikasi sistem benar.

### 3. Klasifikasi Citra Daun Jambu Merah

Adapun tahapanya pada sistem aplikasi dilakukan dengan cara yang sama seperti pada proses sebelumnya, sehingga adapun hasil dari klasifikasi keseluruhan untuk citra daun jambu merah adalah sebagai berikut :



**Gambar 4.27** Citra Daun Jambu 1 Diklasifikasi Sebagai Jambu Merah

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu merah tidak berhasil diklasifikasi sebagai jenis “Jambu Merah” melainkan menjadi “Jambu Bol”, sehingga klasifikasi sistem tidak benar. Berikut adalah klasifikasi untuk jenis daun jambu merah 2 dengan hasil sebagai berikut :





**Gambar 4.28** Citra Daun Jambu 2 Diklasifikasi Sebagai Jambu Merah

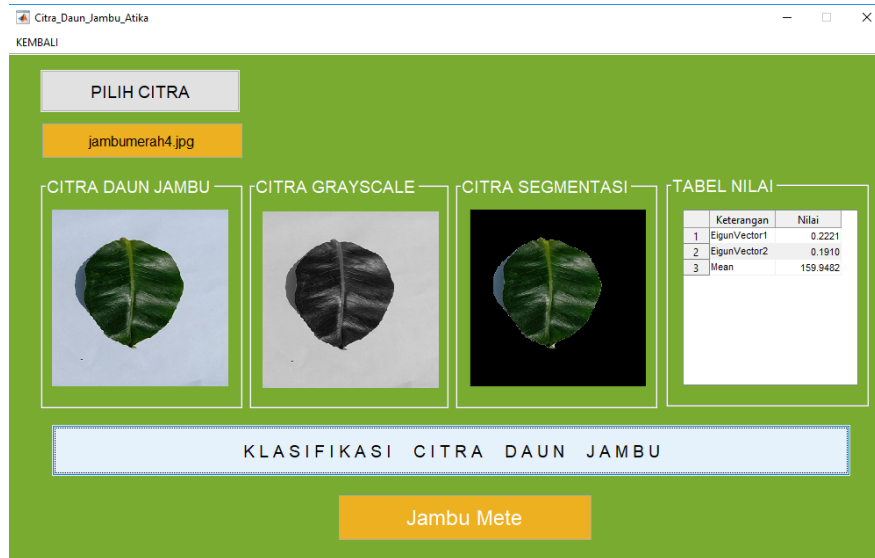
Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu merah berhasil diklasifikasi sebagai jenis “Jambu Merah”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu merah 3 dengan hasil sebagai berikut :



**Gambar 4.29** Citra Daun Jambu 3 Diklasifikasi Sebagai Jambu Merah

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu merah berhasil diklasifikasi sebagai jenis “Jambu Merah”, sehingga klasifikasi sistem

benar. Berikut adalah klasifikasi untuk jenis daun jambu merah 4 dengan hasil sebagai berikut :



**Gambar 4.30** Citra Daun Jambu 4 Diklasifikasi Sebagai Jambu Merah

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu merah tidak berhasil diklasifikasi sebagai jenis “Jambu Merah” melainkan menjadi “Jambu Mete”, sehingga klasifikasi sistem tidak benar.

#### 4. Klasifikasi Citra Daun Jambu Mete

Adapun tahapanya pada sistem aplikasi dilakukan dengan cara yang sama seperti pada proses sebelumnya, sehingga adapun hasil dari klasifikasi keseluruhan untuk citra daun jambu mete adalah sebagai berikut :



**Gambar 4.31** Citra Daun Jambu 1 Diklasifikasi Sebagai Jambu Mete

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu mete berhasil diklasifikasi sebagai jenis “Jambu Mete”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu mete 2 dengan hasil sebagai berikut :



**Gambar 4.32** Citra Daun Jambu 2 Diklasifikasi Sebagai Jambu Mete

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu mete tidak berhasil diklasifikasi sebagai jenis “Jambu Mete” melainkan menjadi “Jambu

Merah”, sehingga klasifikasi sistem tidak benar. Berikut adalah klasifikasi untuk jenis daun jambu mete 3 dengan hasil sebagai berikut :



**Gambar 4.33** Citra Daun Jambu 3 Diklasifikasi Sebagai Jambu Mete

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu mete berhasil diklasifikasi sebagai jenis “Jambu Mete”, sehingga klasifikasi sistem benar. Berikut adalah klasifikasi untuk jenis daun jambu mete 4 dengan hasil sebagai berikut :



**Gambar 4.34** Citra Daun Jambu 4 Diklasifikasi Sebagai Jambu Mete

Berdasarkan pada gambar di atas, proses klasifikasi berhasil diterapkan pada sistem aplikasi, dimana dengan citra uji daun jambu jenis jambu mete berhasil diklasifikasi sebagai jenis “Jambu Mete”, sehingga klasifikasi sistem benar. Sehingga hasil keseluruhan dapat dilihat pada hasil pengujian sistem.

Berdasarkan dari hasil pengujian klasifikasi dengan 16 citra uji yang terdiri dari 4 jenis jambu didapati hasil sebagai berikut:

**Tabel 4.14** Hasil Pengujian

No	Nama Citra	Jenis Jambu	Hasil Klasifikasi	Keterangan
1	Jambu1.jpg	Jambu Biji	Jambu Biji	Benar
2	Jambu2.jpg	Jambu Biji	Jambu Biji	Benar
3	Jambu3.jpg	Jambu Biji	Jambu Biji	Benar
4	Jambu4.jpg	Jambu Biji	Jambu Biji	Benar
5	Jambuboll1.jpg	Jambu Bol	Jambu Bol	Benar
6	Jambuboll2.jpg	Jambu Bol	Jambu Bol	Benar
7	Jambuboll3.jpg	Jambu Bol	Jambu Bol	Benar
8	Jambuboll4.jpg	Jambu Bol	Jambu Bol	Benar
9	Jambumerah1.jpg	Jambu Merah	Jambu Bol	Salah
10	Jambumerah2.jpg	Jambu Merah	Jambu Merah	Benar
11	Jambumerah3.jpg	Jambu Merah	Jambu Merah	Benar
12	Jambumerah4.jpg	Jambu Merah	Jambu Mete	Salah
13	Jambumete1.jpg	Jambu Mete	Jambu Mete	Benar
14	Jambumete2.jpg	Jambu Mete	Jambu Merah	Salah
15	Jambumete3.jpg	Jambu Mete	Jambu Mete	Benar
16	Jambumete4.jpg	Jambu Mete	Jambu Mete	Benar

Berdasarkan pada tabel hasil pengujian di atas, didapati 13 citra uji yang berhasil diklasifikasi dengan benar dan 3 citra uji yang diklasifikasi dengan salah. Adapun selanjutnya menghitung tingkat akurasi berdasarkan citra uji yang dipakai. Adapun rumusnya adalah sebagai berikut :

$$Akurasi = \frac{Jumlah\ Klasifikasi\ Benar}{Jumlah\ Data} \times 100\%$$

$$Akurasi = \frac{13}{16} \times 100\% = 81,25 \%$$

Berdasarkan dari hasil uji akurasi, didapatkan nilai akurasi sebesar 81,25% untuk proses klasifikasi jenis jambu berdasarkan citra daun jambu sebanyak 16 data.

#### **4.2.2 Penerapan**

Aplikasi/penggunaan sistem ini adalah untuk mengklasifikasikan daun jambu biji berdasarkan bentuknya. Daun diklasifikasikan menurut pola dan bentuk daunnya. Dengan menggunakan pengolahan citra digital, dimungkinkan untuk mengenali citra seperti penglihatan manusia. Penerapan sistem ini dapat diterapkan oleh pelajar dan mahasiswa yang ingin mengetahui klasifikasi daun jambu biji berdasarkan bentuk daunnya. Sistem mana yang akan memudahkan peneliti selanjutnya dan mahasiswa yang ingin penelitian dalam mengklasifikasikan daun sehingga perbedaan dan prosesnya dapat diketahui dengan

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berikut kesimpulan yang dapat diambil dari pengujian yang dilakukan dengan mengklasifikasikan jenis jambu biji berdasarkan bentuk daun jambu biji menggunakan *Principal Component Analysis* dan *K-Nearest Neighborhood Analysis* sebagai berikut:

1. Dengan menggunakan metode PCA untuk melakukan ekstraksi citra daun jambu yang ditransformasi tetap berhasil mempertahankan akurasi klasifikasi dengan metode KNN.
2. Metode KNN dapat melakukan klasifikasi dengan mencari nilai terdekat citra uji hasil transformasi dengan citra latih yang disimpan.
3. Proses klasifikasi dengan data uji sebanyak 16 citra berhasil mendapatkan nilai akurasi sebesar 81,25 %, dimana nilai akurasi ini dapat dikatakan baik.
4. Berdasarkan hasil pengujian, akurasi klasifikasi tentu akan berubah dengan semakin banyaknya data yang diuji.

#### **5.2 Saran**

Untuk mengembangkan sistem yang lebih baik dan menguji keakuratan klasifikasi jenis jambu biji berdasarkan bentuk citra daun jambu biji, penulis dapat memberikan saran sebagai berikut:

1. Untuk meningkat akurasi klasifikasi, maka perlu adanya perbandingan metode ekstraksi PCA dengan metode ekstraksi yang lain.
2. Perlu ditambahkan ada uji dan data latih untuk mendapatkan hasil klasifikasi berupa akurasi yang lebih optimal.
3. Untuk pengembangan system yang lebih baik, perlu adanya sistem yang *real time* dalam melakukan klasifikasi jenis jambu serta dapat ditambahkan fitur pemilihan folder citra uji dan folder citra latih secara sistem.

## DAFTAR PUSTAKA

Al-Qur'an.

Abdi, H dan Williams LJ., 2010. *Principal Component Analysis*. Willey Interdisciplinary reviews.

Andono, Pulung Nurtantio. 2017. *Pengolahan Citra Digital*. Edisi 1. Yogyakarta.

Budi, A., & Maulana, H. (2016). Pengenalan Citra Wajah Sebagai Identifier Menggunakan Metode Principal Component Analysis (PCA). *Jurnal Teknik Informatika*, 9(2).

D. R. Taningrum, B. Hidayat, And Y. S. Hariyani, "Sistem Pengidentifikasian Plat Nomor Kendaraan Mobil Menggunakan *Principal Component Analysis* Dan Klasifikasi K-NN".*E-Proceeding Eng.*, **Vol. 3**, No. 2, Pp. 1868–1876, 2016.

Fadhillah, A. N., Novamizanti, L., & Atmaja, R. D. (2015). Analisis dan Implementasi Klasifikasi *K-nearest Neighbor* (K-NN) Pada Sistem Identifikasi Biometrik Telapak Kaki Manusia. *eProceedings of Engineering*, 2(2).

Furqan, M., Embong, A., Awang, S., Purnami, S. W., & Sembiring, S. (2009). *Smooth support vector machine for face recognition using Principal Component Analysis*.

Furqan, M., Sriani., Harahap, L.S., 2020, Klasifikasi Daun Bugenvil Menggunakan *Gray Level Co-Occurrence Matrix* Dan *K-Nearest Neighbor*. *Jurnal CoreIT*, **Vol.6**.

Gonzalez, R.C., Woods, R.E. (2008). *Digital Image Processing Third Edition*. New Jersey: Prentice Hall.

Gulhane, V. A., & Kolekar, M. H. (2014, December). *Diagnosis Of Diseases On Cotton Leaves Using Principal Component Analysis Classifier*. In *2014 Annual IEEE India Conference (INDICON)* (pp. 1-5). IEEE

Han, J., & Kamber, M. 2006. *Data Mining Concept and Tehniques*. San Fransisco: Morgan Kauffman

Handaya, S., (2013), *Kandungan Kulit Batang dan Daun*, Fakultas Perikanan dan Ilmu Kelautan, Institut Pertanian Bogor.



- Herfina, H. (2013). *Pengenalan Pola Bentuk Bunga Menggunakan Principle Component Analysis dan K-NN*. SEMNASTEKNOMEDIA ONLINE, 1(1),07-25.
- Hutabri, R. W., & Fu'adah, R. Y. N. (2018). Perancangan Sistem Deteksi Katarak Menggunakan Metode Principal Component Analysis (PCA) Dan Knearest Neighbor (K-NN) Analisis Performansi Sistem Deteksi Katarak. *Prosiding SENIATI*, 321-327.
- Ide, Pangkalan., 2011. *Health Secret Of Turmeric*. Jakarta : PT Elex Media Komputindo.
- Isnanto, R. R., Ajub A. Z., dan Eko D. W. 2015. Analisis Kinerja Pengenalan Telapak Tangan Menggunakan Ekstraksi Ciri *Principal Component Analysis* (PCA) dan *Overlapping Block*. *Scientific Journal of Informatics*, Vol.2 No.2.
- Kusumadewi S, dkk. 2009. *Informatika Kesehatan*. Yogyakarta: Graha Ilmu
- Kusrini, Luthfi Taufiq Emha, (2009), *Algoritma Data Mining*, Penerbit Andi, Yogyakarta
- Maisa Hana. 2014. "Sistem Identifikasi *Biometrik Finger Knuckle Print* Menggunakan *Histogram Equalization dan Principal Component Analysis*.
- McAndrew Alasdair, 2004, *An Introduction to Digital Image Processing with Matlab, School Of Computer Science and Mathematics, Victoria University of Technology*
- Munir, Rinaldi. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung : Informatika.
- Muntasa, A., & Purnonmo, M. H. 2010. *Konsep Pengolahan Citra Digital dan Ekstrasi Fitur*. Yogyakarta: Graha Ilmu.
- Nugraheny, Dwi 2013, Hasil Ekstrasi Algoritma *Principal Component Analysis* (PCA) Untuk Pengenalan Wajah, Teknik Informatika STTA, Yogyakarta.
- Priyawati. "Teknik Pengolahan Citra Digital Berdomain Spasial Untuk Peningkatan Citra Sinar X " vol. II, pp. 44-50,20
- Putra, Darma. (2010). *Pengolahan Citra Digital*. Yogyakarta: Penerbit ANDI.
- Salamah N. A. (2019). *Pengolahan Citra Untuk Pengenalan Wajah Manusia Menggunakan Principal Component Analysis dan Euclidean Distance*. Jurnal Teknik Informatika, 9(2).

- Suprapti, M. L. 2004. *Jelly Jambu Mete*. Kanisius, Yogyakarta.
- Sary, D. M. (2019). Studi Penggunaan Metode *Principal Component Analysis* Dan *Euclidean Distance* Dalam Pengenalan Sistem Kualitas Buah Tomat.
- Soeharso & Retnoningsih, A. (2011). *Kamus Besar Bahasa Indonesia*. Semarang: Widya Karya
- Sutoyo, T., Mulyanto, Edy., Suhartono, Vincent., Nurhayati, O.Dwi., Wijanarto. (2009). *Teori Pengolahan Citra Digital*. Yogyakarta: Penerbit ANDI.
- Tanri, Abeng, 2006. *Profesi Manajemen*, Jakarta: Gramedia Pustaka Utama.
- Turk, M., Pentland, A 1991. *Eigenfaces for Recognition*. *Massachusetts Institute Of Technology. Journal od Cognitive Neuroscience*, Vol. 3, No. 1 / 1991.
- Utami, Ema dan Wulanningrum. 2014. Penggunaan *Principal Component Analysis* dan *Euclidean Distance* untuk Identifikasi Citra Tanda Tangan. STIMIK AMIKOM. Yogyakarta.
- Wahyumianto, A, Purnama K.E, Christyowidiasmoro. 2009. *Identifikasi Tumbuhan Berdasarkan Tulang Daun*. Institut Teknologi Sepuluh November.
- Zein, A. (2018, July). Pendeteksian Multi Wajah Dan *Recognition* Secara *Real Time* Menggunakan Metoda *Principal Component Analysis* (PCA) Dan *Eigenface*. In *ESIT* (Vol. 12, No. 1, pp. 1-7).

## LAMPIRAN-LAMPIRAN

### Lampiran 1

Hasil uji citra daun jambu pada “Klasifikasi Jenis Buah Jambu Berdasarkan Daun Menggunakan Metode *Principal Component Analysis*”

No	Nama Citra	Jenis Jambu	Hasil Klasifikasi	Keterangan
1	Jambu1.jpg	Jambu Biji	Jambu Biji	Benar
2	Jambu2.jpg	Jambu Biji	Jambu Biji	Benar
3	Jambu3.jpg	Jambu Biji	Jambu Biji	Benar
4	Jambu4.jpg	Jambu Biji	Jambu Biji	Benar
5	Jambuboll1.jpg	Jambu Bol	Jambu Bol	Benar
6	Jambuboll2.jpg	Jambu Bol	Jambu Bol	Benar
7	Jambuboll3.jpg	Jambu Bol	Jambu Bol	Benar
8	Jambuboll4.jpg	Jambu Bol	Jambu Bol	Benar
9	Jambumerah1.jpg	Jambu Merah	Jambu Bol	Salah
10	Jambumerah2.jpg	Jambu Merah	Jambu Merah	Benar
11	Jambumerah3.jpg	Jambu Merah	Jambu Merah	Benar
12	Jambumerah4.jpg	Jambu Merah	Jambu Mete	Salah
13	Jambumete1.jpg	Jambu Mete	Jambu Mete	Benar
14	Jambumete2.jpg	Jambu Mete	Jambu Merah	Salah
15	Jambumete3.jpg	Jambu Mete	Jambu Mete	Benar
16	Jambumete4.jpg	Jambu Mete	Jambu Mete	Benar

## Lampiran 2

Tools Matlab untuk Klasifikasi Jenis Jambu Berdasarkan Daun Menggunakan Metode *Principal Component Analysis*:

Tools	Fungsi
Menu_Utama.fig	Menampilkan Tampilan Awal Sistem
Menu_Info.fig	Menampilkan Tampilan Menu InfoPenulis
Citra_Daun_Jambu.fig	Menampilkan Tampilan Sistem Klasifikasi
Pushbutton1	Menampilkan Tampilan Sistem Klasifikasi
Pushbutton2	Menampilkan Tampilan Menu InfoPenulis
Pushbutton3	Mengambil Gambar
Pushbutton4	Proses Klasifikasi Citra Daun Jambu
Axes1	Menampilkan Citra Yang Dipilih
Axes2	Menampilkan Citra Grayscale
Axes3	Menampilkan Citra Segmentasi
Uitable1	Menampilkan Tabel Nilai
Edit1	Menampilkan Nama File Citra
Edit2	Menampilkan Hasil Citra Klasifikasi
Pushbutton5	Menu Kembali

### Lampiran 3

Kode program matlab untuk “Klasifikasi Jenis Jambu Berdasarkan Daun Menggunakan Metode *Principal Component Analysis*” :

```
CITRA DAUN JAMBU ATIKA

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
[nama_daun,nama_path] = uigetfile({'*.*'});
    % menampilkan citra yang dipilih pada box axes1
if ~isequal(nama_daun,0)
    citra_daun_jambu = imread(fullfile(nama_path,nama_daun));
    axes(handles.axes1)
    imshow(citra_daun_jambu)

    % menampilkan nama file citra pada edit1
    set(handles.edit1,'String',nama_daun)
    handles.citra_daun_jambu = citra_daun_jambu;
    guidata(hObject,handles)
else
    return
end

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

%melakukan perhitungan parameter citra daun jambu
I = handles.citra_daun_jambu;
J = I(:,:,1);
K = im2bw(J,.6);
L = imcomplement(K);
str = strel('disk',5);
M = imclose(L,str);
N = imfill(M,'holes');
O = bwareaopen(N,5000);
stats = regionprops(O,'Area','Perimeter','Eccentricity');
area = stats.Area;
perimeter = stats.Perimeter;
metric = 4*pi*area/(perimeter^2);
eccentricity = stats.Eccentricity;

input = [metric;eccentricity];
```

```

load net
%merubah citra daun jambu menjadi grayscale pada box axes2
output = round(sim(net,input));
grayscale = rgb2gray (I);
axes(handles.axes2)
imshow(grayscale)

%melakukan segmentasi citra daun jambu berdasarkan bentuk daunnya
pada box
%axes3
R = I(:,:,1);
G = I(:,:,2);
B = I(:,:,3);
R(~O) = 0;
G(~O) = 0;
B(~O) = 0;
RGB = cat(3,R,G,B);
axes(handles.axes3)
imshow(RGB)

%Menentukan nilai eigun (PC) citra daun jambu
CirIMEAN = mean2(grayscale);
EV1 = mean2(R) / 47';
EV2 = mean2(B) / 47';
data2 = cell(3,2);
data2{1,1} = 'EigenVector1';
data2{2,1} = 'EigenVector2';
data2{3,1} = 'Mean';
data2{1,2} = EV1;
data2{2,2} = EV2;
data2{3,2} = CirIMEAN;
ciri_ujiZ = zeros(jumlah_file,16);
for k = 1:jumlah_file
    ciri_ujiZ(k,:)=(ciri_uji(k,:)- muZ)./sigmaZ;
    score_uji = ciri_ujiZ*coeff;
    PC1 = score_uji(:,1);
    PC2 = score_uji(:,2);
end;

set(handles.uitable1,'Data',data2,'ForegroundColor',[0 0 0])

%melakukan klasifikasi berdasarkan nilai PCA dengan jarak KNN pada
dataset
if output == 1
    PCA = 'Jambu Biji';
elseif output == 2
    PCA = 'Jambu Bol';
elseif output == 3
    PCA = 'Jambu Merah';
elseif output == 4
    PCA = 'Jambu Mete';
end

set(handles.edit2,'String',PCA)

```

```

% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% -----
----
function Untitled_1_Callback(hObject, eventdata, handles)
% hObject    handle to Untitled_1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
close;
guidata(Menu_Utama);

MENU INFO

function varargout = Menu_Info(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Menu_Info_OpeningFcn, ...
                  'gui_OutputFcn',  @Menu_Info_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB

```

```

% handles      structure with handles and user data (see GUIDATA)
close;
guidata(Menu_Utama);

MENU UTAMA
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',           mfilename, ...
                  'gui_Singleton',     gui_Singleton, ...
                  'gui_OpeningFcn',    @Menu_Utama_OpeningFcn, ...
                  'gui_OutputFcn',     @Menu_Utama_OutputFcn, ...
                  'gui_LayoutFcn',     [], ...
                  'gui_Callback',      []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton1 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
close;
guidata(Citra_Daun_Jambu_Atika);

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton2 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
close;
guidata(Menu_Info);

PELATIHAN

clc; clear; close all;

image_folder = 'Citra Latih';
filenames = dir(fullfile(image_folder, '*.jpg'));
total_images = numel(filenames);

area = zeros(1,total_images);
perimeter = zeros(1,total_images);
metric = zeros(1,total_images);
eccentricity = zeros(1,total_images);

```



```

for n = 1:total_images
    full_name= fullfile(image_folder, filenames(n).name);
    I = imread(full_name);
    J = I(:,:,1);
    K = im2bw(J,.6);
    L = imcomplement(K);
    str = strel('disk',5);
    M = imclose(L,str);
    N = imfill(M,'holes');
    O = bwareaopen(N,5000);
    stats = regionprops(O,'Area','Perimeter','Eccentricity');
    area(n) = stats.Area;
    perimeter(n) = stats.Perimeter;
    metric(n) = 4*pi*area(n)/(perimeter(n)^2);
    eccentricity(n) = stats.Eccentricity;
end

input = [metric;eccentricity];
target = zeros(1,47);
target(:,1:12) = 1;
target(:,13:23) = 2;
target(:,24:35) = 3;
target(:,36:47) = 4;

net = fitcknn(input,target,[10 5],{'logsig','logsig'},'trainlm');
net.trainParam.epochs = 1000;
net.trainParam.goal = 1e-6;
net = train(net,input,target);
output = round(sim(net,input));
save net.mat net

[m,n] = find(output==target);
akurasi = sum(m)/total_images*100

PENGUJIAN
clc; clear; close all;

image_folder = 'Citra Uji';
filenames = dir(fullfile(image_folder, '*.jpg'));
total_images = numel(filenames);

area = zeros(1,total_images);
perimeter = zeros(1,total_images);
metric = zeros(1,total_images);
eccentricity = zeros(1,total_images);

for n = 1:total_images
    full_name= fullfile(image_folder, filenames(n).name);
    I = imread(full_name);
    J = I(:,:,1);
    K = im2bw(J,.6);
    L = imcomplement(K);
    str = strel('disk',5);
    M = imclose(L,str);
    N = imfill(M,'holes');

```

```
O = bwareaopen(N,5000);
stats = regionprops(O,'Area','Perimeter','Eccentricity');
area(n) = stats.Area;
perimeter(n) = stats.Perimeter;
metric(n) = 4*pi*area(n)/(perimeter(n)^2);
eccentricity(n) = stats.Eccentricity;
end

input = [metric;eccentricity];
target = zeros(1,16);
target(:,1:4) = 1;
target(:,5:8) = 2;
target(:,9:12) = 3;
target(:,13:16) = 4;

load net
output = round(sim(net,input));

[m,n] = find(output==target);
akurasi = sum(m)/total_images*100
```

### **LAMPIRAN 3**



**DAFTAR RIWAYAT HIDUP**  
**(*CURRICULUM VITAE*)**



Nama : Atikah Khairiyah Hasibuan  
NIM : 0701163078  
Tempat/Tanggal Lahir : Gunung Kidul, 14 Agustus 1998  
Jenis Kelamin : Perempuan  
Alamat : Jl. Rahmadsyah Gg Dame No.487D Medan  
Agama : Islam  
Status Nikah : Belum Menikah  
No. HP : 082294822808  
Nama Orang Tua :  
    Ayah : Aswar  
    Ibu : Suliyanti  
Pendidikan Formal :  
2003-2009 : SD Negeri 5 Tambun Selatan Bekasi  
2009-2013 : SMP Negeri 5 Tambun Selatan Bekasi  
2013-2015 : SMK 10 November Tambun Selatan Bekasi  
2016-2020 : Universitas Islam Negeri Sumatera Utara