

**PENGENALAN POLA CITRA DENGAN EKSTRAKSI *GLCM* DAN
KLASIFIKASI *K-MEANS* UNTUK MENDETEKSI PENYAKIT PADA
DAUN CENGKEH**

SKRIPSI



Oleh:

HASBULLAH

2018020223

TEKNIK INFORMATIKA

KONSENTRASI REKAYASA PERANGKAT LUNAK

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) HANDAYANI
MAKASSAR**

2021

**PENGENALAN POLA CITRA DENGAN EKSTRAKSI *GLCM* DAN
KLASIFIKASI *K-MEANS* UNTUK MENDETEKSI PENYAKIT PADA
DAUN CENGKEH**

SKRIPSI

Diajukan sebagai salah satu syarat untuk mencapai gelar Sarjana Komputer

Jurusan Teknik Informatika

Disusun dan diajukan oleh:

HASBULLAH

2018020223

**JURUSAN TEKNIK INFORMATIKA
KONSENTRASI REKAYASA PERANGKAT LUNAK**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) HANDAYANI**

MAKASSAR

2021

HALAMAN PERSETUJUAN SEMINAR UJIAN SKRIPSI

Judul : Pengenalan Pola Citra Dengan Ekstraksi *GLCM* Dan Klasifikasi
K-Means Untuk Mendeteksi Penyakit Pada Daun Cengkeh

HASBULLAH

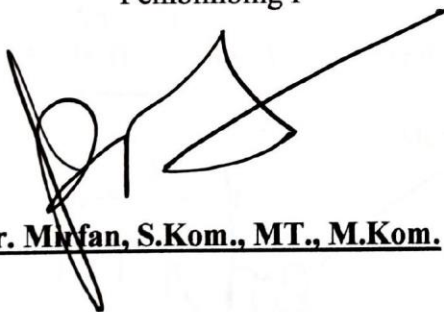
2018020223

TEKNIK INFORMATIKA

Telah diperiksa dan disetujui untuk diajukan sebagai salah satu syarat untuk menempuh Ujian Skripsi guna memperoleh gelar Sarjana Komputer.

Makassar, 07 April 2021

Pembimbing I



Ir. Mirfan, S.Kom., MT., M.Kom.

Pembimbing II



Nurilmiyanti Wardhani, S.Kom., MT.

MENGETAHUI,
KETUA JURUSAN TEKNIK INFORMATIKA



Ir. Billy Eden Wiliam Asrul, S.Kom, MT

HALAMAN PENGESAHAN
PENGENALAN POLA CITRA DENGAN EKSTRAKSI GLCMDAN
KLASIFIKASI K-MEANS UNTUK MENDETEKSI PENYAKIT PADA DAUN
CENGKEH

Oleh :
HASBULLAH
2018020223

Telah dipertahankan di hadapan sidang Penguji Skripsi STMIK Handayani
pada Hari Kamis, Tanggal 07 April 2021
dan dinyatakan telah memenuhi syarat memperoleh gelar Sarjana Komputer

Tim Penguji :

1. Ir. Mirfan, S.Kom., MT., M.Kom.
Ketua sidang
2. Nurilmiyanti Wardhani, S.Kom., MT.
Sekretaris Sidang
3. Drs. H. Abd Rochman, M.Ag., M.Kom.
Anggota
4. Dra. Najirah Umar, S.Kom., MT.
Anggota
5. Ir. Billy Eden William Asrul, S.Kom., MT.
Anggota

Tanda Tangan

1.
2.
3.
4.
5.

Mengesahkan :

Wakil Ketua I,

Dr. Darlan Sidik, M.Pd.

Ketua Jurusan Teknik Informatika

Ir. Billy Eden William Asrul, S.Kom., MT.

Mengetahui :
Ketua STMIK Handayani Makassar

Dr. Nasrullah, M.Si.

PENGENALAN POLA CITRA DENGAN EKSTRAKSI *GRAY LEVEL CO-OCCURRENCE MATRIX* DAN KLASIFIKASI *K-MEANS* UNTUK MENDETEKSI PENYAKIT PADA DAUN CENGKEH

Hasbullah ,(2020)

Dibimbing oleh : Mirfan dan Nurilmiyanti Wardhani

ABSTRAK

Para petani/pengurus tanaman cengkeh yang melakukan diagnosa terhadap gejala-gejala yang muncul pada tanaman mereka hanya dengan bermodalkan pengalaman atau bahkan mereka harus mencari informasi dari para petani/pengurus tanaman cengkeh lainnya yang lebih berpengalaman atau bahkan menghubungi pakar yang ahli. Tujuan dari penelitian ini adalah untuk merancang dan membangun pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh dan mengimplementasikan pengenalan pola citra dengan metode *GLCM* dalam mendapatkan nilai ekstraksi sebuah gambar dan mengklasifikasikan penyakit atau hama tanaman cengkeh dengan metode *K-Means*.

Penelitian ini menggunakan dua metode dalam mempermudah para petani/pengurus cengkeh dalam mendiagnosa penyakit tanamannya yang pertama olah citra *Gray Level Co-Occurrence Matrix* dan klasifikasi penyakit dengan algoritma *K-Means*. Pengembangan sistem yang digunakan yaitu UML (*Unified Modeling Language*). Sedangkan pembuatan perangkat lunak dalam skripsi ini, pembuatan system ini menggunakan perangkat lunak *PHP, HTML, CSS, Java script* dan untuk *database* menggunakan *MySql*.

Hasil dari penelitian ini berupa sistem informasi yang memberikan hasil berupa klasifikasi jenis penyakit dengan menggunakan algoritma *k-means* berdasarkan hasil olah citra *Gray Level Co-Occurrence Matrix* sebagai ekstraksi citra gambar dari gambar daun tanaman cengkeh yang terdampak penyakit dan juga sistem ini dapat digunakan untuk membantu para petani/pengurus tanaman cengkeh dalam mendiagnosa penyakit apa yang menjangkit tanamannya hanya dengan mengunggah foto dari daun tanaman cengkeh yang terdampak.

Kata Kunci: *K-Means, GLCM, Olah Citra, Tanaman Cengkeh, Diagnosa*

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatu.

Dengan menyebut nama Allah SWT yang Maha Pengasih lagi Maha Penyayang, Kami panjatkan puja dan puji syukur atas kehadiran-Nya, yang telah melimpahkan rahmat, hidayah, dan inayah-Nya kepada kami, sehingga penulis dapat menyelesaikan penyusunan skripsi ini. Shalawat beserta salam semoga senantiasa terlimpah curahkan kepada Nabi Muhammad Shallallahu „alaihi wasallam.

Skripsi ini diajukan untuk memenuhi salah satu syarat memperoleh gelar sarjana (S1) pada STMIK Handayani Makassar dan juga merupakan laporan akhir yang wajib diselesaikan oleh setiap mahasiswa STMIK Handayani Makassar. Semoga dapat memberikan nilai tambah bagi semua pihak yang terkait didalamnya dan khususnya pada penulis.

Dalam penyusunan dan penulisan skripsi ini tidak lepas dari bantuan berupa sumbangan pemikiran, bimbingan serta dukungan dari berbagai pihak. Oleh karena itu dalam kesempatan ini penulis mengucapkan terima kasih yang setulus-tulusnya kepada yang terhormat :

1. Allah *subhana wata'ala* yang telah melancarkan segala urusan kami sehingga studi ini bisa kami selesaikan dengan baik dan lancar.

2. Kepada Orang Tua, dan adik – adik saya yang tercinta yang telah memberikan do“a dan restu serta dukungan kepada penulis untuk selalu maju dalam meraih cita – cita.
3. Bapak Dr. Nasrullah, M.Si selaku ketua STMIK Handayani Makassar.
4. Bapak Ir. Billy Eden William Asrul, S.Kom., MT Selaku Ketua Jurusan Teknik Informatika.
5. Ir. Mirfan, S.Kom., MT., M.Kom. sebagai Pembimbing I yang senantiasa memberikan masukan dan bimbingan bagi penulis.
6. Nurilmiyanti Wardhani, S.Kom., MT. sebagai Pembimbing II yang senantiasa memberikan masukan dan bimbingan bagi penulis.
7. Sahabat – Sahabat Saya yang senantiasa selalu mendukung saya. Terima Kasih atas kebersamaannya dalam merasakan susah senang dalam penyusunan skripsi ini, serta yang selalu memberikan dukungan, semangat, solusi dan keceriaannya sehingga memberikan pengaruh positif terhadap penulis.

Akhirnya, penulis menyadari bahwa penulisan skripsi ini masih banyak kekurangan. Oleh karena itu, kritik dan saran yang sifatnya mendidik, dan membangun, senantiasa penulis terima dengan lapang dada.

Akhir kata semoga skripsi ini dapat bermanfaat bagi pembaca, maupun kepada masyarakat umum. Semoga Allah *Subhanahu Wata'ala* memberikan rahmat dan karunia-Nya bagi kita semua, Aamiin.

Wassalamu Alaikum Warahmatullahi Wabarakatu.

Makassar, 07 April 2021

HASBULLAH

DAFTAR ISI

HALAMAN JUDUL.....	ixi
HALAMAN PERSETUJUAN.....	ixii
HALAMAN PENGESAHAN.....	ixv
ABSTRAK.....	v
KATA PENGANTAR	vi
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xv
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Rumusan Masalah	4
C. Batasan Masalah.....	4
D. Tujuan Penelitian	5
E. Manfaat Penelitian.....	6
BAB II LANDASAN TEORI	7
A. Tanaman Cengkeh.....	7
B. Penyakit.....	7
C. Pengenalan Pola	11
D. Pengolahan Citra Digital	11
E. <i>Gray-level co-occurrence matrix (GCLM)</i>	12
F. Data Mining	15
G. Klasterisasi	16
H. Algoritma <i>K-Means</i>	16

I. XAMPP	19
J. PHP	20
K. MySQL	21
L. <i>Unified Modeling Language</i>	23
1. <i>Use Case Diagram</i>	23
2. <i>Class Diagram</i>	25
3. <i>Sequence Diagram</i>	27
4. <i>State Diagram</i>	28
5. <i>Activity Diagram</i>	29
6. <i>Deployment Diagram</i>	31
M. Pengujian <i>Black Box</i>	32
N. Penelitian Terkait	33
O. Kerangka Pikir	34
BAB III METODE PENELITIAN.....	34
A. Penggambaran sistem.....	36
1. Sistem yang sedang berjalan	36
B. Analisis Kebutuhan Sistem	37
1. Kebutuhan Functional	37
2. Kebutuhan non Functional	38
C. Teknik Pengumpulan Data.....	39
1. Teknik Observasi	37
2. Studi Literatur	37
3. Wawancara.....	37
D. Metode Pengembangan Sistem	40
1. Analisis Kebutuhan.....	40
2. Desain Sistem	40
3. Penulisan Kode Program	40

BAB IV METODE PENELITIAN	42
A. Pemodelan sistem.....	42
1. <i>Use Case Diagram</i>	43
2. <i>Activity Diagram</i>	45
3. <i>Squence Diagram</i>	46
4. <i>Class Diagram</i>	48
B. Perancangan <i>DataBase</i>	48
C. Perancangan Antar Muka	51
1. Perancangan <i>Output</i>	51
2. Perancangan <i>Input</i>	54
D. Hasil Penelitian	56
1. Implementasi Algoritma	56
2. Tampilan Aplikasi.....	103
E. Pengujian Sistem	110
1. Pengujian blackbox	110
<u>BAB V PENUTUP</u>	116
A. Kesimpulan	116
B. Saran.....	117

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2.1 Bercak daun <i>pestalotia</i>	8
Gambar 2.2 Bercak <i>sooty mold</i>	8
Gambar 2.3 Bercak <i>antraknose</i>	9
Gambar 2.4 Bercak <i>coniella</i>	10
Gambar 2.5 Bercak <i>Cephaleuros</i>	10
Gambar 2.6 Sudut yang dibentuk	13
Gambar 2.7 Contoh olah citra	13
Gambar 2.8 Bagian-Bagian <i>Class Diagram</i>	25
Gambar 2.9 Kerangka Pikir	35
Gambar 3.1 use case diagram sistem yang sedang berjalan	36
Gambar 4.1 use case sistem diagram sisten yang diusulkan	43
Gambar 4.2 activity diagram admin	45
Gambar 4.3 activity diagram petani	46
Gambar 4.4 sequence diagram	47
Gambar 4.5 Class diagram	48
Gambar 4.6 rancangan relasi tabel	50
Gambar 4.7 Rancangan <i>Output</i> Halaman Utama	51
Gambar 4.8 Rancangan <i>Output</i> data penyakit	51
Gambar 4.9 Rancangan <i>Output</i> data sampel penyakit	52
Gambar 4.10 Rancangan <i>Output</i> data latihan	52

Gambar 4.11 Rancangan <i>Output</i> klasifikasi sampel	53
Gambar 4.12 Rancangan <i>input</i> login.....	53
Gambar 4.13 Rancangan <i>Input</i> data penyakit	54
Gambar 4.14 Rancangan <i>Input</i> ubah data penyakit	54
Gambar 4.15 Rancangan <i>Input</i> tambah sampel latihan penyakit.....	55
Gambar 4.16 Rancangan <i>Input</i> ubah sampel data latihan.....	55
Gambar 4.17 Rancangan <i>Input</i> klasifikasi sampel penyakit	56
Gambar 4.18 gambar citra asli	57
Gambar 4.19 gambar citra grayscale.....	58
Gambar 4.20 Hasil proses perubahan citra <i>RGB</i> ke <i>Gray scale</i>	59
Gambar 4.21. Sampel perhitungan.....	60
Gambar 4.22. hasil olah fitur <i>GLCM</i>	82
Gambar 4.23 sampel data latihan.....	83
Gambar 4.24 data uji	86
Gambar 4.25 Hasil klasifikasi sampel uji	92
Gambar 4.26 Pengambilan gambar sampel yang di anjurkan.....	103
Gambar 4.27 tampilan <i>form login</i>	103
Gambar 4.28 tampilan halaman utama.....	104
Gambar 4.29 tampilan menu data penyakit.....	104
Gambar 4.30 tampilan menu tambah data penyakit.....	105
Gambar 4.31 tampilan menu ubah data penyakit.....	105
Gambar 4.32 tampilan menu data sampel penyakit	106

Gambar 4.33 tampilan tambah data sampel latih	106
Gambar 4.34 tampilan proses olah citra sampel latih baru	107
Gambar 4.35 tampilan menu klasifikasi.....	108
Gambar 4.36 tampilan proses olah citra sampel uji baru	108
Gambar 4.37 tampilan detail proses olah citra.....	109
Gambar 4.38 tampilan <i>report</i> hasil proses klasifikasi.....	110

DAFTAR TABEL

Tabel 1.1 Data produksi	2
Tabel 2.1 Simbol – Simbol <i>Use Case Diagram</i>	24
Tabel 2.2 Simbol – Simbol <i>Class Diagram</i>	26
Tabel 2.3 Simbol – Simbol <i>Sequence Diagram</i>	27
Tabel 2.4 Simbol – Simbol <i>State Diagram</i>	29
Tabel 2.5 Simbol – Simbol <i>Activity Diagram</i>	30
Tabel 2.6 Simbol – Simbol <i>Deployment Diagram</i>	31
Tabel 2.7 Penelitian terkait	33
Tabel 4.1 users	48
Tabel 4.2 penyakit.....	49
Tabel 4.3 data_awal	49
Tabel 4.4 center	49
Tabel 4.5 center_lama	50
Tabel 4.6 klas	50
Tabel 4.7. tabel matriks perhitungan.....	60
Tabel 4.8 matriks 0 derajat.....	61
Tabel 4.9 matriks transpose 0 derajat.....	62
Tabel 4.10 nilai instensitas 0 derajat	62
Tabel 4.11 matriks ko-okurasi 0 derajat.....	63
Tabel 4.12 matriks 45 derajat.....	65

Tabel 4.13 matriks <i>transpose</i> 45 derajat	65
Tabel 4.14 nilai instensitas 45 derajat	66
Tabel 4.15 matriks ko-okurasi 45 derajat.....	67
Tabel 4.16 matriks ko okurasi 90 derajat	68
Tabel 4.17 matriks transpose 90 derajat.....	69
Tabel 4.18 nilai instensitas 90 derajat	69
Tabel 4.19 matriks ko-okurasi 90 derajat.....	70
Tabel 4.20 matriks 135 derajat.....	72
Tabel 4.21 matriks transpose 135 derajat.....	72
Tabel 4.22 nilai instensitas 135 derajat	73
Tabel 4.23 matriks ko-okurasi 135 derajat.....	74
Tabel 4.24 hasil fitur energi	76
Tabel 4.25 fhasil fitur entropi.....	78
Tabel 4.26 hasil fitur homogeniti	79
Tabel 4.27 hasil fitur kontras	80
Tabel 4.28 rata-rata ciri fitur ke-empat derajat	81
Tabel 4.29 data sampel latih.....	83
Tabel 4.30 penentu centeroid ke-1	84
Tabel 4.31 penentu centeroid ke-2	84
Tabel 4.32 penentu centeroid ke-3	84
Tabel 4.33 Tabel klaster penyakit	85

Tabel 4.34 Data penyakit dengan nilai <i>GLCM</i>	87
Tabel 4.35 Nilai kedekatan iterasi pertama.....	88
Tabel 4.36 Tabel centroid baru	89
Tabel 4.37 Tabel nilai kedekatan ke dua.....	89
Tabel 4.38 Tabel nilai centroid ke dua.....	90
Tabel 4.39. Tabel iterasi ke tiga.....	90
Tabel 4.40. Tabel kluster ke tiga	91
Tabel 4.41. Tabel iterasi ke empat	91
Tabel 4.42 Tabel pengujian gambar sampel	96
Tabel 4.43 Rencana pengujian	111
Tabel 4.44 Hasil Pengujian Login.....	112
Tabel 4.45 Hasil Pengujian menu data penyakit.....	113
Tabel 4.46 Hasil Pengujian menu data sampel latih	114
Tabel 4.47 Hasil Pengujian proses olah citra <i>GLCM</i>	115
Tabel 4.48 Hasil Pengujian proses klasifikasi <i>K-Means</i>	115

BAB I

PENDAHULUAN

A. Latar Belakang

Pertanian merupakan sumber perekonomian nasional. Banyak orang Indonesia yang bekerja sebagai petani. Mampu meningkatkan kesejahteraan petani berarti meningkatkan produktivitas lahan garapan pertanian. Produktivitas pertanian pada dasarnya adalah suatu cara untuk meningkatkan produksi dengan kualitas dan kuantitas yang maksimal dengan menggunakan pupuk dan obat-obatan yang minimal serta dengan penggunaan sumber daya alam yang terbatas (air, energi, kondisi tanah).

Cengkeh merupakan salah satu produk pertanian yang bernilai ekonomis tinggi. Cengkeh dapat tumbuh di tanah yang gembur dan berdrainase baik. Tanaman ini merupakan tanaman tropis yang sangat cocok tumbuh di Indonesia, Cengkeh banyak digunakan dalam industri sebagai bahan baku produksi tembakau kretek.. Salah satu wilayah penghasil cengkeh di Indonesia berada di kabupaten Kepulauan Selayar. Adapun perkembangan produksi cengkeh dalam empat tahun terakhir adalah:

Tabel 1.1. Data produksi cengkeh

No	Tahun	Hasil (TON)
1	2016	178,35
2	2017	102,49
3	2018	535,09
4	2019	482,07

(sumber: Dinas pertanian dan ketahanan pangan Kepulauan Selayar)

Pada tahun 2016 kabupaten Kepulauan selayar menghasilkan cengkeh dengan jumlah 178,35 ton, namun terdapat penurunan produksi cengkeh di tahun 2017 dengan jumlah cengkeh yang dihasilkan kisaran 102,49 ton, kemudian di tahun 2018 produksi cengkeh kembali naik lagi dengan jumlah produksi 535,09 ton, namun produksi cengkeh pun kembali menurun di tahun 2019 dengan hasil cengkeh 482,07 ton. Tanaman cengkeh memiliki masa tanam setahun sehingga panen yang dapat dilakukan setahun sekali saja. Faktor-faktor yang mempengaruhi pertumbuhan dan produktivitas cengkeh harus diperhatikan. Salah satu faktor yang dapat mempengaruhi pertumbuhan dan produktivitas cengkeh adalah adanya penyakit tanaman cengkeh.

Untuk mengatasi permasalahan yang terjadi, maka penulis perlu suatu sistem berbasis web dengan pengenalan citra untuk membantu memudahkan para petani/pengurus tanaman cengkeh mendapatkan informasi terhadap penanganan atau jenis penyakit apa yang terdapat pada tanaman cengkehnya, dengan menggunakan metode *K-Means clustering* sebagai cara untuk mengklasifikasi jenis penyakit apa yang terdapat pada tanaman cengkeh.

Sebelum petani mendapatkan hasil klasifikasi, para petani harus terlebih dahulu dapat mengetahui gejala-gejala apa saja yang di alami oleh tanaman cengkehnya, untuk mempermudah dalam pengamatan terhadap gejala-gejala tersebut, proses ini akan dilakukan dengan penggunaan sampel gejala yang terjadi kemudian akan di foto dan hasil foto tersebut akan di unggah pada sistem dan akan diekstrak citranya menggunakan metode *GLCM (Gray Level Cooccurrence Matrices)*.

Metode *GLCM* atau *Gray Level Cooccurrence Matrices* dilakukan untuk mendapatkan nilai citra dari gambar terhadap gejala-gejala yang dialami oleh daun pada tanamn cengkeh, *Metode Gray Level Co-occurrence Matrix* adalah salah satu ekstraksi order kedua pada fitur statistic tekstur.ekstraksi order kedua menunjukkan hubungan statistik antara 2 piksel. *GLCM* adalah sebuah matriks dengan jumlah baris dan kolom sebanding dengan jumlah gray level(G) dalam suatu citra. Metode *GLCM* menggunakan citra berskala keabuan (*greyscale*).

Algoritma *K-Means* bertujuan mengelompokkan data berdasarkan kemiripan karakteristik yang dimiliki oleh data tersebut. *K-Means* merupakan salah satu metode data klastering partitional (non hirarki) yang mempartisi data ke dalam klaster/kelompok sehingga data yang memiliki karakteristik yang sama dikelompokkan dalam satu klaster. Selain itu, *K-Means* juga termasuk sebagai teknik klasterisasi yang terawasi sehingga pada awal komputasi telah ditetapkan terlebih dahulu jumlah klaster yang diinginkan.

Dari pemaparan masalah tersebut, maka dirancang sebuah sistem untuk menyampaikan informasi yang dihasilkan dalam bentuk website , adapun judul penelitian yaitu “**Pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh**”.

B. Rumusan Masalah

Berdasarkan latar belakang masalah di atas, maka rumusan masalah yang mendasari penyusunan skripsi ini adalah sebagai berikut :

1. Bagaimana merancang pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh?
2. Bagaimana mengimplementasikan pengenalan pola citra dengan metode *GLCM* dalam mendapatkan nilai ekstraksi sebuah gambar dan mengklasifikasikan penyakit tanaman cengkeh dengan metode *K-Means*?

C. Batasan Masalah

Untuk menghindari meluasnya pembahasan mengenai pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh, maka penulis membatasi permasalahan dan hanya mencakup hal – hal di bawah ini.

1. Pengenalan pola citra menggunakan metode *GLCM* dengan ciri hanya empat ciri fitur yang digunakan yaitu *energi*, *entropi*, *homogeniti*, dan *kontras*.
2. Penerapan algoritma *K-Means* dilakukan terhadap hasil ciri fitur metode *GLCM*.
3. Penyakit yang diteliti hanya yang memiliki dampak pada daun saja yaitu, *pestalotia*, *sooty mold*, *antraknose*, *coniella*, dan *cephaleuros*.
4. Output yang dihasilkan berupa hasil klasifikasi data uji dengan nama penyakit, cara penanganan dan nilai uji metode *GLCM*.

D. Tujuan Penelitian

Adapun tujuan penelitian yaitu sebagai berikut:

1. Untuk merancang dan membangun pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh
2. Untuk mengimplementasikan pengenalan pola citra dengan metode *GLCM* dalam mendapatkan nilai ekstraksi sebuah gambar dan mengklasifikasikan penyakit tanaman cengkeh dengan metode *K-Means*

E. Manfaat Penelitian

1. Manfaat bagi tempat penelitian.
 - a. Memudahkan para petani/pengurus tanaman cengkeh dalam mendiagnosa penyakit apa yang terdapat pada tanaman cengkehnya dan penanganannya.
 - b. Membuat para petani/pengurus tanaman cengkeh meminimalisir keperluan tambahan diluar biaya penanganan penyakit atau ham itu sendiri.
2. Manfaat bagi dunia pendidikan
 - a. Menambah referensi teori kepada para penulis dimasa mendatang.
 - b. Menambah pengetahuan terhadap para pelajar/mahasiswa yang sedang mencari informasi terkait penelitian ini.
3. Manfaat bagi penulis
 - a. Sebagai sarana dalam mengaplikasikan ilmu yang telah diperoleh selama mengikuti proses belajar-mengajar dibangku perkuliahan.
 - b. Dapat melatih diri dalam menganalisa masalah, merancang, dan memahami perancangan aplikasi.
 - c. Dapat mengetahui, mengerti, dan merasakan praktik pelaksanaan komputerisasi dalam suatu rancangan aplikasi yang nyata.

BAB II

LANDASAN TEORI

A. Tanaman Cengkeh

Cengkeh tentunya tergolong tanaman industri atau perkebunan yang sangat baik di Indonesia. Cengkeh dapat tumbuh di tanah yang bersifat gembur dan berdrainase baik. Tanaman ini merupakan tanaman tropis yang sangat cocok tumbuh di Indonesia. Hal ini dikarenakan cengkeh dapat menghasilkan minyak atsiri yang dapat digunakan sebagai bahan baku dalam industri makanan dan farmasi. Yang menonjol di Indonesia tentu saja digunakan sebagai bahan baku produksi tembakau.(Setiawan et al., 2018).

B. Penyakit

Penyakit tanaman dapat disebut juga organisme pengganggu tanaman. Terdapat beberapa penyakit yang menyerang tanaman tembakau. Berbagai penyakit dapat dihindari dari berbagai gejala yang ditimbulkan (Ariyani, 2019)

Berdasarkan pedoman budi daya cengkeh di kebun campur yang dirilis oleh balai penelitian tanaman rempah dan obat tahun 2015 terdapat beberapa jenis penyakit yang dapat menjangkit tanaman cengkeh seperti yang tertera berikut (Wahyuno & Martini, 2015) :

1. Bercak daun *pestalotia* (*pestalotia versicolor*)



Gambar 2.1. Bercak daun *pestalotia*

a) Gejala:

- 1) Daun berbercak dengan warna hitam kecoklatan dan bercak meluas ke seluruh bagian daun.

b) Cara mengatasi:

- 1) Gemburkan tanah disekitar pohon dan beri pupuk organik.
- 2) Penyemprotan fungisida.

2. Bercak *sooty mold* (*capnodium sp.* Dan *limacinula samoensis*).



Gambar 2.2. Bercak *sooty mold*

a) Gejala:

- 1) Pada permukaan daun ada jelaga berwarna coklat, abu-abu kehitaman.

b) Pengendalian:

- 1) Menyemprot daun dengan insektisida.
- 2) Menggosok tangan ke daun yang terkena dan menyiramkan air ke daun.

3. Bercak *antraknose* (*colletotrichum gloeosporioides*)



Gambar 2.3. Bercak *antraknose*

a) Gejala:

- 1) Pada bagian ujung atau bagian lain dari daun terlihat berwarna coklar mengering.

b) Pengendalian

- 1) Memiliki drainase.
- 2) Menggemburkan tanah.
- 3) Melakukan pemupukan yang sempurna.
- 4) Melakukan penyiraman yang jenuh pada musim kemarau

4. Bercak *coniella* (*coniella castaneicola*)



Gambar 2.4. Bercak *coniella*

a) Gejala:

1) Pada daun terdapat bercak menguning membundar.

b) Cara mengatasi:

1) Penyemprotan pestisida sistematik berbahan aktif benomil, metil, trofanat, kependonzim, dan fungisida kontak.

5. Bercak *cephaleuros* (*cephaleuros virescens*)



Gambar. 2.5. Bercak *cephaleuros*

a) Gejala:

1) Bercak-bercak nekrosis kemerahan berbentuk, dengan ukuran sekitar 1-3mm dan tidak di batasi oleh tulang.

b) Cara mengatasi:

1) Menyemprotkan tanaman dengan ekstrak kompos yang mengandung *chitosan*.

2) Pemberian bahan organik (pupuk kandang atau pupuk kompos) dan pelumas.

C. Pengenalan Pola

Pengenalan Pola (Pattern Recognition) Pola adalah kumpulan dari ciri-ciri yang merupakan sifat dari sebuah objek. Dalam klasifikasi, pola berupa sepasang variable (x, ω) , dengan x adalah sekumpulan pengamatan atau ciri (vektor ciri) dan ω adalah konsep di balik pengamatan (label) (Waliyansyah et al., 2018).

D. Pengolahan Citra Digital

Citra digital adalah sebuah presentasi fungsi intensitas cahaya $f(x,y)$ dari setiap piksel pada sebuah citra dengan x dan y menunjukkan koordinat spasial dan nilai dari fungsi yang menunjukkan kecerahan citra pada setiap titik piksel (x,y) tersebut (Waliyansyah et al., 2018).

Pemrosesan gambar adalah studi tentang algoritma yang mengambil gambar sebagai input dan mengembalikannya sebagai output. Saat ini,

pemrosesan gambar merupakan alat penting dalam banyak bidang ilmiah seperti ilmu komputer, teknik listrik dan elektronik, robotika, fisika, kimia, ilmu lingkungan, biologi, dan psikologi ilmiah. Pemrosesan gambar adalah subkelas pemrosesan sinyal. Input untuk pemrosesan sinyal adalah gambar dan output juga merupakan gambar atau fitur/karakteristik yang terkait dengan gambar tersebut. Pengolahan citra saat ini berkembang pesat dan banyak kajian tentang pengolahan citra di bidang teknologi dan ilmu informasi (Johan Wahyudi & Ihdahubbi Maulida, 2019).

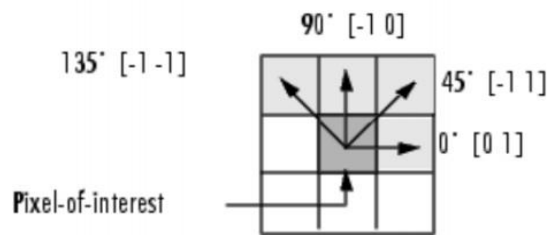
E. *Gray-Level Co-occurrence Matrix (GLCM)*

Metode *Gray-Level Co-occurrence Matrix* adalah salah satu ekstraksi order kedua pada fitur statistik tekstur. Ekstraksi order kedua menunjukkan hubungan statistik antara 2 piksel. *GLCM* adalah sebuah matriks dengan jumlah baris dan kolom sebanding dengan jumlah gray level (G) dalam suatu citra. Metode *GLCM* menggunakan citra berskala keabuan (*greyscale*) (Alvansga, 2019)

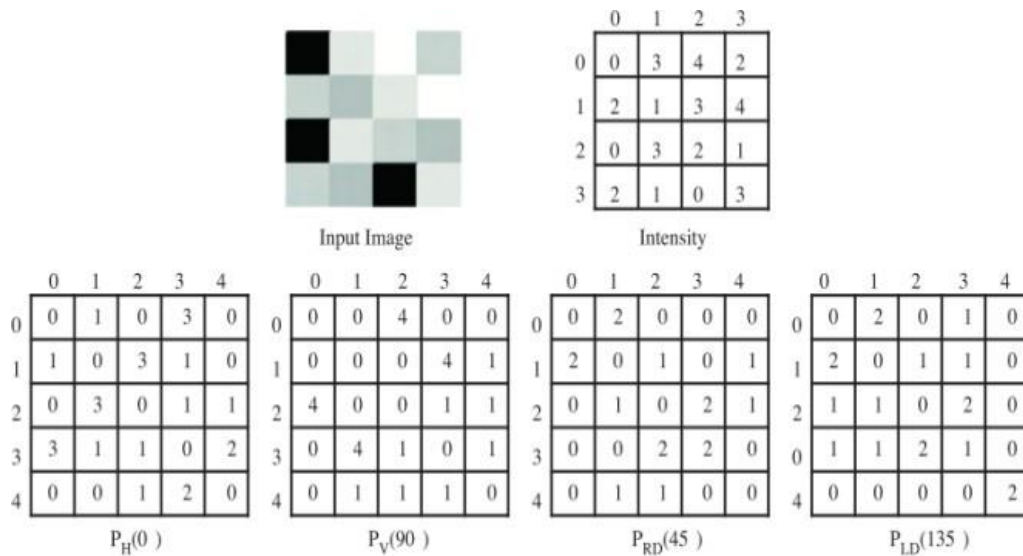
Cara menemukan tingkat keabuan reguler dengan menganalisis piksel dalam gambar. Metode ini juga digunakan untuk mentabulasi kombinasi frekuensi nilai piksel yang ditampilkan dalam sebuah gambar.

Jika Anda ingin melakukan analisis gambar berdasarkan distribusi statistik intensitas piksel, Anda dapat melakukannya dengan mengekstrak fitur tekstur. *GLCM* adalah metode ekstraksi fitur berbasis statistik yang mengumpulkan fitur dari nilai array piksel yang memiliki nilai tertentu dan

membentuk sudut sampel. Sudut yang terbentuk dari nilai piksel suatu citra dengan menggunakan *GLCM* adalah 0 derajat, 45 derajat, 90 derajat, dan 135 derajat, dan sudut yang terbentuk dapat dilihat pada gambar berikut (Rahman dan Wijayanto, 2015).



Gambar 2.6. Sudut yang dibentuk.



Gambar 2.7. Contoh olah citra.

Grayscale co-occurrence matrix (GLCM) pertama kali diusulkan oleh Hararikku pada tahun 1979 dengan 28 fungsi untuk membangun model spasial. Langkah pertama dalam menghitung karakteristik *GLCM* adalah mengubah citra *RGB* menjadi citra grayscale. Langkah kedua adalah membuat

matriks *co-occurrence* dan selanjutnya menentukan hubungan spasial antara piksel referensi dan piksel yang berdekatan sebagai fungsi sudut dan jarak d . Langkah selanjutnya adalah membuat suatu matriks simetris dengan cara menambahkan matriks kovariat ke matriks transpos. kemudian berikut ini dilakukan dengan menghitung probabilitas setiap elemen matriks normalisasi dari matriks simetris. Kemudian pada langkah terakhir adalah menghitung suatu fungsi dari algoritma *GLCM*. Setiap objek dihitung pada jarak 1 piksel dalam empat arah (0° , 45° , 90° , 135°) untuk deteksi *co-occurrence* (Rahman & Wijayanto, 2015).

Matriks *co-occurrence* memiliki sebelas fitur tekstur yang dapat diperoleh dari citra yang digunakan untuk membedakan citra satu layer dengan citra lainnya, lima di antaranya adalah tekstur realistik tipe fitur dan enam lainnya. Dari kelima fungsi tersebut. Fitur-fitur tersebut adalah sebagai berikut (Ganis et al., 2011)..

1. Energi (*Eenergy*)

$$= \sum \sum ()$$

2. Entropi (*Entropy*)

$$= \sum \sum () ()$$

3. Kontras (*Contrast*)

$$= \sum \sum () ()$$

4. Homogenitas (*Homogeneity*)

$$= \sum \sum \frac{(\quad)}{\quad}$$

5. Korelasi (*Correlation*)

$$= \sum \sum \frac{(\quad)(\quad)}{\quad}$$

Dengan

$$\sum \sum (\quad)$$

$$\sum \sum (\quad)$$

$$\sum \sum (\quad)(\quad)$$

$$\sum \sum (\quad)(\quad)$$

Pada penelitian ini ciri tekstur yang akan digunakan hanya empat dari lima ciri tekstur dasar yaitu , energi, entropi, kontras dan homogenitas.

F. Data Mining

Data mining adalah metode pemrosesan data untuk menemukan pola tersembunyi dalam data. Hasil pengolahan data dengan menggunakan teknologi data mining ini dapat digunakan untuk pengambilan keputusan di masa yang akan datang. Data mining disebut juga dengan pengenalan pola (Santosa, 2007). Karena data mining adalah metode pengolahan data skala besar, data mining memainkan peran penting dalam industri, keuangan, meteorologi, ilmu pengetahuan dan teknologi. Secara umum, penelitian data mining menggambarkan metode seperti pengelompokan, klasifikasi, regresi,

pemilihan variabel, dan analisis keranjang pasar (Johan Wahyudi & Ihdahubbi Maulida, 2019).

G. Klasterisasi

Pada dasarnya, *clustering* adalah suatu metode untuk menemukan dan mengelompokkan data yang memiliki karakteristik yang mirip (*similarity*) antara data yang terkumpul dengan data lainnya. Clustering adalah metode penambangan data tanpa pengawasan. Dengan kata lain, metode yang diterapkan tanpa pendidikan (pelatihan) dan tanpa guru (teacher) dan tidak memerlukan sasaran hasil. Data mining memiliki dua jenis metode clustering yang digunakan untuk mengelompokkan data yaitu, hierarchical clustering dan non-hierarchical clustering (Johan Wahyudi & Ihdahubbi Maulida, 2019)..

H. Algoritma K-Means

Pada penelitian ini digunakan teknik *clustering* menggunakan algoritma *K-Means* untuk mengelompokkan data berdasarkan kesamaan karakteristik data. *K-Means* merupakan salah satu metode clustering data (*non-hierarchical*) yang membagi data menjadi cluster/grup sehingga data dengan karakteristik seperti *K-Means* dikelompokkan menjadi satu cluster. Selain itu, karena guru juga memasukkan metode *K-Means* sebagai teknik *clustering*, maka jumlah cluster yang dibutuhkan ditentukan di awal perhitungan (Iswari dan Ayu 2015).

Algoritma *K-Means* secara umum memiliki tahapan sebagai berikut (Iswari & Ayu, 2015)

1. Pada awal iterasi, ada pemilihan pusat kluster secara bebas. Dalam hal ini dapat dipilih secara random sejumlah anggota dari dataset sesuai dengan jumlah kluster (k) yang telah ditetapkan.
2. Menghitung jarak setiap data dengan setiap pusat kluster. Untuk melakukan penghitungan jarak dapat digunakan formula Euclidean (Lizda Iswari1 2015) seperti tampak pada rumus :

$$\sqrt{\sum (\quad)}$$

Dengan:

d_{ik} = jarak antara data ke- i dengan titik pusat kluster ke- k

m = jumlah atribut

x_j = data ke- i

c_k = data pusat kluster ke- k .

3. Jika jarak antara semua cluster yang ada dengan jarak paling minimal, diantara semua data anggota cluster.
4. Mengelompokkan data anggota dari setiap cluster.
5. Memperbaharui nilai *centroid cluster* dapat dihitung dengan mencari nilai rata-rata jumlah anggota di setiap cluster menggunakan rumus berikut:

$$\sqrt{\frac{\sum}{p}}$$

Dengan:

C_{kj} = Pusat Cluster

p = Jumlah semua anggota cluster

h = Jumlah awal anggota cluster

Y_{hj} = Jumlah data

6. Melakukan perulangan langkah 2 sampai 5 sampai tidak ada data yang ditransfer ke *cluster* lain..

Algoritma *K-Means* sendiri tentunya memiliki keunggulan dan juga kelemahan dalam proses nya, sebelum mengetahui apa saja keunggulan dari algoritma ini perlu diketahui terlebih dahulu bagaimana karakteristik algoritma *K-Means* itu sendiri seperti :

1. *K-Means* sangat cepat selama pengelompokan.
2. *K-Means* sangat sensitif terhadap pembuatan centroid awal.
3. Menengah cluster tidak memiliki anggota.
4. Hasil clustering *K-Means* tidak unik (variable) selalu berubah-ubah.
5. *K-Means* sangat sulit untuk mencapai optimasi secara keseluruhan.

Berdasarkan karakteristik yang telah disebutkan di atas, di ketahui keunggulan algoritma *K-Means* khususnya terhadap penelitian ini antara lain:

1. Proses pengelompokkan data yang cepat dan efisien
2. Penentuan *centroid* (pusat kluster) awal yang tepat untuk pemilihan kluster tingkat kerawanan dapat membuat kluster yang baik.
3. Untuk kategori selain tingkat kerawanan, setiap data dapat menjadi anggota kelompok yang berbeda berdasarkan jumlah kluster yang ingin di buat, menjadikan tingkat pemilihan anggota untuk tiap kluster lebih baik.
4. Penerapan algoritma *K-Means* yang terbilang mudah untuk di pelajari.

I. XAMPP

XAMPP dapat diinstal pada sistem operasi *Windows, Linux, Solaris* atau *Mac OS*, dan paket server web yang lengkap memudahkan pengembang untuk membangun server web lintas platform dan berlisensi. Paket *XAMPP* mencakup *Apache* sebagai *server web*, *MySQL* sebagai sistem database, *PHP* sebagai skrip sisi *server*, dan banyak perpustakaan lainnya. *XAMPP* dikembangkan dari *Apache Friends*.

XAMPP Web Server (X (/Window/Linux) *Apache MySQL PHP* dan *Perl*) Paket *database PHP* dan *MySQL* paling populer di kalangan pengembang web yang menggunakan *PHP* dan *MySQL* sebagai *database*..

XAMPP merupakan program yang digunakan oleh *server* untuk menjalankan fungsi-fungsi yang ada pada halaman web yang kita buat sekaligus merender halaman web tersebut untuk dikunjungi pengguna (Ardianto & Nst, 2016).

J. PHP

PHP adalah singkatan dari *PHP Hypertext Preprocessor*. Bahasa *scripting* yang diimplementasikan dan diproses oleh *server*. Hasilnya dikirim ke klien dan pengguna menggunakan browser (Firman et al., 2016)

PHP adalah singkatan dari *PHP Hypertext Preprocessor*. Bahasa *scripting* digunakan dan diproses di *server*. Hasil dikirim ke klien dan pengguna menggunakan browser (Firman et al., 2016).

Singkatan *PHP* atau *Hypertext Preprocessor* merupakan salah satu bahasa pemrograman *open source* yang dapat diintegrasikan ke dalam makalah *HTML* yang sangat cocok atau cocok untuk pengembangan web. Anda dapat menulis bahasa pemrograman seperti *C*, *Java*, *Perl*, dll yang mudah untuk diucapkan dan dipelajari dalam bahasa *PHP*.

PHP adalah bahasa *scripting server side* di mana pemrosesan data dilakukan di sisi *server*. Sederhananya, itu menerjemahkan skrip program *server* dan hasilnya dikirim ke klien yang membuat permintaan. Cara lain untuk memahami *PHP* adalah singkatan dari *Hypertext Preprocessor*. Ini adalah bahasa pemrograman berbasis kode (*script*) yang digunakan untuk memproses data dan mengembalikannya ke *browser web* sebagai kode *HTML*.

Sebuah prinsip sistem kerja pada *PHP* dimulai dengan permintaan ke halaman web melalui *browser*. berdasarkan *URL* atau alamat situs web Di Internet, *browser* menemukan alamat di *server web*, menemukan halaman yang diinginkan, dan mengirimkan semua informasi yang diperlukan ke *server*

web. *Server web* kemudian menemukan file yang diminta dan menampilkan isinya di *browser*. *Browser* yang mendapatkan konten atau sebuah perintah segera mengonversi kode *HTML* dan menampilkannya pada sebuah aplikasi. Bagaimana jika halaman yang berisi skrip *PHP* ini dipanggil oleh pengguna. Prinsipnya sama dengan pemanggilan kode *HTML*, namun ketika sebuah *request* dikirimkan ke *web server*, *web server* mengecek jenis file yang diminta oleh user. Jika jenis file yang diminta adalah *PHP*, periksa konten skrip di halaman *PHP* dan kemuaan akan menampilkannya.

Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode *PHP* untuk mengirimkan permintaan ke *server* (Firman et al., 2016)

K. My SQL

Sebuah program untuk membuat dan memelihara *database My Structured Query Language (MySQL)*, umumnya dikenal sebagai sistem manajemen *database (DBMS)*. Sifat *DBMS* ini adalah *open source*. *MySQL* dapat digunakan untuk aplikasi multi user (*multi user*) karena juga memiliki program akses *database* jaringan. Keuntungan lain dari *MySQL* adalah menggunakan bahasa *query SQL standar*

SQL adalah suatu bahasa permintaan yang terstruktur, *SQL* telah distandarkan untuk semua program pengakses *database* (Firman et al., 2016).

MySQL adalah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS* multithreaded dan multi-pengguna. *MySQL* dibangun, disebar, dan didukung dari *MySQL AB*. *MySQL AB* adalah perusahaan perdagangan yang disponsori oleh pengembang *MySQL*. *MySQL* dapat didefinisikan dalam sistem manajemen basis datanya sendiri, yang merupakan struktur untuk menyimpan data..

Untuk mengakses, mengolah, dan mengolah data yang tersimpan dalam basis data komputer, diperlukan sistem manajemen basis data seperti *server MySQL*. atau, *MySQL* dapat dianggap sebagai *Database Terhubung (RDBMS)*. Database tertaut menyimpan data dalam tabel terpisah.

Server database MySQL dapat diakses, mudah digunakan dan dapat diandalkan. Sejak *MySQL* telah berhasil digunakan selama bertahun-tahun, dikembangkan untuk mempercepat database besar, *MySQL Server* sangat cocok untuk mengakses database dari *Internet*. *MySQL* juga merupakan sistem *client-server* yang terdiri dari *server SQL* multithreaded yang mendukung berbagai perangkat lunak dan pustaka klien.

SQL (Structured Query Language) adalah sebuah bahasa standar yang digunakan untuk mengakses *database server*. Bahasa ini awalnya dikembangkan oleh *IBM*, tetapi sejak itu diterima dan digunakan sebagai standar *industri*. *SQL* membuat akses *database* lebih *user-friendly*.

Dalam konteks sebuah *SQL*, informasi biasanya disimpan dalam tabel dua arah logis yang terdiri dari dua baris data (baris dan catatan) dalam satu atau lebih kolom. Baris tabel sering disebut sebagai contoh data, dan kolom yang sering disebut sebagai properti atau bidang. Seluruh tabel dikumpulkan menjadi satu kesatuan yang disebut *database*. (Firman et al., 2016)..

L. *Unified Modeling Language (UML)*

Unified Modeling Language, kependekan dari *UML*, adalah bahasa pemodelan untuk membuat perangkat lunak berdasarkan konsep pemrograman berorientasi objek, atau umumnya *Object Oriented Programming (OOP)*. *Unified Modeling Language* merupakan bahasa visual untuk pemodelan dan juga berfungsi sebagai sarana komunikasi dengan sistem yang menggunakan teks dan gambar pendukung. *UML* dibagi menjadi beberapa tampilan, yang merupakan bagian sederhana dari struktur pemodelan *UML* yang mewakili aspek sistem (Bin Ladjamudin, 2013)

Dalam *UML*, terdapat beberapa diagram antara lain sebagai berikut.

1. *Use Case Diagram*




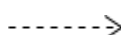
Diagram use case atau *use case diagram* merupakan pemodelan untuk menggambarkan kelakuan sistem yang akan dibuat. Diagram ini mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan

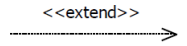

Sistem yang direncanakan atau dibuat. *Use case diagram* digunakan untuk menentukan fitur-fitur yang terdapat pada sistem Anda dan siapa yang berwenang untuk menggunakan fitur-fitur tersebut..

Ada beberapa notasi untuk deskripsi use case diagram, seperti use case dan hubungan dengan aktor. Untuk kasus penggunaan yang menggunakan kata kerja, aktor terlibat langsung dengan sistem dan aktor melakukan tugas yang terkait dengan kasus penggunaan.

Berikut ini adalah simbol-simbol yang terdapat pada *use case*.

Tabel 2.1. Simbol – Simbol *Use Case Diagram*

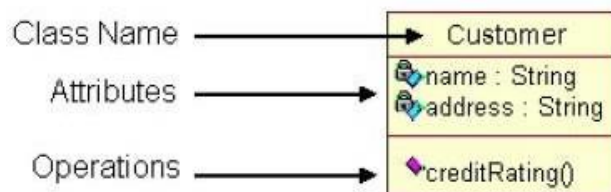
No.	Gambar	Nama	Keterangan
1.		<i>Aktor</i>	Orang lain, proses, atau sistem yang berinteraksi dengan sistem yang dibuat pengguna. Aktor bertanggung jawab untuk berinteraksi dengan <i>use case</i> .
2.		<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> berpartisipasi dalam <i>use case</i> atau <i>use case</i> yang berinteraksi dengan agen.
3.		<i>Generalization</i>	Hubungan antara generalisasi dan spesialisasi (umum-spesifik) antara dua kasus penggunaan di mana satu fitur lebih umum daripada yang lain.
4.		<i>Include</i>	Kasus penggunaan tambahan berisi kasus penggunaan yang memerlukan kasus penggunaan sebagai kondisi, atau kasus penggunaan tambahan untuk

			menjalankan fungsinya. dijalankan <i>use case</i> ini.
5.		<i>Extend</i>	<i>Use Case</i> yang ditambahkan bisa independen tanpa <i>use case</i> ditambahkan ke Hubungan <i>Use Case</i> Tambahan dari <i>Use Case</i> .
6.		<i>Use case</i>	Menjelaskan serangkaian tugas yang dilakukan pada sistem yang menghasilkan hasil yang terukur untuk agen.

Sumber simbol: (Bin Ladjamudin, 2013)

2. *Class Diagram*

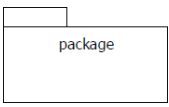
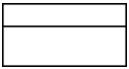



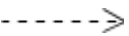

Diagram kelas atau diagram kelas menggambarkan struktur sistem yang mendefinisikan kelas-kelas yang dibuat untuk membangun sistem. Kelas memiliki properti dan metode, atau tugas.



Gambar 2.8. Bagian-Bagian *Class Diagram*

Berikut ini adalah simbol-simbol yang ada pada *class diagram* :

Tabel 2.2. Simbol – Simbol *Class Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>Package</i>	<i>Package</i> adalah paket yang terdiri dari satu atau lebih kelas.
2.		<i>Class</i>	Lapisan yang berhubungan dengan struktur sistem.
3.		<i>Association</i>	Secara umum, hubungan antara kelas terkait sering dikaitkan dengan tingkat keragaman tertentu.
4.		<i>Directed Association</i>	Hubungan antar kelas berarti bahwa satu kelas digunakan oleh yang lain, association biasanya juga disertai dengan multiplicity.
5.		<i>Generalization</i>	Generalisasi Hubungan antar kelas memiliki arti spesialisasi (intrinsik umum).
6.		<i>Dependency</i>	Hubungan antar kelas berarti ada hubungan ketergantungan antar kelas.
7.		<i>Aggregation</i>	Hubungan antar kelas yang memiliki arti keseluruhan (<i>whole-part</i>).

Sumber simbol: (Bin Ladjamudin, 2013)


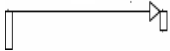
3. *Sequence Diagram*

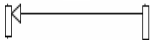
Sequence Diagram atau diagram urutan menggambarkan perilaku suatu objek dalam kasus penggunaan dengan menggambarkan masa hidup suatu objek dan pesan yang dikirim dan diterima antar objek. Jadi, untuk menggambar diagram urutan, Anda perlu mengetahui objek mana yang terdapat dalam use case Anda dan metode kelas yang dibuat dari objek tersebut.

Suatu hal yang penting bahwa semua use case yang mendefinisikan interaksi jalur pesan ditentukan, bahwa jumlah diagram urutan yang akan diplot sama dengan definisi use case yang memiliki prosesnya sendiri. Semakin spesifik kasus penggunaan Anda, semakin banyak diagram urutan yang harus Anda buat.

Berikut ini adalah simbol-simbol yang terdapat pada *sequence diagram*.

Tabel 2.3. Simbol – Simbol *Sequence Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>Lifeline</i>	Objek entitas, antarmuka interaktif.
2.		<i>Message</i>	Spesifikasi dan komunikasi antar objek berisi informasi tentang aktivitas yang terjadi.

3.		<i>Message</i>	Spesifikasi komunikasi objek ke objek berisi informasi tentang aktivitas yang sedang berlangsung.
----	---	----------------	---

Sumber simbol: (Bin Ladjamudin, 2013)

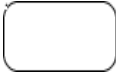



4. *State Diagram*

Diagram keadaan, grafik keadaan, atau grafik keadaan menunjukkan keadaan yang terjadi atau dapat terjadi pada suatu objek, seperti halnya setiap objek memiliki grafik keadaan.

State diagram menunjukkan semua *state/status*, dan Anda dapat memiliki objek kelas dan kejadian yang mengubah status objek tersebut. Perubahan keadaan disebut juga transisi. Transisi juga dapat memiliki tindakan yang dilakukan pada keadaan, dan lebih tepatnya tindakan yang dilakukan pada transisi keadaan. Gambar ini menunjukkan cara kerja sistem. *State* adalah suatu keadaan dimana suatu objek atau interaksi ada ketika suatu kondisi terpenuhi, suatu tindakan diambil, atau suatu peristiwa diharapkan.

Berikut ini adalah simbol-simbol yang terdapat dalam *state diagram*.

Tabel 2.4. Simbol – Simbol *State Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>State</i>	Nilai <i>link</i> dan nilai properti yang terkait dengan momen tertentu milik suatu objek.
2.		<i>Initial pseudo state</i>	Bagaimana suatu objek terbentuk atau dimulai.
3.		<i>Final state</i>	Bagaimana suatu objek terbentuk atau di akhiri.
4.		<i>Transition</i>	Peristiwa memicu objek status dengan memperbarui salah satu nilai properti..






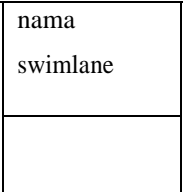
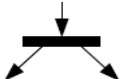
Sumber simbol: (Bin Ladjamudin, 2013)


5. *Activity Diagram*

Diagram aktivitas atau diagram aktivitas menggambarkan alur kerja atau aktivitas di sistem Anda. Satu hal yang perlu diingat di sini adalah bahwa grafik aktivitas mewakili aktivitas sistem, bukan apa yang dilakukan aktor.

Berikut ini adalah simbol-simbol yang terdapat pada *activity diagram*.

Tabel 2.5. Simbol – Simbol *Activity Diagram*

No.	Gambar	Nama	Keterangan
1.		Status Awal	Diagram aktivitas dalam keadaan awal operasi sistem awal.
2.		Aktivitas	Operasi yang dilakukan dalam sistem, operasi, biasanya dimulai dengan kata kerja.
3.		Percabangan / <i>decision</i>	Jika Anda memiliki beberapa koneksi titik atau opsi aktif.
4.		Penggabungan / <i>join</i>	Ini adalah integrasi gabungan yang menggabungkan beberapa tugas menjadi satu.
5.		Status akhir	Dalam diagram aktivitas di mana sistem dalam keadaan Jalankan Selesai, itu dalam keadaan Selesai
6.		<i>Swimlane</i>	Pisahkan organisasi profesi yang bertanggung jawab atas kegiatan yang terjadi
7.		<i>Fork</i>	Digunakan untuk menampilkan operasi yang berjalan secara paralel.

8.		<i>Join</i>	Ini digunakan untuk menandai kegiatan yang berpartisipasi.
----	---	-------------	--

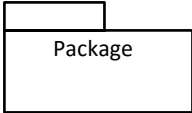
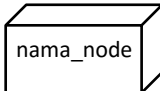
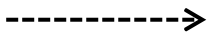
Sumber simbol: (Bin Ladjamudin, 2013)


6. *Deployment Diagram*

Diagram deployment menunjukkan komposisi komponen yang menjalankan aplikasi. Karena diagram tata letak mewakili tampilan pengembangan sistem, hanya ada satu diagram tata letak untuk sistem. Peta tata letak berisi node, yang merupakan perangkat keras fisik yang digunakan untuk menyebarkan aplikasi.

Adapun simbol-simbol yang terdapat pada *deployment diagram*.

Tabel 2.6. Simbol – Simbol *Deployment Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>Package</i>	Mewakili satu atau lebih grup tombol atau node
2.		<i>Node</i>	Umumnya digunakan untuk menyebut perangkat keras (hardware) dan perangkat lunak (software) yang tidak dibuat sendiri.
3.		Kebergantungan <i>/ dependency</i>	Dependensi antar node, arah panah yang menunjuk ke node digunakan.

4.		<i>Link</i>	Relasi antar <i>node</i> .
----	---	-------------	----------------------------

Sumber simbol: (Bin Ladjamudin, 2013)

M. Pengujian *Black Box*

Langkah-langkah pengujian penuh diambil untuk memastikan bahwa sistem atau aplikasi berfungsi dengan baik dan lebih mungkin untuk menemukan bug. Pengujian ini dilakukan dengan menggunakan metode pengujian *black box*.

Pengujian *black box* adalah teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari suatu aplikasi. (Jaya, 2018). Pengujian kotak hitam memungkinkan pengembang perangkat lunak untuk mendapatkan serangkaian kriteria entri yang sepenuhnya memenuhi semua persyaratan fungsional program.

Pengujian *black box* cenderung mencari kategori kesalahan fungsi dari sebuah sistem, sebagai berikut :

1. Kesalahan antarmuka (*interface errors*).
2. Kesalahan pada struktur data dan akses basis data.
3. Kesalahan performansi (*performance errors*).
4. Kesalahan inisialisasi dan terminasi.

N. Penelitian Terkait

Tabel 2.7. penelitian terkait

no	Nama	Judul	Hasil penelitian	Perbandingan dengan penelitian sekarang
1	Lilis indrayani .2020	Pengenalan Pola Citra dengan Klasifikasi <i>K-Means</i> untuk Mendeteksi Penyakit pada Daun Padi	Mendeteksi penyakit pada tanaman padi dari dampak atau gejala pada daunnya, dengan pola citra <i>glcm</i> yang citranya di segmentasi dan menggunakan 5 dari 6 ciri tekstur metode <i>glcm</i> , yang kemudian diklasifikasikan menggunakan algoritma <i>K-Means</i>	Penelitian sekarang menggunakan 4 dari 6 ciri tekstur metode <i>glcm</i> dan tanpa segmentasi terlebih dahulu demi menjaga detail pada gambar yang ada dan mengefisienkan proses komputasinya.
2	Yudhistira Ganis K,dkk. 2018	KLASIFIKASI CITRA DENGAN MATRIKS KO-OKURENSI ARAS KEABUAN (Gray Level Co-occurrence Matrix- <i>GLCM</i>) PADA LIMA KELAS BIJI-BIJIAN	Melakukan klasifikasi pada biji-bijian dengan algoritma k-nn dengan lima kelas yang ada, data yang akan diklasifikasikan berasal dari ektaksi pola citra metode <i>GLCM</i>	Penelitian sekarang menggunakan algoritma <i>K-Means</i> dalam klasifikasi datanya guna melakukan konfirmasi terhadap setiap iterasi perhitungan yang terjadi.
3	Johan Wahyudi, dan Ihdahubbi maulida. 2019	PENGENALAN POLA CITRA KAIN TRADISIONAL MENGGUNAKAN <i>GLCM</i> DAN KNN	Mengkalsifikasikan hasil pola citra metode <i>glcm</i> dengan menggunakan 5 dari 6 ciri tekstur yang ada dengan metode knn terhadap kain tradisional persentasi keberhasilan tertinggi sebanyak 63%	penelitian sekarang menggunakan algoritma <i>K-Means</i> dalam klasifikasi guna melakukan konfirmasi terhadap setiap iterasi perhitungan.dan penggunaan ciri tekstur hanya 4. Dengan tingkat keberhasilan diatas 80%.

O. Kerangka Pikir

Dibawah ini gambaran alur kerangka pikir dalam proses berjalannya program dengan ekstraksi dan klasifikasi dengan metode yang di terapkan dalam mendeteksi penyakit pada daun cengkeh.

Para petani/pengurus tanaman cengkeh melakukan diagnosa terhadap gejala-gejala yang muncul pada tanaman mereka hanya dengan bermodalkan pengalaman atau bahkan mereka harus mencari informasi dari para petani/pengurus tanaman cengkeh lainnya yang lebih berpengalaman atau bahkan menghubungi pakar yang ahli. Kendala yang terjadi adalah para petani menambah biaya dan tenaga yang lebih di luar perawatan tanaman cengkeh.



Dengan melihat masalah yang ada, maka penulis akan membangun sebuah Sistem untuk dapat memberikan hasil diagnosa penyakit tanaman cengkeh hanya berdasarkan gambar dampak ataupun gejala pada daun tanaman cengkehnya.



tahap pertama yang di butuhkan adalah sampel uji sebagai acuan untuk klasifikasi jenis penyakitnya, ketika sample uji telah memiliki semua jenis penyakit yang ada pada sistem barulah dapat melakukan klasifikasi pada sampel latih yang akan diunggah pada sistem nantinya. Ketika pengguna mengunggah gambar gejala atau dampak pada daun tanaman cengkehnya gambar tersebut pertama-tama akan dilakukan *resizing* dan akan di ubah dari mode RGB ke mode greyscale kemudian dari mode greyscale itu akan dapat dibuat matriks ko-okurasi yang akan di ubah nilainya menjadi nilai ciri tekstur metode *GLCM* yaitu energi, kontras, entropi, homogenitas, nilai-nilai itu akan dikalsifikasikan menggunakan algorith *K-Means* berdasarkan nilai ciri tekstur dari sampel uji yang kemudian akan diketahui jenis penyakit nya dan akan ditampilkan nama penyakitnya beserta cara penanganannya.



Diharapkan dengan adanya aplikasi ini dapat membantu para petani/pengurus tanaman cengkeh dalam memberikan informasi tentang nama dan cara penanganan penyakit yang dialami, hanya dengan mengambil gambar terhadap gejala atau dampak yang terjadi pada daun tanaman cengkehnya.

Gambar 2.9. Kerangka Pikir

BAB III

METODE PENELITIAN

A. Penggambaran Sistem

1. Sistem berjalan.

Adapun prosedur sistem berjalan tentang pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh seperti gambar di bawah ini :



Gambar 3.1. Use Case Diagram Sistem Yang sedang berjalan

Pada gambar use case di atas dapat di lihat bahwa para petani/pengurus cengkeh jika terdapat gejala yang muncul pada tanaman cengkehnya, petani/pengurus cengkeh akan mencari informasi terkait gejala tersebut dan sumber informasi yang bisa didapatkan adalah pakar yang ahli dan petani/pengurus tanaman cengkeh yang lebih berpengalaman. Hasil diagnosa terhadap gejala yang muncul bisa diperoleh dari sumber informasi yang telah ada dan dapat menambah waktu dan biaya tentunya

demi mendapat sebuah informasi, atau berdasarkan pengalaman petani/pengurus cengkeh itu sendiri. Jika diagnosa terhadap penyakit telah didapatkan barulah bisa diberikan penanganan berupa pestisida atau sejenisnya.

B. Analisis Kebutuhan Sistem

1. Kebutuhan Fncional

1) *Functional* penginputan data

- a) Sistem memiliki fungsi penginputan data jenis penyakit pada tanamana cengkeh
- b) Sistem memiliki fungsi penginputan data sample uji dari foto terhadap gejala-gejala dari penyakit yang telah ada dan berdampak kepada daun.
- c) Sistem memiliki fungsi penginputan sample latih dari foto terhadap gejala-gejala yang terjadi pada daun yang belum terklasifikasikan dan akan diklasifikasikan terlebih dahulu oleh system
- d) Sistem akan memiliki fungsi inputan foto gejala yang ingin diklasifikasi penyakitnya.

2) *Functional* pemrosesan/pengolahan data

- a) Sistem memiliki fungsi dalam mengambil nilai ciri tekstru metode *GLCM* yang akan ditambahkan kedalam data sample uji maupun sample latih
- b) Sistem memiliki fungsi untuk melakukan klasifikasi jenis penyakit menggunakan algoritma *K-Means*.

- c) Sistem memiliki fungsi login untuk membedakan otoritas admin dan pengguna biasa.

3) *Functional Output*

- a) Sistem memiliki fungsi menampilkan jenis penyakit pada tanaman cengkeh
- b) Sistem memiliki fungsi menampilkan hasil diagnose dari klasifikasi terhadap jenis penyakit berdasarkan nilai ciri tekstur yang telah diekstrak dari foto gejala yang telah diunggah.

2. *Kebutuhan Non Functional*

1) Perangkat yang digunakan

- a) Sebuah komputer dengan spesifikasi sebagai berikut:

Proc : intel core i3-3240

RAM : 4 GB ddr3

HDD : 1 TB

VGA : Nvidia GTX 750-ti

- b) Peripheral tambahan untuk komputer seperti Keyboard dan LCD 19 inci

- c) Kamera dengan resolusi 16 *megapixel*

2) Aplikasi yang digunakan

- a) Sistem Operasi Windows 10

b) *Xampp*

c) Sublime text

d) Mozilla Firefox

C. Teknik Pengumpulan Data

Dalam pelaksanaan penelitian ini dibutuhkan data dan informasi sebagai bahan dasar dalam pengembangan sistem. Adapun metode pengumpulan data yang digunakan dalam penelitian ini yaitu sebagai berikut.

1. Teknik Observasi

Teknik observasi adalah teknik pengumpulan data dengan cara peneliti melakukan pengamatan secara langsung di lapangan. Pengamatat disebut observer yang diamati disebut observer. Dalam hal ini penulis mendatangi para petani/ pengurus tanaman cengkeh untuk mendapatkan sampel terhadap gejala yang terjadi pada daun tanaman cengkeh dan jenis penyakit apa yang dialami.

2. Studi literature

Studi literatur merupakan tahap pertama melakukan penelitian. Studi literatur dilakukan untuk mendapatkan gambaran umum maupun khusus mengenai objek maupun teori pendukung dalam penelitian ini. Studi literatur yang digunakan yakni: buku pedoman, buku online, jurnal online, dan skripsi terkait dengan penelitian yang dibutuhkan

3. Wawancara

Pengumpulan data dengan melakukan tanya jawab langsung dengan narasumber dari objek yang diteliti untuk memperoleh data yang diinginkan dalam kasus ini merupakan petani/pengurus tanaman cengkeh. Wawancara dilakukan guna mendapatkan sampel atas jenis penyakit tanaman cengkeh yang memiliki gejala fisik pada daunnya.

D. Metode Pengembangan Sistem

Dalam mengembangkan suatu sistem dibutuhkan metode pengembangan sistem yang akan digunakan sebagai petunjuk bagaimana dan apa yang harus dikerjakan selama proses pengembangannya. Pada aplikasi indeks berbasis *mobile* android ini, metode yang digunakan adalah metode pengembangan sistem secara *waterfall*, adapun tahapan – tahapannya yaitu sebagai berikut.

1. Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengetahui kebutuhan sistem yang akan dibangun yang dilakukan dengan cara mengumpulkan data melalui dokumen yang telah ada, yang kemudian akan diidentifikasi untuk desain sistem (desain *interface software*).

2. Desain Sistem

Pada tahap desain sistem atau perancangan aplikasi menggunakan UML (*Unified Modeling Language*) yang terdiri dari *Use Case Diagram*, *Class Diagram*, *Sequence Diagram*, *State Diagram*, *Activity Diagram*, dan *Deployment Diagram*.

3. Penulisan Kode Program

Pada tahap ini, desain sistem yang telah dirancang akan diubah ke dalam bahasa pemrograman bahasa yang bisa dimengerti oleh komputer. Proses penulisan kode program dilakukan dengan menggunakan bahasa pemrograman *PHP* dan *MySQL* dari layanan *PHPmyadmin* sebagai basis datanya.

4. *Testing*

Selanjutnya tahap pengujian (*testing*). Adapun metode pengujian yang digunakan adalah metode pengujian *black box*. Pengujian ini dilakukan guna menemukan kesalahan - kesalahan yang mungkin terjadi pada aplikasi. Jika ditemukan kesalahan (*error*) maka akan diperbaiki dengan kembali ke tahapan sebelumnya.

5. *Maintenance*

Tahap terakhir adalah pemeliharaan (*maintenance*). Pemeliharaan dapat dilakukan dengan beberapa cara untuk mengamankan dan menjaga data seperti, *backup database* dan melakukan pengembangan sistem serta pemeliharaan data dapat dilakukan dengan pengarsipan melalui media komputer dan proses pemeliharaannya jauh lebih mudah jika dilakukan dengan komputerisasi.

BAB IV

PERANCANGAN SISTEM

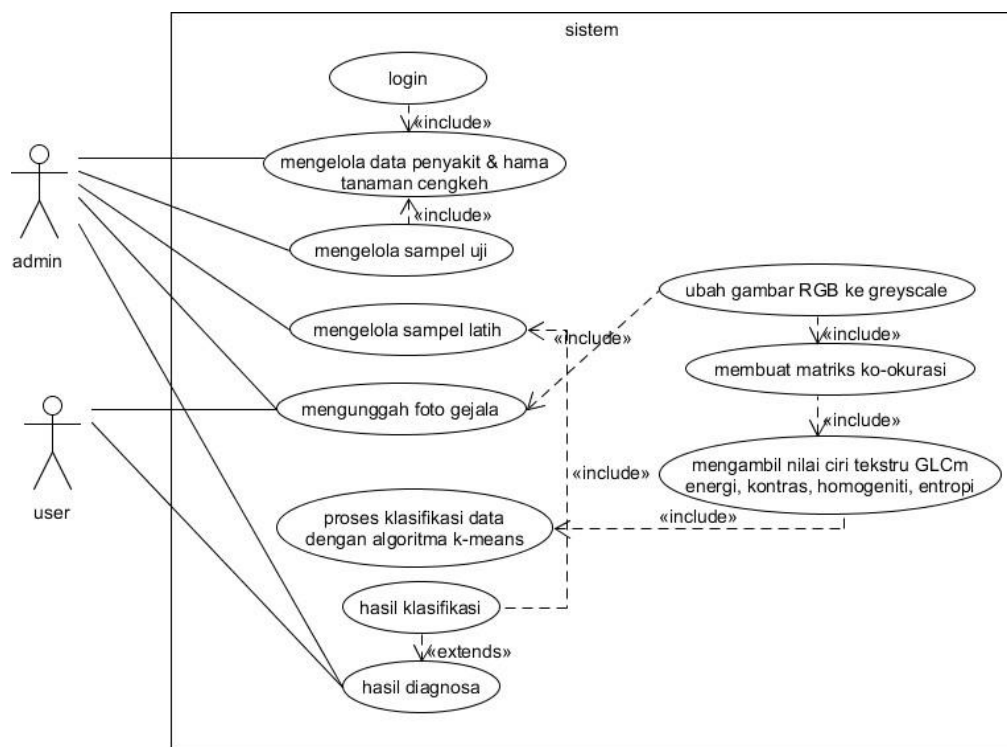
A. Pemodelan Sistem

Pemodelan sistem adalah proses membangun atau membentuk sebuah model dari suatu sistem nyata dalam bahasa formal tertentu. Untuk memodelkan suatu sistem maka diperlukan gambaran permasalahan yang ada serta hubungan antar komponen, variabel dan parameter-parameter sistemnya. Sehingga agar kita dapat memodelkan suatu masalah yang rumit maka kita memerlukan suatu metode untuk menggambarkan suatu situasi. Untuk itu proses pemodelan sangat penting didalam proses perancangan sistem yang dimana akan menggambarkan alur dari sistem itu sendiri.

Pada pemodelan ataupun perancangan desain sistem, item-item dan komponen dirancang dengan tujuan untuk memudahkan dan mengefisienkan kinerja dan aktifitas kerja serta memberikan gambaran umum tentang bagaimana mekanisme yang tepat untuk mendesain suatu sistem sesuai dengan kebutuhan dari pengguna, keuntungan dan kelebihan yang disajikan lewat suatu perancangan aplikasi berdasarkan tujuan dan rumusan masalah yang telah dibuat. Adapaun pemodelan yang akan digunakan pada perancangan kalin ini yaitu, *Unified Modeling Language (UML)* yang di dalamnya terdiri dari *Use Case*, *Activity Diagram*, dan *sequence diagram*. Berikut pemodelan yang dibuat untuk pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh.

1. Use Case Diagram

Setelah analisa sistem yang berjalan dilaksanakan maka kemudian dilakukan desain sistem yang baru. Tujuan utama dari sistem yang baru adalah penyempurnaan dari sistem yang lama atau yang telah ada untuk dapat digunakan oleh pengguna.



Gambar 4.1. Use Case diagram Sistem Yang Diusulkan

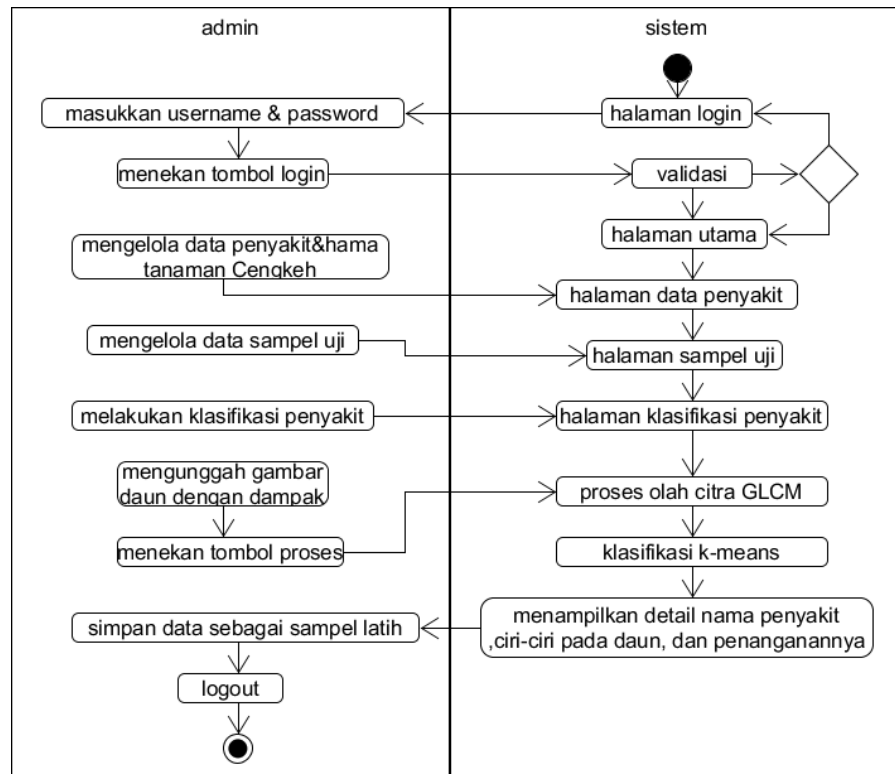
Deskripsi:

- Admin login ke system
- Admin mengelola data penyakit & hama tanaman cengkeh
- Admin dapat mengelola sampel uji.
- Admin dapat mengelola sampel latih.

- e. Admin dapat melakukan proses klasifikasi penyakit dengan mengunggah gambar dampak yang terjadi pada daun dan gambar tersebut diolah dan di ekstrak ciri citranya dengan metode *GLCM* yang kemudian ciri tersebut akan diklasifikasikan menggunakan metode *K-Means*.
- f. Admin dapat melihat hasil dari klasifikasi akan penyakit terhadap gambar yang sebelumnya di unggah
- g. Pengguna biasa (petani/ pengurus cengkeh) dapat melakukan proses klasifikasi penyakit dengan mengunggah gambar dampak yang terjadi pada daun dan gambar tersebut diolah dan di ekstrak ciri citranya dengan metode *GLCM* yang kemudian ciri tersebut akan diklasifikasikan menggunakan metode *K-Means*.
- h. Pengguna biasa (petani/ pengurus cengkeh) dapat melihat hasil dari klasifikasi akan penyakit terhadap gambar yang sebelumnya di unggah.

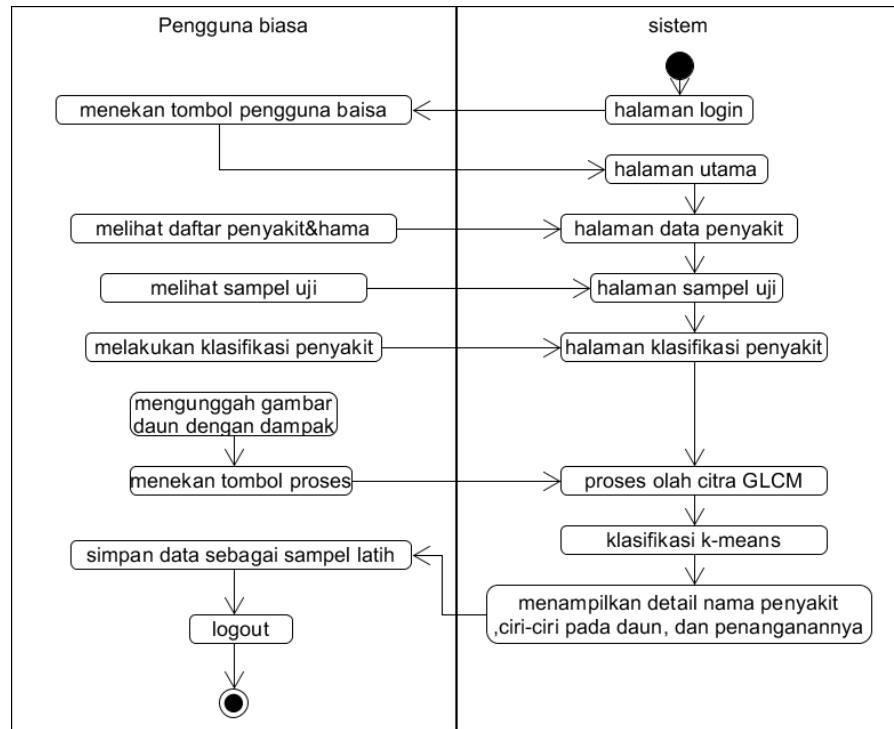
2. Activity Diagram

a. Admin



Gambar 4.2. Activity Diagram admin

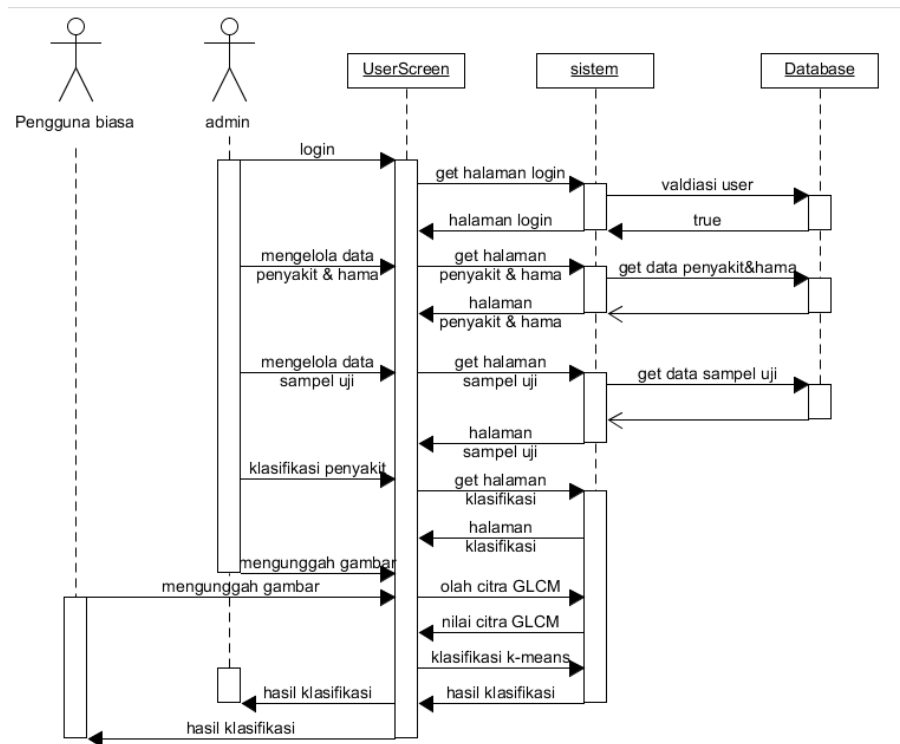
b. Petani/pengurus tanaman cengkeh



Gambar 4.3. Activity Diagram pengguna biasa

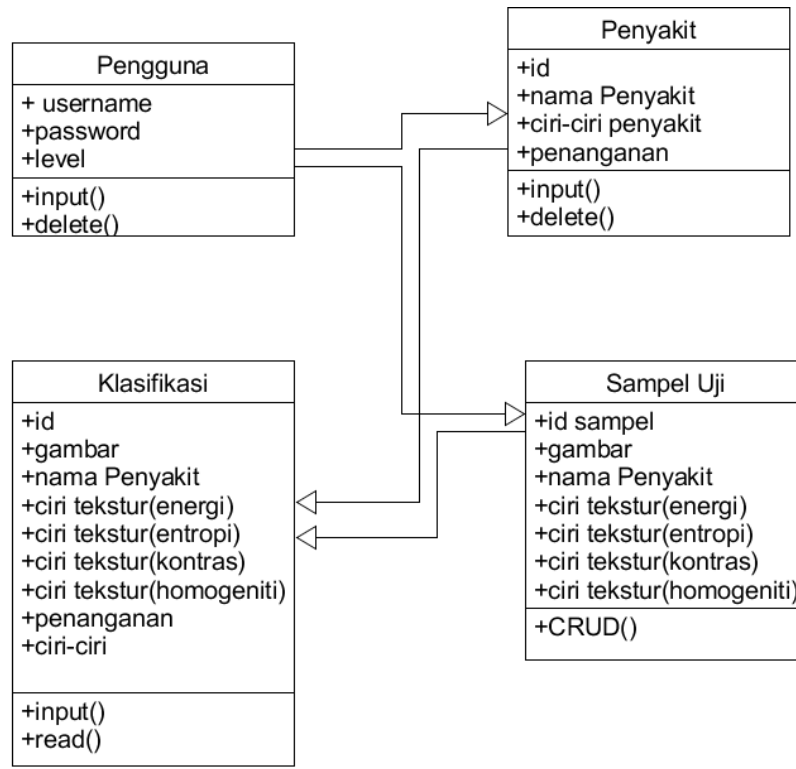
3. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segi empat bernama



Gambar 4.4. *Sequence Diagram*

4. Class Diagram



Gambar 4.5. Class Diagram

B. Perancangan Database

Tabel 4.1. users

No	Nama Field	Type	Ukuran	Keterangan
1	username	varchar	50	
2	password	varchar	50	

Tabel 4.2. Penyakit

No	Nama Field	Type	Ukuran	Keterangan
1	Id	Int	5	auto_increment
2	nama	varchar	100	
3	Penanganan	Varchar	1000	
4	Ciri	Varchar	500	

Tabel 4.3. data_awal

No	Nama Field	Type	Ukuran	Keterangan
1	Id	Int	5	auto_increment
2	Kode_penyakit	varchar	5	
3	gambar	Varchar	50	
4	Nama_pengguna	varchar	100	
5	No_hp	int	12	
6	energi	varchar	10	
7	Entropi	varchar	10	
8	Homogeniti	varchar	10	
9	Kontras	varchar	10	
10	status	varchar	1000	

Tabel kebutuhan algoritma klasifikasi *K-Means*

Tabel 4.4. center

No	Nama Field	Type	Ukuran	Keterangan
1	Id	varchar	5	
2	energi	varchar	10	
3	Entropi	varchar	10	
4	Homogeniti	varchar	10	
5	Kontras	varchar	10	

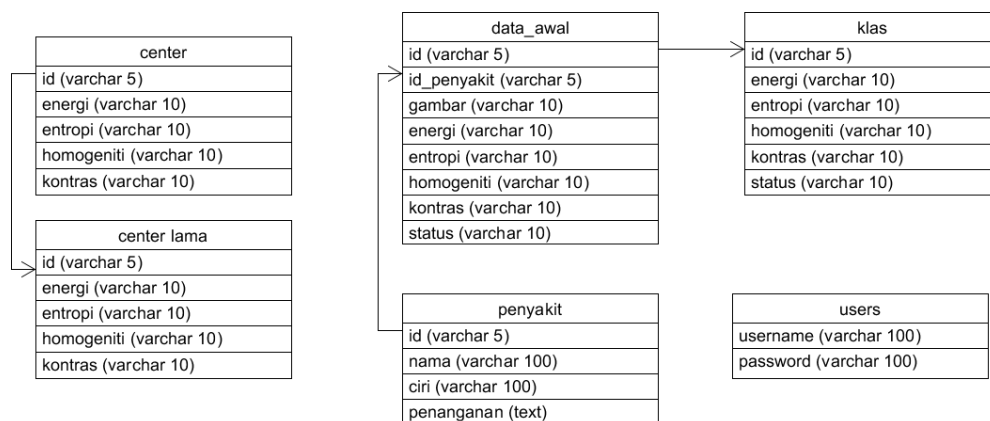
Tabel 4.5. center_lama

No	Nama Field	Type	Ukuran	Keterangan
1	Id	varchar	5	
2	energi	varchar	10	
3	Entropi	varchar	10	
4	Homogeniti	varchar	10	
5	Kontras	varchar	10	

Tabel 4.6. klas

No	Nama Field	Type	Ukuran	Keterangan
1	Id	varchar	5	
2	energi	varchar	10	
3	Entropi	varchar	10	
4	Homogeniti	varchar	10	
5	Kontras	varchar	10	
6	status	varchar	10	

Adapun rancangan relasi terhadap tabel-tabel yang telah dibuat adalah sebagai berikut:



Gambar 4.6. Rancangan relasi tabel

C. Perancangan Antar Muka

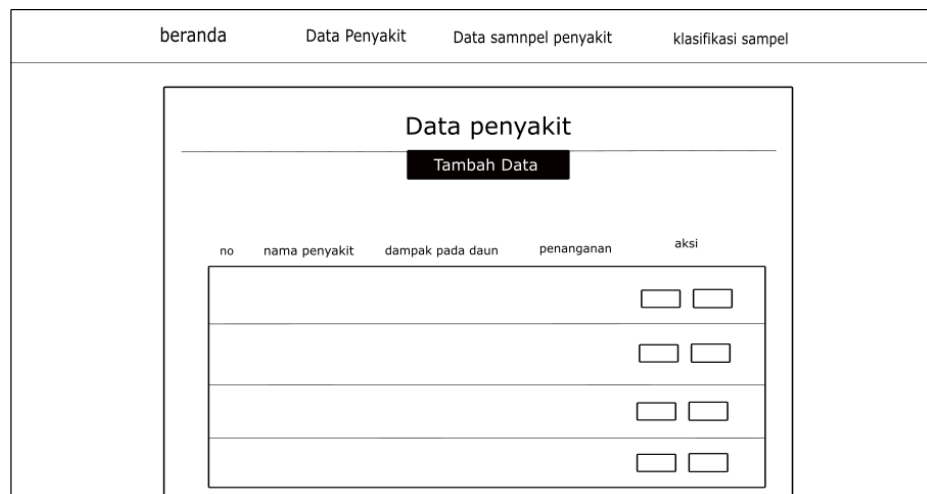
1. Rancangan *Output*

a. Rancangan *Output* Halaman Utama



Gambar 4.7. Rancangan *Output* Halaman Utama

b. Rancangan *Output* Data penyakit



Gambar 4.8. Rancangan *Output* Data penyakit

c. Rancangan *Output* data sampel penyakit

beranda	Data Penyakit	Data sampel penyakit	klasifikasi sampel
---------	---------------	----------------------	--------------------

Data sampel penyakit

Tambah Data

no	gambar	nama penyakit	ciri faktor			aksi
			energi	entropi	kontras homogeniti	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Gambar 4.9. Rancangan *Output* data sampel penyakit

d. Rancangan *Output* Data Uji

beranda	Data Penyakit	sampel latih	sampel uji	klasifikasi sampel
---------	---------------	--------------	------------	--------------------

Data sampel uji

Cetak laporan data

<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>

Gambar 4.10. Rancangan *Output* data latih

e. Rancangan *Output* klasifikasi sampel

The wireframe shows a web page for sample classification. At the top, there is a navigation menu with four items: 'beranda', 'Data Penyakit', 'Data sampel penyakit', and 'klasifikasi sampel'. The main content area is enclosed in a box with the title 'Klasifikasi sampel'. Below the title, there are two main sections. The first section, labeled 'Pilih gambar sampel', contains a 'Browse' button and a 'Proses' button. The second section, labeled 'preview', contains a large empty rectangular box for displaying the sample image.

Gambar 4.11. Rancangan *Output* klasifikasi sampel

2. Rancangan Input

a. Rancangan *Input Login*

The wireframe shows a login page with a central box containing the title 'Silahkan Masuk'. Below the title, there are two input fields: 'Nama Pengguna' and 'Kata Sandi'. Below these fields is a 'Masuk' button.

Gambar 4.12. Rancangan Input Login

b. Rancangan *Input* tambah data penyakit

The wireframe shows a navigation bar at the top with four items: 'beranda', 'Data Penyakit', 'Data sampel penyakit', and 'klasifikasi sampel'. Below the navigation bar is a main content area titled 'Data sampel penyakit'. Inside this area, there is a 'Tambah Data' button. Below the button is the text 'Tambah Data'. A modal form titled 'Penyakit Ke- n' is displayed, containing three input fields labeled 'Nama', 'Dampak pada daun', and 'Penanganan'. At the bottom of the modal are two buttons: 'Simpan' and 'batal'.

Gambar 4.13. Rancangan *Input* tambah data penyakit

c. Rancangan *Input* Ubah data Penyakit

The wireframe shows a navigation bar at the top with four items: 'beranda', 'Data Penyakit', 'Data sampel penyakit', and 'klasifikasi sampel'. Below the navigation bar is a main content area titled 'Data sampel penyakit'. Inside this area, there is a 'Tambah Data' button. Below the button is the text 'Ubah Data'. A modal form titled 'Penyakit Ke- n' is displayed, containing three input fields labeled 'Nama', 'Dampak pada daun', and 'Penanganan'. At the bottom of the modal are two buttons: 'Simpan' and 'batal'.

Gambar 4.14. Rancangan *Input* Ubah data penyakit

d. Rancangan *Input* Tambah sampel latihan penyakit

The wireframe shows a navigation bar at the top with four items: 'beranda', 'Data Penyakit', 'Data sampel penyakit', and 'klasifikasi sampel'. Below this is a main container titled 'Data sampel penyakit'. Inside this container, there is a sub-section titled 'Tambah Data' with a black button labeled 'Tambah Data'. Below the 'Tambah Data' section, there is a form area titled 'Pilih gambar sampel' on the left and 'preview' on the right. The 'Pilih gambar sampel' area contains a 'Browse' button and a 'Proses' button. The 'preview' area contains a large empty rectangular box.

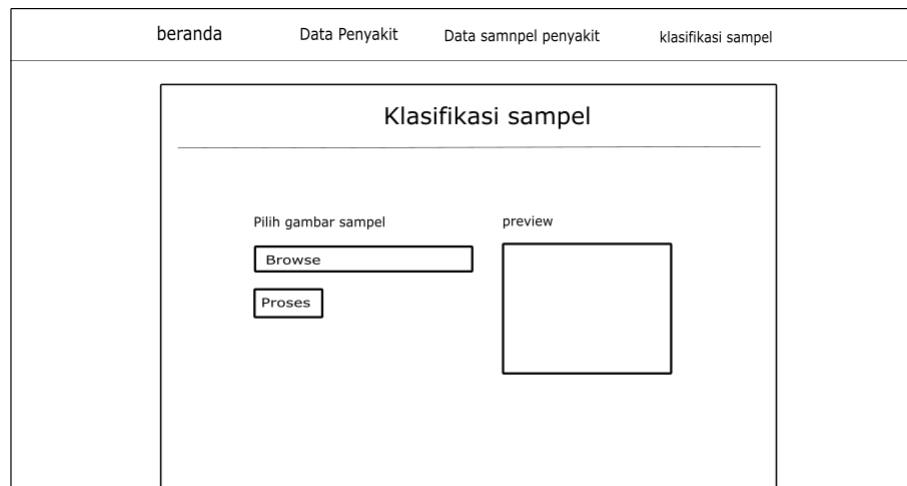
Gambar 4.15. Rancangan *Input* Tambah sampel latihan penyakit

e. Rancangan *Input* ubah sampel latihan penyakit

The wireframe shows a navigation bar at the top with four items: 'beranda', 'Data Penyakit', 'Data sampel penyakit', and 'klasifikasi sampel'. Below this is a main container titled 'Data sampel penyakit'. Inside this container, there is a sub-section titled 'Ubah Data' with a black button labeled 'Tambah Data'. Below the 'Ubah Data' section, there is a form area. On the left, there is a 'gambar sampel' label above a large empty rectangular box. Below the image box is a 'Jenis Penyakit' label above a text input field. At the bottom left of the form area are two buttons: 'Simpan' and 'batal'. On the right side of the form area, there is a 'Ciri fitur' label followed by a list of features: 'energi', 'entropi', 'Homogeniti', and 'Kontras'.

Gambar 4.16. Rancangan *Input* ubah sampel data latihan

f. Rancangan *Input* klasifikasi sampel penyakit



Gambar 4.17. Rancangan *Input* klasifikasi sampel penyakit

D. Hasil Penelitian

1. Implementasi Algoritma

Tahapan yang digunakan dalam implementasi pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh yaitu sebagai berikut:

a. Olah citra dengan metode *GLCM*

1) *Image Acquisition*

Akuisisi citra merupakan tahap awal, pada tahap ini citra akan diambil/ditangkap. Kamera akan dipergunakan untuk tahapan ini. Kamera yang umum dipergunakan pada tahap ini, dapat berupa kamera digital. Citra yang dimaksudkan yaitu gambar daun cengkeh yang memiliki dampak/gejala penyakit dengan ukuran 150x150 pixel.



Gambar 4.18. gambar citra asli

2) *Pra Processing*

Pada Tahap ini dilakukan proses grayscale terhadap citra yang akan diekstrak ciri fitur *GLCM*-nya.

- a) Proses grayscale : proses untuk mengubah gambar yang memiliki warna menjadi gambar yang memiliki tingkat warna ke abu-abuan (grayscale).
- b) Gambar yang akan di-gray-scailing nilai tiap titik akan disamakan dengan nilai red-green-blunya sehingga untuk tiap titik hanya memiliki 1 nilai saja, yang disebut nilai gray-level-nya.
- c) Pada dasarnya proses ini dilakukan dengan meratakan nilai pixel dari 3 nilai RGB menjadi 1 nilai.



Gambar 4.19. Gambar citra greyscale

implementasi pada sistem:

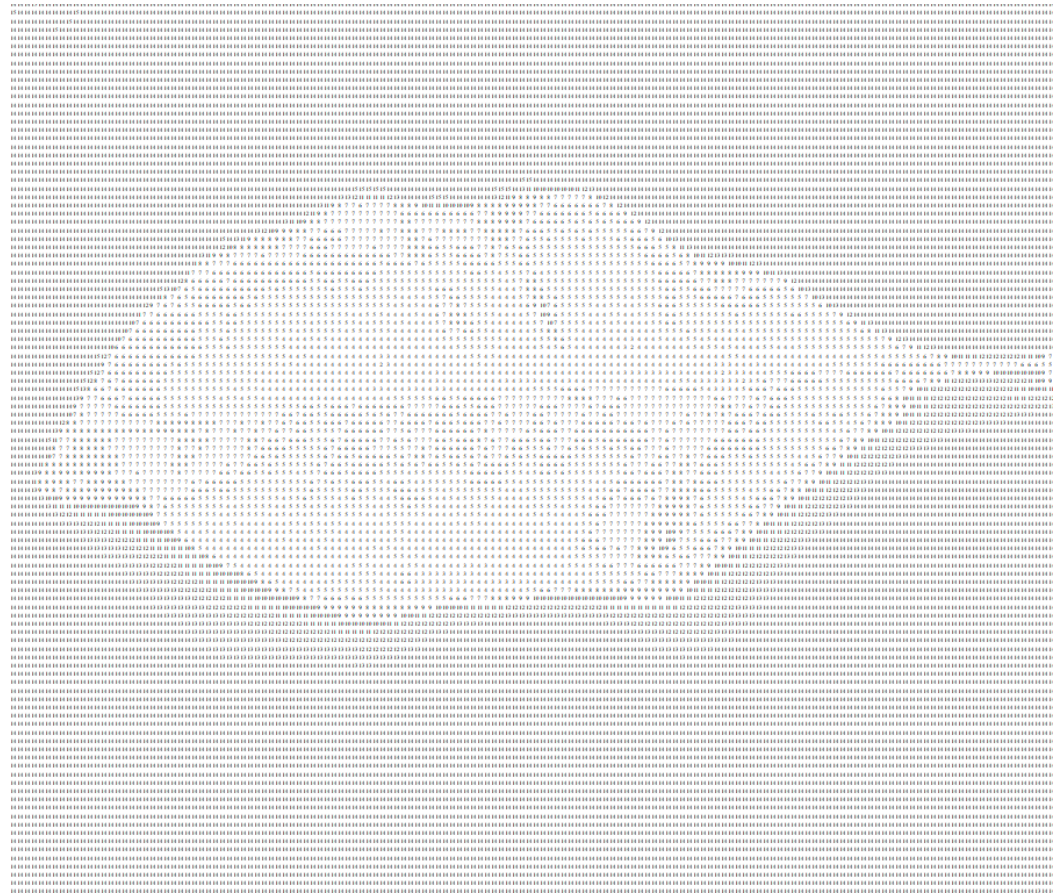
```
<?php
$w = imagesx($im);
$h = imagesy($im);
$r = $g = $b = 0;
for($y = 0; $y < $h; $y++) {
for($x = 0; $x < $w; $x++) {
  $rgb = imagecolorat($im, $x, $y);
  $r = ($rgb >> 16) & 0xFF;
  $g = ($rgb >> 8) & 0xFF;
  $b = $rgb & 0xFF;
  $scl=(0.21*$r)+(0.72*$g)+(0.07*$b);
  $abu/$x/$y/=(($scl*15)/255);
  imagesetpixel($im, $x, $y, imagecolorallocate($im, $scl, $scl, $scl));}
ob_start();
imagejpeg($im);
$content = ob_get_contents();
ob_end_clean();
$base64 = "data:image/jpeg;base64," . base64_encode($content);
?>
```

Proses diatas merupakan proses dimana mengubah citra *RGB* dari gambar sebelumnya menjadi citra skala keabuan dengan skala 4 bit dimana rentan skala nya dari 0-15 saja.

3) Segmenntasi

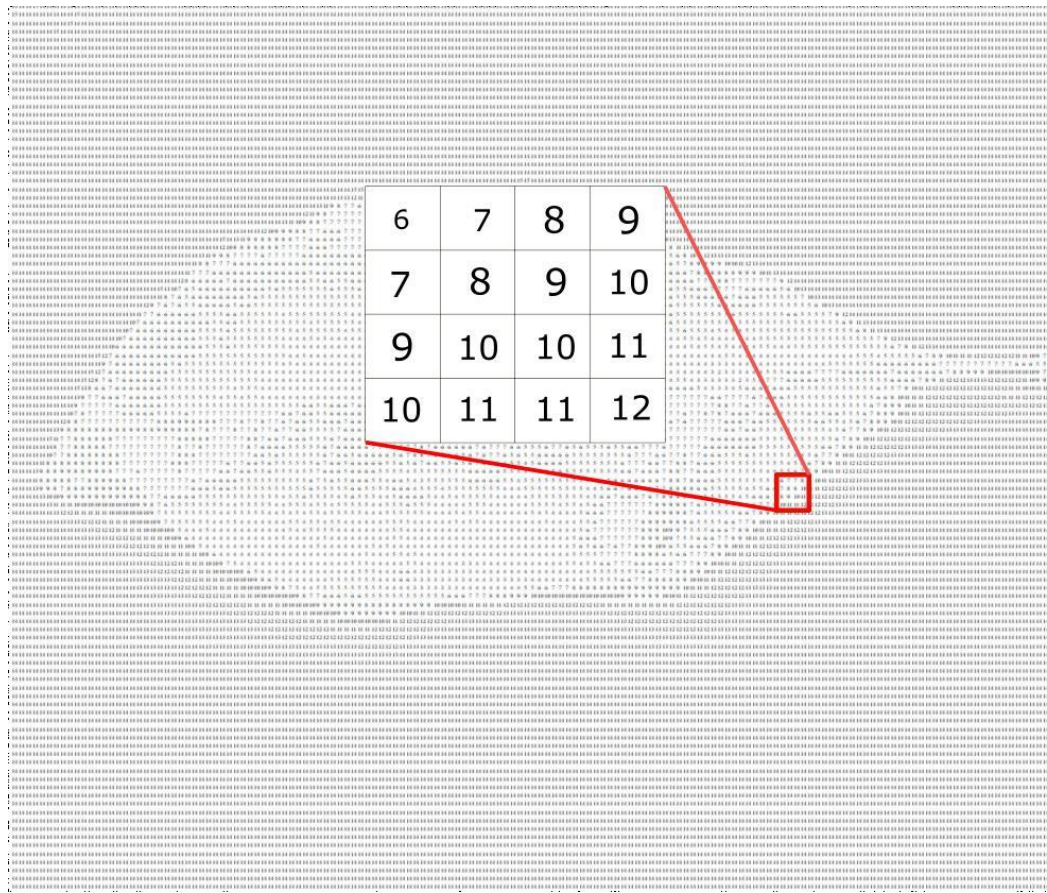
Pada tahap ini citra yang telah diubah menjadi grayscale akan diekstrak empat ciri fitur *GLCM*-nya yaitu energi, entropi, homogeniti, dan kontras menggunakan rumus masing-masing ke-empat fitur.

Berikut merupakan tabel greyscale citra.



Gambar 4. 20. Hasil proses perubahan citra *RGB* ke *Gray scale*

Berdasarkan matriks dari proses perubahan citra *RGB* ke *gray scale* akan dilanjutkan ketahap pencarian nilai matriks 0 derajat dan seterusnya berdasarkan matriks 4x4 dari matriks tersebut yang telah ditentukan.



Gambar 4.21. Sampel perhitungan

Tabel 4.7. tabel matriks perhitungan

6	7	8	9
7	8	9	10
9	10	10	11
10	11	11	12

Berdasarkan nilai skala keabuan yang telah diubah maka dapat dibentuk matriks ko-okurasi dengan derajat 0, 45, 90, dan 135 sebagai berikut:

a) Pencarian matriks 0 derajat:

Tabel 4.8. matriks 0 derajat

	6	7	8	9	10	11	12
6	0	1	0	0	0	0	0
7	0	0	2	0	0	0	0
8	0	0	0	2	0	0	0
9	0	0	0	0	2	0	0
10	0	0	0	0	1	2	0
11	0	0	0	0	0	1	1
12	0	0	0	0	0	0	0

Tabel 4.9. matriks transpose 0 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0
8	0	2	0	0	0	0	0
9	0	0	2	0	0	0	0
10	0	0	0	2	1	0	0
11	0	0	0	0	2	1	0
12	0	0	0	0	0	1	0

Tabel 4.10. Nilai instensitas 0 derajat

	6	7	8	9	10	11	12
6	0	1	0	0	0	0	0
7	1	0	2	0	0	0	0
8	0	2	0	2	0	0	0
9	0	0	2	0	2	0	0
10	0	0	0	2	1	1	0
11	0	0	0	0	2	1	1
12	0	0	0	0	0	1	0

Nilai instensitas didapat dari matriks awal yang ditambah dengan matriks transposenya, nilai instensitas 0 derajat memiliki total nilai 23.

Mencari nilai matriks ko-okurasi 0 derajat dengan cara membagi setiap nilai intesitas 0 derajat dengan total nilainya.

Tabel 4.11. Matriks ko-okurasi 0 derajat

	6	7	8	9	10	11	12
6	0	0.043	0	0	0	0	0
7	0.043	0	0.086	0	0	0	0
8	0	0.086	0	0.086	0	0	0
9	0	0	0.086	0	0.086	0	0
10	0	0	0	0.086	0.043	0.043	0
11	0	0	0	0	0.086	0.043	0.043
12	0	0	0	0	0	0.043	0

Implemetasi pada sistem:

```
<?php
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mh|$i||$j|=0;
        for ($m=0; $m <$h ; $m++) {
            for ($n=0; $n <$w-1 ; $n++) {
```

```

                                if ( round($abu|$m|$n) == $i and
round($abu|$m|$n+1) == $j ) {
                                $mh|$i|$j|++;
                                }
                                }
                                }
                                }
}
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mt|$j|$i|=$mh|$i|$j|;
    }
}

$total=0;
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mf|$i|$j|=$mt|$i|$j|+$mh|$i|$j|;
        $total+=$mf|$i|$j|;
    }
}
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $my|$i|$j|=$mf|$i|$j|/$total;
    }
}

?>

```

Source code diatas merupakan tahap pencarian matriks ko-
okurasi 0 derajat.

b) Pencarian matriks 45 derajat:

Tabel 4.12 matriks 45 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	0	1	0	0	0	0	0
8	0	0	1	0	0	0	0
9	0	0	1	1	0	0	0
10	0	0	0	1	2	0	0
11	0	0	0	0	1	1	0
12	0	0	0	0	0	0	0

Tabel 4.13. matriks transpose 45 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	0	1	0	0	0	0	0
8	0	0	1	1	0	0	0
9	0	0	0	1	1	0	0
10	0	0	0	0	2	1	0
11	0	0	0	0	0	1	0
12	0	0	0	0	0	0	0

Tabel 4.14. Nilai instensitas 45 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	0	2	0	0	0	0	0
8	0	0	2	1	0	0	0
9	0	0	1	2	1	0	0
10	0	0	0	1	4	1	0
11	0	0	0	0	1	2	0
12	0	0	0	0	0	0	0

Nilai instensitas didapat dari matriks awal yang ditambah dengan matriks transposenya, nilai instensitas 45 derajat memiliki total nilai 18.

Mencari nilai matriks ko-okurasi 45 derajat dengan cara membagi setiap nilai intesitas 45 derajat dengan total nilainya.

Tabel 4.15. Matriks ko-okurasi 45 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	0	0.111	0	0	0	0	0
8	0	0	0.111	0.055	0	0	0
9	0	0	0.055	0.111	0.055	0	0
10	0	0	0	0.055	0.222	0.055	0
11	0	0	0	0	0.055	0.111	0
12	0	0	0	0	0	0	0

Implementasi pada sistem:

```

<?php
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mh|$i||$j|=0;
        for ($m=0; $m <$h-1 ; $m++) {
            for ($n=0; $n <$w-1 ; $n++) {
                if ( round($abu|$m+1||$n) == $i and
round($abu|$m||$n+1) == $j ) {
                    $mh|$i||$j|++;
                }
            }
        }
    }
}
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mt|$j||$i|=$mh|$i||$j|;
    }
}

```

```

}
$total=0;
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mf[$i][$j]=$mt[$i][$j]+$mh[$i][$j];
        $total+=$mf[$i][$j];
    }
}
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $my[$i][$j]=$mf[$i][$j]/$total;
    }
}
?>

```

Source code diatas merupakan tahap pencarian matriks ko-
okurasi 45 derajat

c) Pencarian matriks 90 derajat:

Tabel 4.16. Matriks 90 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0
8	0	1	0	0	0	0	0
9	0	1	1	0	0	0	0
10	0	1	3	0	0	0	0
11	0	0	0	0	3	0	0
12	0	0	0	0	0	1	0

Tabel 4.17. Matriks transpose 90 derajat

	6	7	8	9	10	11	12
6	0	1	0	0	0	0	0
7	0	0	1	1	1	0	0
8	0	0	0	1	3	0	0
9	0	0	0	0	0	0	0
10	0	0	0	0	0	3	0
11	0	0	0	0	0	0	1
12	0	0	0	0	0	0	0

Tabel 4.18. nilai instensitas 90 derajat

	6	7	8	9	10	11	12
6	0	1	0	0	0	0	0
7	1	0	1	1	1	0	0
8	0	1	0	1	3	0	0
9	0	1	1	0	0	0	0
10	0	1	3	0	0	3	0
11	0	0	0	0	3	0	1
12	0	0	0	0	0	1	0

Nilai instensitas didapat dari matriks awal yang ditambah dengan matriks transposenya, nilai instensitas 90 derajat memiliki total nilai 24.

Mencari nilai matriks ko-okurasi 90 derajat dengan cara membagi setiap nilai intesitas 90 derajat dengan total nilainya.

Tabel 4.19. Matriks ko-okurasi 90 derajat

	6	7	8	9	10	11	12
6	0	0.041	0	0	0	0	0
7	0.041	0	0.041	0.041	0.041	0	0
8	0	0.041	0	0.041	0.125	0	0
9	0	0.041	0.041	0	0	0	0
10	0	0.041	0.125	0	0	0.125	0
11	0	0	0	0	0.125	0	0.041
12	0	0	0	0	0	0.041	0

Implementasi pada sistem:

```

<?php
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mh|$i||$j|=0;
        for ($m=0; $m <$h-1 ; $m++) {
            for ($n=0; $n <$w ; $n++) {
                if ( round($abu|$m+1||$n) == $i and
round($abu|$m||$n) == $j ) {

```


d) Pencarian matriks 135 derajat:

Tabel 4.20. Matriks 135 derajat

	6	7	8	9	10	11	12
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0
8	1	0	0	0	0	0	0
9	0	1	0	0	0	0	0
10	0	1	2	0	0	0	0
11	0	0	0	1	1	0	0
12	0	0	0	0	1	0	0

Tabel 4.21. Matriks transpose 135 derajat

	6	7	8	9	10	11	12
6	0	0	1	0	0	0	0
7	0	0	0	1	1	0	0
8	0	0	0	0	2	0	0
9	0	0	0	0	0	1	0
10	0	0	0	0	0	1	1
11	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0

Tabel 4.22. Nilai instensitas 135 derajat

	6	7	8	9	10	11	12
6	0	0	1	0	0	0	0
7	0	0	0	1	1	0	0
8	1	0	0	0	2	0	0
9	0	1	0	0	0	1	0
10	0	1	2	0	0	1	1
11	0	0	0	1	1	0	0
12	0	0	0	0	1	0	0

Nilai instensitas didapat dari matriks awal yang ditambah dengan matriks transposenya, nilai instensitas 135 derajat memiliki total nilai 16.

Mencari nilai matriks ko-okurasi 135 derajat dengan cara membagi setiap nilai intesitas 135 derajat dengan total nilainya.

Tabel 4.23. Matriks ko-okurasi 135 derajat

	6	7	8	9	10	11	12
6	0	0	0.062	0	0	0	0
7	0	0	0	0.062	0.062	0	0
8	0.062	0	0	0	0.125	0	0
9	0	0.062	0	0	0	0.062	0
10	0	0.062	0.125	0	0	0.062	0.062
11	0	0	0	0.062	0.062	0	0
12	0	0	0	0	0.062	0	0

Implementasi pada sistem:

```

<?php
//135
for ($i=0; $i < $pc ; $i++) {
    for ($j=0; $j < $pc ; $j++) {
        $mh|$i|$j|=0;
        for ($m=0; $m < $h-1 ; $m++) {
            for ($n=0; $n < $w-1 ; $n++) {
                if ( round($abu|$m+1|$n+1) == $i and
round($abu|$m|$n) == $j ) {
                    $mh|$i|$j|++;
                }
            }
        }
    }
}
}
}

```

```

for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mt|$j|$i|=$mh|$i|$j|;
    }
}

$total=0;
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $mf|$i|$j|=$mt|$i|$j|+$mh|$i|$j|;
        $total+=$mf|$i|$j|;
    }
}
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $my|$i|$j|=$mf|$i|$j|/$total;
    }
}

?>

```

Source code diatas merupakan pencarian matriks ko-okurasi 135 derajat.

Setelah mendapatkan setiap matriks ko-okurasi dari ke-empat derajat yang ada kemudian akan di hitung ekstraksi ciri fitur *GLCM* dari ke-empat derajat yang ada.

a. Fitur Energi :

Dalam fitur energi pada *GLCM* ini dilakukan perhitungan dengan mengkuadratkan semua nilai hasil dari matriks ko-okurasi, dan di jumlahkan seluruh hasil dari seluruh nilai kuadratnya.

$$\sum \sum ()$$

Tabel 4.24. Tabel hasil fitur energi

Fitur Energi	
Ko-okurasi 0 derajat	0.0647
Ko-okurasi 45 derajat	0.1167
Ko-okurasi 90 derajat	0.0826
Ko-okurasi 135 derajat	0.0773

b. Fitur Entropi:

$$\sum \sum () ()$$

$$\begin{aligned} & (\quad) (\quad) \\ & (\quad) (\quad) \\ & () () \end{aligned}$$

() ()
() ()
() ()
() ()

()

()()

() ()
() ()
() ()
() ()

()

()()

() ()
() ()
() ()
() ()
() ()
() ()

()()

()()

() ()
() ()
() ()
() ()
() ()

()()

Tabel 4.26. tabel hasil fitur homogeniti

Fitur Homogeniti	
Ko-okurasi 0 derajat	0.4945
Ko-okurasi 45 derajat	0.831
Ko-okurasi 90 derajat	0.4406
Ko-okurasi 135 derajat	0.3416

d. Fitur Kontras:

$$\sum \sum ()$$

$$\begin{aligned} & (\quad) (\quad) (\quad) (\quad) \\ & \quad) (\quad) (\quad) (\quad) (\quad) \\ &) () (\quad) (\quad) (\quad) (\quad) \\ &) () (\quad) (\quad) (\quad) (\quad) \\ & \quad) () \end{aligned}$$

$$\begin{aligned} & (\quad) (\quad) (\quad) (\quad) \\ & \quad) () () \\ &) () (\quad) (\quad) (\quad) (\quad) \\ & \quad) (\quad) (\quad) (\quad) (\quad) \\ &) \end{aligned}$$

$$\begin{aligned} & (\quad) (\quad) (\quad) (\quad) \\ & \quad) (\quad) (\quad) (\quad) (\quad) \\ &) () (\quad) (\quad) (\quad) (\quad) \\ &) () (\quad) (\quad) (\quad) (\quad) \\ & \quad) (\quad) (\quad) (\quad) (\quad) \\ &) \end{aligned}$$

```

        (          ) (          ) (
            ) (          ) (          ) (
        ) ( ) (
            ) (          ) (          ) (
        ) ( ) (
            ) ( )
    
```

Tabel 4.27. Tabel hasil fitur kontras

Fitur Kontras	
Ko-okurasi 0 derajat	0.817
Ko-okurasi 45 derajat	0.33
Ko-okurasi 90 derajat	1.324
Ko-okurasi 135 derajat	1.988

Implementasi pada sistem:

```

for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $energi0+= ($my|$i|$j)*($my|$i|$j);
        $kontras0+=((($i-$j)*($i-$j))*$my|$i|$j);
        $homo0+=($my|$i|$j)/(1+((($i-$j)*($i-$j)));
        $entropi0+=($my|$i|$j)*0.301*$my|$i|$j;
    }
}

for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $energi45+= ($my|$i|$j)*($my|$i|$j);
        $kontras45+=((($i-$j)*($i-$j))*$my|$i|$j);
        $homo45+=($my|$i|$j)/(1+((($i-$j)*($i-$j)));
        $entropi45+=($my|$i|$j)*0.301*$my|$i|$j;
    }
}

for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $energi9+= ($my|$i|$j)*($my|$i|$j);
    }
}
    
```

```

$kontras9+=$((i-j)*(i-j))*$my|i||$j|;
$homo9+=$my|i||$j|/(1+((i-j)*(i-j)));
$entropi9+=$my|i||$j|*0.301*$my|i||$j|;

}
}
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $energi135+= ($my|i||$j|)*($my|i||$j|);
        $kontras135+=$((i-j)*(i-j))*$my|i||$j|;
        $homo135+=$my|i||$j|/(1+((i-j)*(i-j)));
        $entropi135+=$my|i||$j|*0.301*$my|i||$j|;

    }
}
?>

```

Source code diatas merupakan tahap perhitungan nilai fitur GLCM yang akan digunakan yaitu energi, entropi, kontras, dan homogeniti dari setiap derajat yang ada.

4) *Post Processing*

Pada tahap ini akan dihasilkan nilai ciri ke-empat fitur *GLCM* yang diperoleh dari citra keabuan dan telah di rata-ratakan untuk setiap nilai fiturnya yang kemudian akan memasuki tahap klasifikasi.

Tabel 4.28. rata-rata ciri fitur ke-empat derajat

Rata-rata ciri fotir GCLM	
Energi	0.0853
Entropi	0.0245
Homogeniti	0.5269
Kontras	1.1147

Menghasilkan

Sampel Uji Ke-22	
Gambar Sampel:	Ciri Faktor:
	Energi :0.41221764
	Entropi :0.12407751
	Kontras :0.31672935
	Homogeniti :0.93585564

Gambar 4.22. hasil olah fitur *GLCM*

Implementasi pada sistem:





```
<?php
for ($i=0; $i <$pc ; $i++) {
    for ($j=0; $j <$pc ; $j++) {
        $energi+= ($my|$i|$j)*($my|$i|$j);
        $kontras+=((($i-$j)*($i-$j))*$my|$i|$j);
        $homo+=($my|$i|$j)/(1+((($i-$j)*($i-$j)));
        $entropi+=($my|$i|$j)*0.301*$my|$i|$j;
    }
}
?>
```

Source code diatas merupakan tahap pencarian rata-rata setiap fitur *GLCM* yang digunakan.

c. Klasifikasi metode *K-Means*

1) Penentuan *centeroid*

Penentuan *centeroid* pada penelitian ini dilakukan dengan mengambil nilai rata-rata dari setiap data sampel latihan yang ada dengan penyakit atau hama yang sama.

Beranda		Data Penyakit	Data Sampel Penyakit	Klasifikasi Sampel			
No Sampel	Gambar	Nama Penyakit	Ciri Faktor				
			Energi	Entropi	Homogeniti	Kontras	
22		Penyakit Daun - Glorosporium piperantum	0.41221764	0.12407751	0.93585564	0.31672935	
23		Penyakit Daun - Glorosporium piperantum	0.31736948	0.09552821	0.92293432	0.35920138	
24		Penyakit Daun - Glorosporium piperantum	0.46041932	0.13858621	0.92658341	0.34919973	
25		Penyakit Daun - Glorosporium piperantum	0.37878840	0.11401531	0.94549869	0.22031582	

Gambar 4.23. sampel data latih

Tabel 4.29. Tabel data sampel latih

no	No sampel	Nama penyakit	Energi	entropi	homogeni ti	kontras
1	22	Glorosporium piperantum	0.412218	0.124078	0.935856	0.316729
2	23	Glorosporium piperantum	0.317369	0.095528	0.922934	0.359201
3	24	Glorosporium piperantum	0.460419	0.138586	0.926583	0.3492
4	25	Glorosporium piperantum	0.378788	0.114015	0.945499	0.220316
5	26	Glorosporium piperantum	0.614287	0.184901	0.94462	0.25599
6	27	Glorosporium piperantum	0.422356	0.127129	0.929819	0.382211
7	28	Glorosporium piperantum	0.381447	0.114815	0.926102	0.35025
8	29	Glorosporium piperantum	0.512352	0.154218	0.93281	0.31608
9	30	Cylindrocladium quinqeseptatum	0.336931	0.101416	0.927757	0.427745
10	31	Cylindrocladium quinqeseptatum	0.519217	0.156284	0.928216	0.381862
11	32	Cylindrocladium quinqeseptatum	0.25014	0.075292	0.925322	0.455506
12	33	Cylindrocladium quinqeseptatum	0.421118	0.126756	0.929386	0.400628
13	34	Cylindrocladium quinqeseptatum	0.368483	0.110913	0.92433	0.379082
14	35	Penyakit Busuk Akar	0.392023	0.117999	0.927311	0.305117
15	36	Penyakit Busuk Akar	0.391144	0.117734	0.920823	0.351689
16	37	Penyakit Busuk Akar	0.303675	0.091406	0.938985	0.269312
17	38	Penyakit Busuk Akar	0.285243	0.085858	0.937932	0.229614

Tabel 4.30. Tabel penentu centeroid ke-1

no	No sampel	Nama <i>centeroid</i> ke-1	Energy	entropi	homogeniti	kontras
1	22	Glorosporium piperantum	0.41222	0.1241	0.935856	0.3167
2	23		0.31737	0.0955	0.922934	0.3592
3	24		0.46042	0.1386	0.926583	0.3492
4	25		0.37879	0.114	0.945499	0.2203
5	26		0.61429	0.1849	0.94462	0.256
6	27		0.42236	0.1271	0.929819	0.3822
7	28		0.38145	0.1148	0.926102	0.3503
8	29		0.51235	0.1542	0.93281	0.3161
Total			3.4992	1.0532	7.4642	2.5499
Rata-rata			0.4374	0.1316	0.933	0.31876

Tabel 4.31. Tabel penentu centeroid ke-2

no	No sampel	Nama <i>centeroid</i> ke-2	Energy	entropi	homogeniti	kontras
1	30	Cylindrocladium quinqeseptatum	0.33693	0.1014	0.927757	0.4277
2	31		0.51922	0.1563	0.928216	0.3819
3	32		0.25014	0.0753	0.925322	0.4555
4	33		0.42112	0.1268	0.929386	0.4006
5	34		0.36848	0.1109	0.92433	0.3791
Total			1.8958	0.5706	4.635	2.0448
Rata-rata			0.3791	0.1141	0.927	0.4089

Tabel 4.32. Tabel penentu centeroid ke-3

no	No sampel	Nama <i>centeroid</i> ke-3	Energy	entropi	homogeniti	kontras
1	35	Penyakit Busuk Akar	0.39202	0.118	0.927311	0.3051
2	36		0.39114	0.1177	0.920823	0.3517
3	37		0.30367	0.0914	0.938985	0.2693
4	38		0.28524	0.0859	0.937932	0.2296
Total			1.372	0.4129	3.725	1.1557
Rata-rata			0.3430	0.1032	0.9312	0.2889

Penentuan centeroid seperti ini dapat menghindari daftar penyakit yang ada pada sistem namun belum terdapat sampel latih yang di identifikasikan.

Tabel 4.33. Tabel kluster penyakit

klaster	kluster	energi	entropi	homogeniti	kontras
1	Glorosporium piperantum	0.4374	0.1316	0.933	0.31876
2	Cylindrocladium quinquesepatum	0.3791	0.1141	0.927	0.4089
3	Penyakit busuk akar	0.3430	0.1032	0.9312	0.2889

Implementasi pada sistem:

```

<?php
    $i=1 ;
    $ccc=0;
    $gh=0;
    $penyakit=$this->user_model->getdb('penyakit')->result();
    $sawal=$this->user_model->getdb('data_awal')->result();
    foreach ($penyakit as $p){
        $cn/$i/=0;
        $ce/$i/=0;
        $ch/$i/=0;
        $ck/$i/=0;
        $pn=0;
        foreach ($sawal as $a) {
            if ($p->id == $a->kode) {
                $ip/$i=$p->id;
                $cn/$i/+= $a->energi;
                $ce/$i/+= $a->entropi;
                $ch/$i/+= $a->homogeniti;
                $ck/$i/+= $a->kontras;
                $gh=1;
                $pn++;
            }
        }

        if ($pn>0) {
            $ccc++;
            $cn/$i/=$pn;
            $ce/$i/=$pn;
            $ch/$i/=$pn;
            $ck/$i/=$pn;
            $data3/"energi"=$cn/$i/;
            $data3/"entropi"=$ce/$i/;
            $data3/"homogeniti"=$ch/$i/;
        }
    }

```

```

$data3|"kontras"|=$ck/$i;
$data3|"id"|=$i;
$st=0;
$this->product_model->tambah($data3,'klas');
}
$i++;
}
$data3|"energi"|=$energi;
$data3|"entropi"|=$entropi;
$data3|"homogeniti"|=$homo;
$data3|"kontras"|=$kontras ;
$data3|"id"|=$i+1;
$this->product_model->tambah($data3,'klas');
?>

```

Source code diatas merupakan tahap penentuan centeroid dari rata-rata setiap nilai fitur GLCM penyakit yang tersimpan sebagai data latih maupun data uji.

2) Menghitung jarak data yang akan diklasifikasi dengan centeroid

Menentukan klasifikasi akan data uji terhadap centeroid yang dibuat dari rata-rata keseluruhan data sampel latih dapat mewakili setiap data yang ada dengan rumus berikut:

$$\sqrt{\sum (\quad)}, i=1,2,3 .. m$$

Data uji:

Detail Olah Citra:		
Citra Asli	Citra Abu-Abu	Nilai Fitur GLCM
		Energi :0.41221764749316 Entropi :0.12407751189544 Kontras :0.31672935153071 Homogeniti :0.93585564628639

Gambar 4.24. Data uji

Tabel 4.34. Data penyakit dengan nilai *GLCM*

no	No sampel	Nama penyakit	Energi	entropi	homogeni ti	kontras
1	22	Glorosporium piperantum	0.412218	0.124078	0.935856	0.316729
2	23	Glorosporium piperantum	0.317369	0.095528	0.922934	0.359201
3	24	Glorosporium piperantum	0.460419	0.138586	0.926583	0.3492
4	25	Glorosporium piperantum	0.378788	0.114015	0.945499	0.220316
5	26	Glorosporium piperantum	0.614287	0.184901	0.94462	0.25599
6	27	Glorosporium piperantum	0.422356	0.127129	0.929819	0.382211
7	28	Glorosporium piperantum	0.381447	0.114815	0.926102	0.35025
8	29	Glorosporium piperantum	0.512352	0.154218	0.93281	0.31608
9	30	Cylindrocladium quinqeseptatum	0.336931	0.101416	0.927757	0.427745
10	31	Cylindrocladium quinqeseptatum	0.519217	0.156284	0.928216	0.381862
11	32	Cylindrocladium quinqeseptatum	0.25014	0.075292	0.925322	0.455506
12	33	Cylindrocladium quinqeseptatum	0.421118	0.126756	0.929386	0.400628
13	34	Cylindrocladium quinqeseptatum	0.368483	0.110913	0.92433	0.379082
14	35	Penyakit Busuk Akar	0.392023	0.117999	0.927311	0.305117
15	36	Penyakit Busuk Akar	0.391144	0.117734	0.920823	0.351689
16	37	Penyakit Busuk Akar	0.303675	0.091406	0.938985	0.269312
17	38	Penyakit Busuk Akar	0.285243	0.085858	0.937932	0.229614
18	40	Data uji	0.412217	0.124077	0.316729	0.935855

$$\sqrt{\frac{0.412218 \times 0.124078 \times 0.935856 \times 0.316729}{0.412218 \times 0.124078 \times 0.935856 \times 0.316729}} = 0.0265$$

$$\sqrt{\frac{0.317369 \times 0.095528 \times 0.922934 \times 0.359201}{0.317369 \times 0.095528 \times 0.922934 \times 0.359201}} = 0.0988$$

$$\sqrt{\frac{0.460419 \times 0.138586 \times 0.926583 \times 0.3492}{0.460419 \times 0.138586 \times 0.926583 \times 0.3492}} = 0.0776$$

$$\sqrt{\frac{0.378788 \times 0.114015 \times 0.945499 \times 0.220316}{0.378788 \times 0.114015 \times 0.945499 \times 0.220316}}$$

$$\sqrt{\frac{0.614287 \times 0.184901 \times 0.94462 \times 0.25599}{0.614287 \times 0.184901 \times 0.94462 \times 0.25599}}$$

$$\sqrt{\frac{0.422356 \times 0.127129 \times 0.929819 \times 0.382211}{0.422356 \times 0.127129 \times 0.929819 \times 0.382211}}$$

$$\sqrt{\frac{0.381447 \times 0.114815 \times 0.926102 \times 0.35025}{0.381447 \times 0.114815 \times 0.926102 \times 0.35025}}$$

$$\sqrt{\frac{0.512352 \times 0.154218 \times 0.93281 \times 0.31608}{0.512352 \times 0.154218 \times 0.93281 \times 0.31608}}$$

$$\sqrt{\frac{0.336931 \times 0.101416 \times 0.927757 \times 0.427745}{0.336931 \times 0.101416 \times 0.927757 \times 0.427745}}$$

Tabel 4.35. Tabel nilai kedekatan iterasi pertama

no	No sampel	nilai kedekatan			status klaster
		klaster 1	klaster 2	klaster 3	
1	22	0.026514	0.098845	0.077609	1
2	23	0.132081	0.081499	0.075673	3
3	24	0.039325	0.10381	0.136736	1
4	25	0.116585	0.189489	0.07941	3
5	26	0.195461	0.289858	0.285543	1
6	27	0.065441	0.052546	0.124815	2
7	28	0.066721	0.058708	0.073504	2
8	29	0.078336	0.167376	0.178953	1
9	30	0.151362	0.047904	0.139032	2
10	31	0.106339	0.148811	0.206207	3
11	32	0.238737	0.142519	0.192857	2
12	33	0.08369	0.044719	0.138361	2
13	34	0.094294	0.031924	0.094281	2
14	35	0.049624	0.104658	0.053855	1
15	36	0.059703	0.058903	0.081112	2
16	37	0.148253	0.160725	0.04615	3
17	38	0.182251	0.204621	0.084834	3
18	40	0.872518	0.807037	0.895183	2

- 3) Setelah melakukan iterasi pertama, proses selanjutnya adalah memperbaharui nilai pusat klaster yang dapat dihitung dengan cara mencari nilai rata-rata sesuai jumlah anggota masing-masing klaster sesuai, kemudian kembali menghitung jarak data yang akan diklasifikasi dengan centeroid.

Tabel 4.36. Tabel centeroid baru

centeroid baru	energi	entropi	homogeniti	kontras
klaster 1	0.4782598	0.1439564	0.933436	0.3086232
klaster 2	0.3729795	0.1122665	0.8500335	0.4603708
klaster 3	0.3608584	0.1086182	0.9347132	0.292061

Tabel 4.37. Tabel nilai kedekatan iterasi ke dua

no	No sampel	nilai kedekatan			status klaster
		klaster 1	klaster 2	klaster 3	
1	22	0.069485	0.172272	0.059048	3
2	23	0.175783	0.137559	0.08191	3
3	24	0.045173	0.162964	0.118917	3
4	25	0.136875	0.258412	0.074928	3
5	26	0.151905	0.337969	0.267291	1
6	27	0.094003	0.123019	0.110795	1
7	28	0.109583	0.134131	0.062629	3
8	29	0.036381	0.221035	0.160032	1
9	30	0.189752	0.092318	0.138141	2
10	31	0.084974	0.188677	0.188297	1
11	32	0.279989	0.148824	0.200429	2
12	33	0.109738	0.111325	0.1256	1
13	34	0.134871	0.110226	0.087999	3
14	35	0.090335	0.17456	0.03584	3
15	36	0.101442	0.131083	0.068911	3
16	37	0.186595	0.222832	0.064047	3
17	38	0.216549	0.263382	0.100726	3
18	40	0.882329	0.715666	0.894009	2

Dari hasil perhitungan diatas didapatkan pusat klaster baru seperti pada table berikut:

Tabel 4.38. Pusat klaster kedua

centeroid baru	energi	entropi	homogeniti	kontras
klaster 1	0.497866	0.1498576	0.9329702	0.3473542
klaster 2	0.333096	0.1002617	0.7232693	0.6063687
klaster 3	0.3690809	0.1110932	0.9306355	0.313051

Tabel 4.39. Tabel iterasi ke tiga

no	No sampel	nilai kedekatan			status klaster
		klaster 1	klaster 2	klaster 3	
1	22	0.094585	0.368662	0.045499	3
2	23	0.189135	0.318163	0.071453	3
3	24	0.039668	0.353768	0.102087	1
4	25	0.178213	0.447995	0.094464	3
5	26	0.152529	0.507931	0.262726	1
6	27	0.086274	0.318745	0.088765	1
7	28	0.121807	0.330587	0.039637	3
8	29	0.034741	0.404003	0.149667	1
9	30	0.186377	0.271547	0.119542	3
10	31	0.041359	0.360813	0.171242	1
11	32	0.280506	0.266627	0.189078	2
12	33	0.096306	0.30539	0.103075	1
13	34	0.139061	0.305696	0.066334	3
14	35	0.118464	0.369016	0.025456	3
15	36	0.112196	0.32797	0.046044	3
16	37	0.217379	0.401353	0.081537	3
17	38	0.25138	0.436488	0.121163	3
18	40	0.856789	0.529777	0.875668	2

Dari hasil perhitungan diatas didapatkan pusat klaster baru seperti pada table berikut:

Tabel 4.40. Tabel kluster ke tiga

centeroid baru	energi	entropi	homogeniti	kontras
klaster 1	0.491625	0.147979	0.931906	0.347662
klaster 2	0.331179	0.099685	0.621026	0.695681
klaster 3	0.356732	0.107376	0.930753	0.320906

Tabel 4.41. Tabel iterasi ke 4

no	No sampel	nilai kedekatan			status kluster
		klaster 1	klaster 2	klaster 3	
1	22	0.007849	0.249885	0.003401	3
2	23	0.03333	0.204575	0.003218	3
3	24	0.001093	0.231631	0.012543	1
4	25	0.030287	0.333726	0.010866	3
5	26	0.024975	0.385453	0.076751	1
6	27	0.006431	0.202683	0.008456	1
7	28	0.013279	0.21515	0.001549	3
8	29	0.001467	0.277104	0.026439	1
9	30	0.032529	0.16591	0.011851	3
10	31	0.002014	0.23141	0.032515	1
11	32	0.075272	0.157442	0.030538	2
12	33	0.008233	0.190964	0.010879	1
13	34	0.017582	0.193746	0.003576	3
14	35	0.01265	0.250388	0.001619	3
15	36	0.01115	0.21213	0.002338	3
16	37	0.044714	0.283713	0.0058	3
17	38	0.060424	0.319949	0.013959	3
18	40	0.73129	0.157442	0.758546	2

Dikarenakan status kluster pada iterasi 3 dan 4 tidak mengalami perubahan, maka proses iterasi dihentikan dan data uji yang memiliki no sampel 40 memiliki status kluster 2 yaitu kluster untuk jenis penyakit *Cylindrocladium quinqeseptatum*, maka di

simpulkan data uji yang ada memiliki klasifikasi penyakit
Cylindrocladium quinqeseptatum.

Menghasilkan:



Gambar 4.25. Hasil klasifikasi sampel uji

Implementasi pada sistem:

```
<?php
$stat=0;
while ($stat ==0) {
    $awal=$this->user_model->getdb('klas')->result();
    $center=$this->user_model->getdb('center')->result();
    $center_lama=$this->user_model->getdb('center_lama')->result();
    $no=1;
    $sd=1;
    $ceks/1/="energi";
    $ceks/2/="entropi";
    $ceks/3/="homogeniti";
    $ceks/4/="kontras";
    $cl=$ccc;
    $x=1;
    $tr=1;
    $tr1=1;
    $stat=0;
    if ($center_lama != null) {
        foreach ($center_lama as $ct) {
            for ($i=1; $i <=count($ceks) ; $i++) {
                $c_lama|$tr|$i/=$ct->{$ceks/$i/};
            }
            $tr++;
        }
    }
    foreach ($center as $ct) {
        for ($i=1; $i <=count($ceks) ; $i++) {
            $c|$tr1|$i/=$ct->{$ceks/$i/};
        }
    }
}
```

```

    $str1++;
}
for ($p=1; $p <=$cl ; $p++) {
    for ($i=1; $i <=count($ceks) ; $i++) {
        if ($c_lama|$p||$i| == $c|$p||$i| ) {
            $stat=1;
        }
    }
}
}
}
}
$center_lama=$this->user_model->getdb('center_lama')->result();
foreach ($awal as $s) {
    $id/$x/=$s->id;
    $x++;
}
if ($center == null) {
    foreach ($awal as $ss) {
        for ($i=1; $i <=count($ceks) ; $i++) {
            $c/$no||$i/=$ss->{$ceks|$i|};
        }
        $no++;
    }
}
else{
    foreach ($center as $cs) {
        for ($i=1; $i <=count($ceks) ; $i++) {
            $c/$no||$i/=$cs->{$ceks|$i|};
        }
        $no++;
    }
}
for ($i=1; $i <=$cl ; $i++) {
    for ($j=1; $j <=count($ceks) ; $j++) {
        $data2/"$ceks|$j|"/=$c|$i||$j|;
        $data2/'id'/$i;
        $where/'id'/$i;
    }
    if ($center_lama == null) {
        $this->product_model->tambah($data2,'center_lama');
    }
    else
    {
        $this->product_model->ubah($where,$data2,'center_lama');
    }
}
}
    $no=1;
    foreach ($awal as $ss) {
        for ($i=1; $i <=count($ceks) ; $i++) {
            $data/$no||$i/=$ss->{$ceks|$i|};
        }
    }
}

```

```

        $no++;
    }
    for ($i=1; $i <=count($awal) ; $i++) {
        for ($j=1; $j <=$cl ; $j++) {
            $dc=0;
            for ($k=1; $k <=count($ceks) ; $k++) {
                $dc+=((($data/$i/$k/-$c/$j/$k))*($data/$i/$k/-$c/$j/$k));
            }
            $dc1/$i/$j=sqrt($dc);
        }
    }
    $i=1;
    for ($f=1; $f <=count($awal) ; $f++) {
        $cek= $dc1/$f/$i;
        $cc=1;
        $ips/$f=$ip/1;
        for ($o=2; $o <=$cl ; $o++) {
            if ($dc1/$f/$o < $cek)
            {
                $cek=$dc1/$f/$o;
                $ips/$f=$ip/$o;
                $cc=$o;
            }
        }
        $data=array(
            'status'=>$cc
        );
        $where=array(
            'id'=>$id/$f
        );
        $this->product_model->ubah($where,$data,'klas');
    }
    $awal1=$this->user_model->getdb('klas')->result();
    for ($r=1; $r <=$cl ; $r++) {
        for ($i=1; $i <=count($ceks) ; $i++) {
            $aa=0;
            $cek=0;
            foreach ($awal1 as $as) {
                if ($as->status == $r){
                    $aa+=$as->{$ceks/$i};
                    $cek++;
                }
            }
            $aa1/$r/$i=$aa/$cek;
        }
    }
    for ($u=1; $u <=$cl ; $u++) {
        for ($i=1; $i <=count($ceks) ; $i++) {
            $data1 ['id']=$u;
            $data1 ["$ceks/$i"]=$aa1/$u/$i;
        }
    }

```

```

    $where/'id'!= $u;
  }
  if ($center == null)
  {
    $this->product_model->tambah($data1,'center');
  }
  else
  {
    $this->product_model->ubah($where,$data1,'center');}
  }
}
for ($f=1; $f <=count($awal) ; $f++) {
  $data=array(
    'status'=>$ips/$f/
  );
  $where=array(
    'id'=>$id/$f/
  );
  $this->product_model->ubah($where,$data,'klas');
}
$this->product_model->hapus_all('center');
$this->product_model->hapus_all('center_lama');



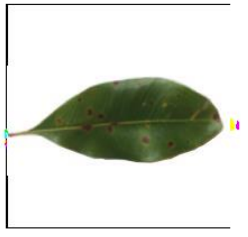
```

?>

Source code diatas merupakan proses dimana algoritma k-means berjalan dari awal hingga akhir.

Pengujian tingkat akurasi berdasarkan jarak ambil gambar sampel untuk memperoleh hasil diagnosa jenis penyakit atau hama yang dihasilkan pada sistem dilakukan dengan menguji tiga tingkatan jarak gambar yang dimaksud dan diuji dengan melakukan diagnosa terhadap 10 contoh gambar sampel dengan penilaian oleh pakar sebagai berikut:

Tabel 4.42. Tabel pengujian gambar sampel

No	Gambar	Jarak pengambilan gambar	Hasil klasifikasi	Keterangan
1		Focus dengan objek (Jarak antara 5-7 cm)	Penyakit Bercak Daun Antraknose (Colletotrichum Gloeosporioides)	Sesuai
Screenshot: 				
2		Jarak menengah (antara 7-10 cm)	Penyakit Bercak Daun Cephaleuros (Cephaleuros Virescens)	Tidak Sesuai

Screenshot:



3






Focus dengan objek (Jarak antara 5-7 cm)

Penyakit Bercak Daun Antraknose (Colletotrichum Gloeosporioide s)

Sesuai

Screenshot:



4		<p>Jarak menengah (antara 7-10 cm)</p>	<p>Penyakit Bercak Daun Antraknose (Colletotrichum Gloeosporioide s)</p>	<p>Sesuai</p>
<p>Screenshot:</p> 				
5		<p>Focus dengan objek (Jarak antara 5-7 cm)</p>	<p>Penyakit Bercak Daun Cephaleuros (Cephaleuros Virescens)</p>	<p>sesuai</p>

Screenshot:

6






Jarak
menengah
(antara 7-
10 cm)

Penyakit
Bercak Daun
Antraknose
(Colletotrichum
Gloeosporioide
s)

Tidak
Sesuai

Screenshot:


7		<p>Focus dengan objek (Jarak antara 5-7 cm)</p>	<p>Penyakit Bercak Daun Cephaleuros (Cephaleuros Virescens)</p>	<p>Sesuai</p>
<p>Screenshot:</p> 				
8		<p>Jarak menengah (antara 7-10 cm)</p>	<p>Penyakit Bercak Daun Antraknose (Colletotrichum Gloeosporioide s)</p>	<p>Tidak Sesuai</p>
<p>Screenshot:</p>				

Beranda Data Penyakit Sampel Latih Sampel Uji Klasifikasi Sampel Admin

Klasifikasi Sampel

Penyakit Bercak Daun Cephaleuros (*Cephaleuros Virescens*)

Gambar Sampel:



Sampel d4.png


Penyakit Bercak Daun Coniella (*Coniella Castaneicola*)

Gambar Sampel:

Penyakit Bercak Daun Pestalotia (*Pestalotia Versicolor*)

Gambar Sampel:

Penyakit Bercak Daun Sooty Mold (*Capnodium Sp. Dan Limacinula Samoensis*)

9		<p>Focus dengan objek (Jarak antara 5-7 cm)</p>	<p>Penyakit Bercak Daun Coniella (<i>Coniella Castaneicola</i>)</p>	<p>Sesuai</p>
---	--	---	---	---------------


Screenshot:

Beranda Data Penyakit Sampel Latih Sampel Uji Klasifikasi Sampel Admin

Klasifikasi Sampel

Penyakit Bercak Daun Coniella (*Coniella Castaneicola*)

Gambar Sampel:



Sampel d3.png


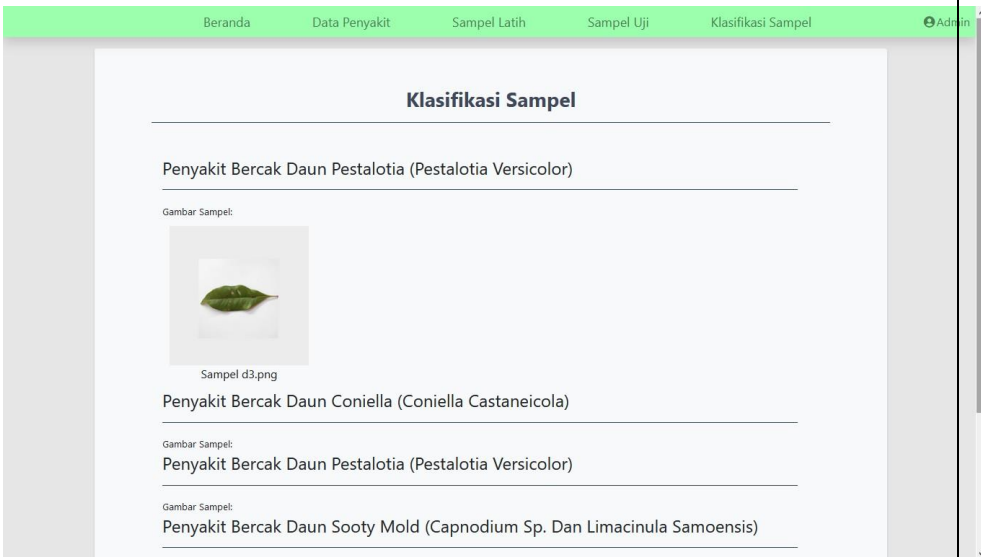
Penyakit Bercak Daun Coniella (*Coniella Castaneicola*)

Gambar Sampel:

Penyakit Bercak Daun Pestalotia (*Pestalotia Versicolor*)

Gambar Sampel:

Penyakit Bercak Daun Sooty Mold (*Capnodium Sp. Dan Limacinula Samoensis*)

10		Jarak menengah (antara 7-10 cm)	Penyakit Bercak Daun Pestalotia (Pestalotia Versicolor)	Tidak Sesuai
<p>Screenshot:</p> 				

Berdasarkan hasil pengujian terhadap jarak pengambilan gambar sampel yang dilakukan, disimpulkan jarak pengambilan gambar yang dianjurkan untuk memperoleh diagnosa yang lebih akruat adalah dengan mengambil gambar sampel yang terfokus dengan objek sampel secara langsung dengan jarak pengambilan gambar sektiar 5-7 CM seperti berikut:



Gambar 4.26. Contoh pengambilan gambar sampel yang dianjurkan

d. Tampilan Aplikasi

Screenshot of the application login page. The page has a dark blue background with green text and buttons. On the left, there is a 'LOGIN ADMIN' section with input fields for 'Nama Pengguna' and 'Kata Sandi', a 'Masuk' button, and a 'lihat password' checkbox. On the right, there is a 'Klasifikasi Penyakit tanaman Cengkeh' section with a 'Lebih Lanjut →' button. An illustration of three green trees is on the right side.

Gambar 4.27. Tampilan *form login*

Pada menu ini, admin dan pengguna biasa akan melakukan login untuk masuk ke dalam system untuk mengelola data penyakit, data sampel penyakit dan proses klasifikasi penyakit dari sampel yang akan diuji.



Gambar 4.28. tampilan halaman utama

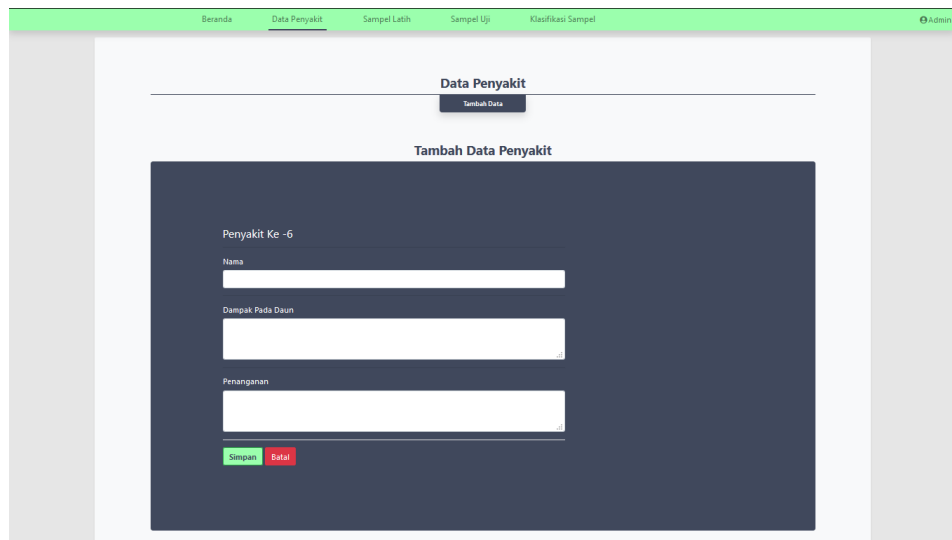
Pada menu halaman utama terdapat beberapa *direct link* yang berguna memandu pengguna diawal penggunaannya. Terdapat 4 menu bar yang terletak dibagian atas yaitu beranda, data penyakit, data sampel penyakit dan klasifikasi sampel .



Gambar 4.29. Tampilan menu data penyakit

Pada halaman ini, admin dan pengguna biasa dapat melihat daftar penyakit & hama apa saja yang terdaftar dalam sistem ini dan ciri-ciri

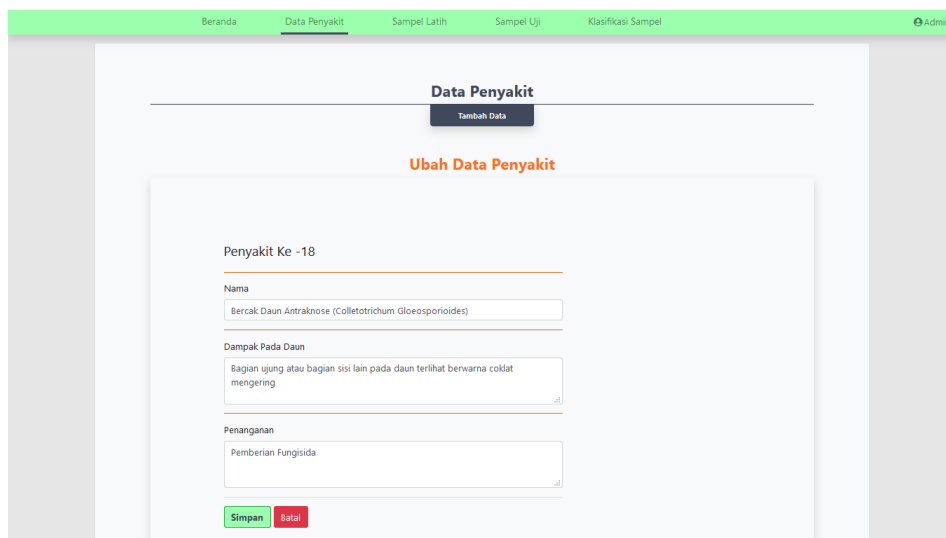
beserta dampaknya, admin dapat melakukan penambahan data tersebut, mengubahnya dan menghapusnya.



The screenshot shows a web application interface with a green navigation bar at the top containing the menu items: Beranda, Data Penyakit, Sampel Latih, Sampel Uji, and Klasifikasi Sampel. The user is logged in as 'Admin'. The main content area is titled 'Data Penyakit' and features a 'Tambah Data' button. Below this, a modal form titled 'Tambah Data Penyakit' is displayed. The form includes the following fields: 'Penyakit Ke -6', 'Nama' (with an empty input field), 'Dampak Pada Daun' (with an empty input field), and 'Penanganan' (with an empty input field). At the bottom of the form are two buttons: 'Simpan' (green) and 'Batal' (red).

Gambar 4.30. Tampilan menu tambah data penyakit

Pada halaman ini, admin dapat menambahkan penyakit dari nama, dampak pada daun dan cara penanganannya.



The screenshot shows the same web application interface as Gambar 4.30, but with the 'Ubah Data Penyakit' modal form open. The form is titled 'Ubah Data Penyakit' and includes the following fields: 'Penyakit Ke -18', 'Nama' (with the value 'Bercak Daun Antraknose (Colletotrichum Gloeosporioides)'), 'Dampak Pada Daun' (with the value 'Bagian ujung atau bagian sisi lain pada daun terlihat berwarna coklat mengering'), and 'Penanganan' (with the value 'Pemberian Fungisida'). At the bottom of the form are two buttons: 'Simpan' (green) and 'Batal' (red).

Gambar 4.31. Tampilan ubah data penyakit

Pada halaman ini, admin dapat mengubah penyakit dari nama, dampak pada daun dan cara menanganannya

The screenshot shows a web application interface for training data. At the top, there are navigation tabs: Beranda, Data Penyakit, **Sampel Latih**, Sampel Uji, and Klasifikasi Sampel. The main content area is titled 'Data Sampel Latih' and includes a 'Tambah Data' button. Below this, there is a search bar and a table with the following columns: No Sampel, Gambar, Nama Penyakit, Energi, Entropi, Homogeniti, Kontras, and Aksi. The table contains three rows of data, each with a sample image and associated GLCM feature values.

No Sampel	Gambar	Nama Penyakit	Energi	Entropi	Homogeniti	Kontras	Aksi
40		Bercak Daun Antraknose (Colletotrichum Gloeosporioides)	0,5834	0,1756	0,9398	0,5324	[Ubah] [Hapus]
41		Bercak Cephaluros (Cephaluros Virescens)	0,3826	0,1152	0,9365	0,2888	[Ubah] [Hapus]
42		Bercak Cephaluros (Cephaluros Virescens)	0,4143	0,1247	0,9268	0,4796	[Ubah] [Hapus]

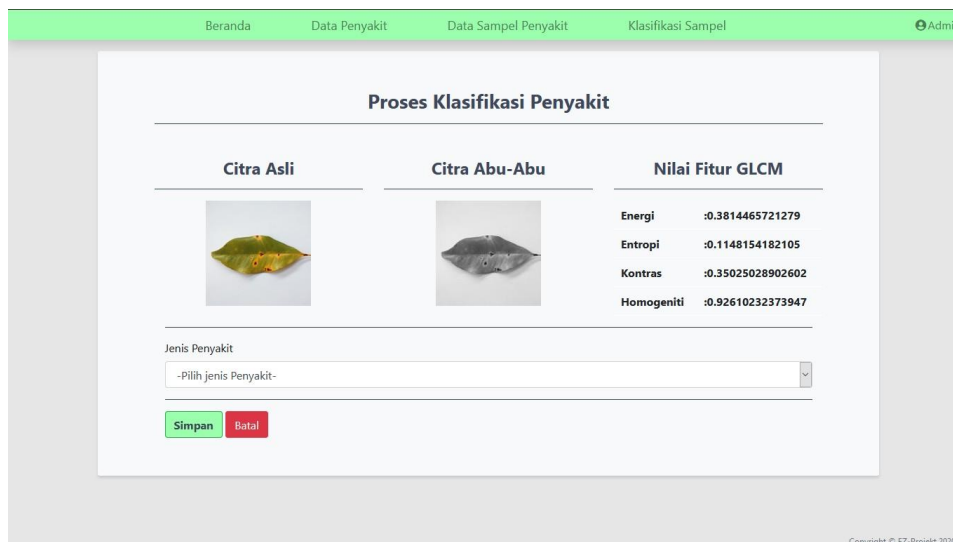
Gambar 4.32. Tampilan menu data sampel penyakit

Pada halaman ini, admin dan pengguna biasa dapat melihat daftar sampel latt yang telah memiliki nama penyakit dan telah memiliki ekstrak ciri fitur metode *GLCM*.

The screenshot shows the 'Tambah Data Sampel Uji' page. It features a 'Tambah Data' button at the top. Below it, the page is titled 'Tambah Data Sampel Uji' and 'Sampel Uji Ke-13'. There is a 'Pilih Gambar Sampel:' section with a file selection button labeled 'Telusuri...' and a message 'Tidak ada berkas dipilih.'. Below this are 'Proses' and 'Batal' buttons. To the right of the file selection is a 'Preview:' area with a large empty box. The page also includes navigation tabs at the top: Beranda, Data Penyakit, **Sampel Latih**, Sampel Uji, and Klasifikasi Sampel.

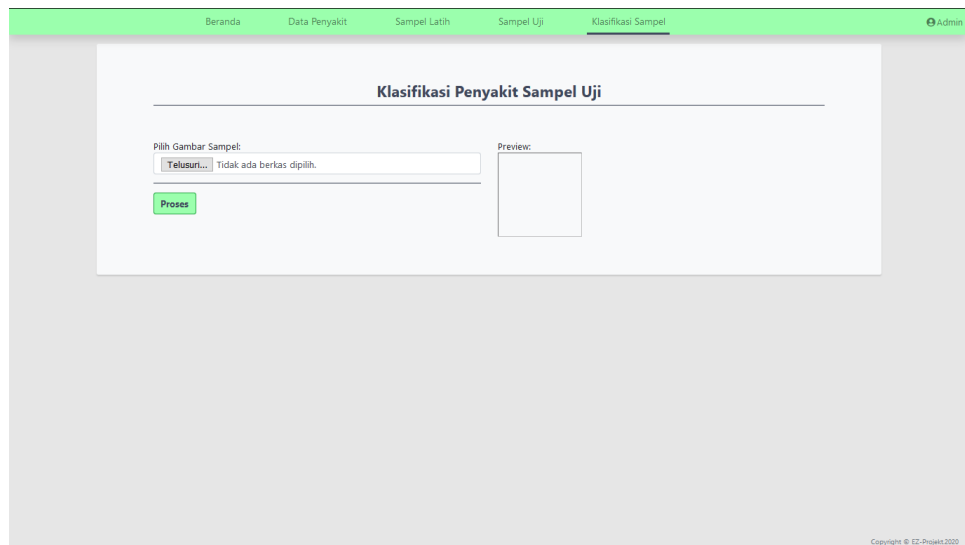
Gambar 4.33. Tampilan tambah data sampel latih

Pada halaman ini, admin dapat memilih gambar yang akan dijadikan sampel latih.



Gambar 4.34. Tampilan proses olah citra sampel latih baru

Jika telah memilih gambar yang akan di tambahkan ke sampel latih kemudian telah menekan tombol proses, admin akan dialihkan ke halaman ini dimana gambar yang ingin di proses akan diperoleh ekstraksi fitur citranya dan akan dipilhkan jenis penyakitnya dan kemudian dapat disimpan kedalam sampel latih.



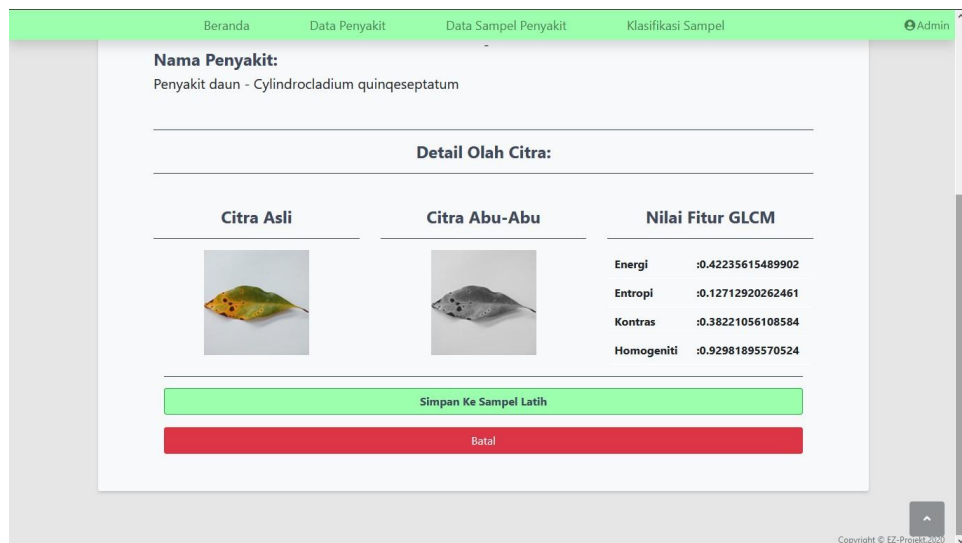
Gambar 4.35. Tampilan menu klasifikasi

Pada halaman ini, admin dan pengguna biasa dapat melakukan klasifikasi dari gambar daun yang memiliki dampak penyakit atau hama kemudian mengunggahnya dan menekan tombol proses.



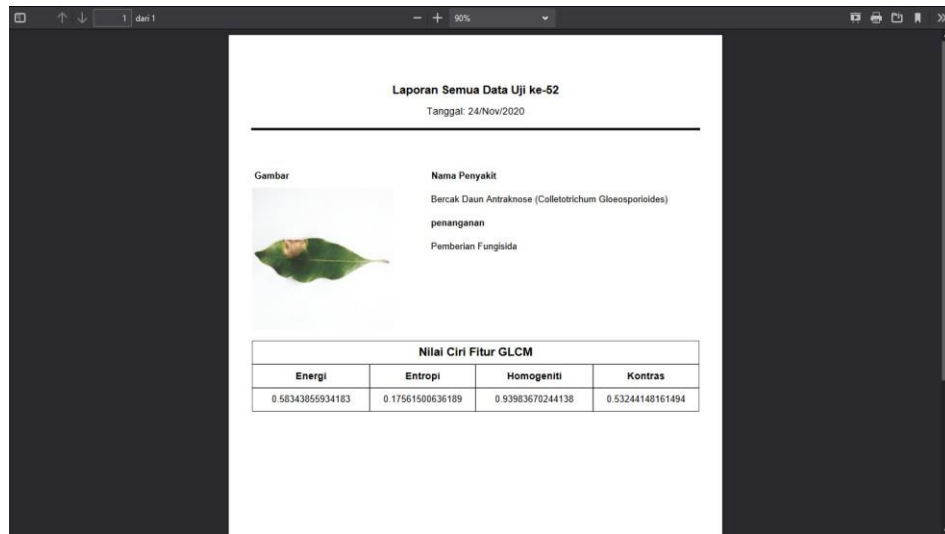
Gambar 4.36. tampilan proses olah citra sampel uji baru

Setelah menekan tombol proses maka admin ataupun pengguna biasa akan dialihkan kehalaman ini, dimana terdapat hasil dari klasifikasi yang di gunakan beserta ciri-ciri fisiknya dan penanganannya , jika tersedia.



Gambar 4.37. tampilan detail proses olah citra

Pada bagian bawah dari halaman ini terdapat detail dari nilai ekstraksi ciri fitur yang didapat dari gambar sampel, kemudia para pengguna baik admin ataupun pengguna biasa dapat menyimpan hasil uji sampel tersebut kedalam kumpulan sampel latih untuk meningkatkan tingkat akurasi dari data latih yang menjadi acuan klasifikasi penyakit.



Gambar 4.38. tampilan *report* hasil proses klasifikasi

e. Pengujian Sistem

Pengujian sistem merupakan bagian yang penting dalam membangun sebuah perangkat lunak. Tujuan pengujian adalah kualitas dan mampu mempresentasikan kajian pokok dari spesifikasi, analisis dan perancangan perangkat lunak itu sendiri. Pengujian yang dilakukan menggunakan pengujian *blackbox*.

1. Pengujian *Blackbox*

Sebuah program diterapkan di instansi, maka program harus bebas dari kesalahan-kesalahan atau *error*. Oleh karena itu, program harus diuji coba terlebih dahulu untuk menemukan kesalahan-kesalahan yang mungkin terjadi, pengujian ini menggunakan metode pengujian *black box*.

Tabel 4.43. Tabel Rencana Pengujian

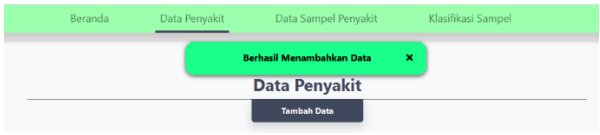
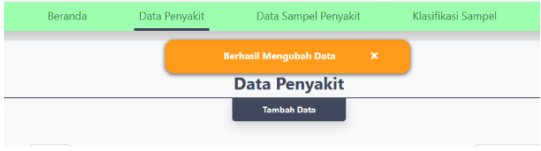
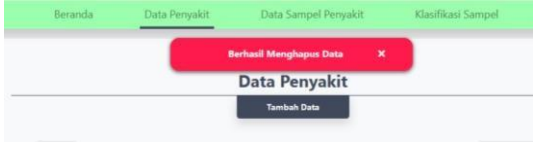
Menu yang diuji	Detail Pengujian	Jenis Pengujian
<i>Login</i>	- Validasi login	<i>Black Box</i>
Menu Data penyakit	-Tambah -Edit -Hapus	<i>Black Box</i>
Menu Data sampel penyakit	-Tambah -Edit -Hapus	<i>Black Box</i>
Menu Proses klasifikasi dan olah citra	- Proses olah citra <i>GLCM</i> - Hasil klasifikasi dengan algoritma <i>K-Means</i>	<i>Black Box</i>
Logout	Validasi logout	<i>Black Box</i>

Berdasarkan rencana pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut :

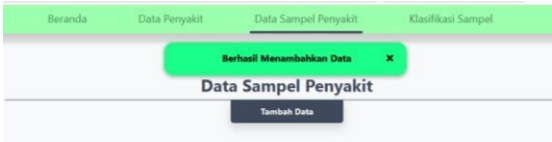

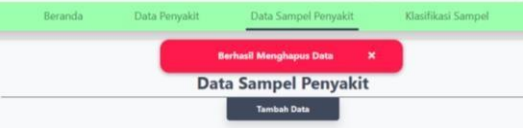
Tabel 4.44. Hasil Pengujian Login

Kasus Dan Hasil Uji			
Kasus	Skenario Uji	Hasil yang diharapkan	Hasil
Login	Menginput username dan <i>Password</i> yang valid	Berhasil masuk ke sistem	Sukses
	Screen shot:		
	Menginput <i>username</i> dan <i>Password</i> yang tidak valid	Menampilkan pesan gagal masuk ke system	Sukses
	Screen shot:		







Tabel 4.45. Hasil Pengujian menu data penyakit

Kasus Dan Hasil Uji			
Kasus	Skenario Uji	Hasil yang diharapkan	Hasil
Menu Data Penyakit	Tambah data penyakit	Berhasil Menambahkan data penyakit	sukses
	Screenshot: 		
	Edit data penyakit	Berhasil Mengubah data penyakit	Sukses
	Screenshot: 		
	Hapus Data penyakit	Berhasil Menghapus penyakit	sukses
Screenshot: 			

Tabel 4.46. Hasil Pengujian menu data sampel latihan

Kasus Dan Hasil Uji			
Kasus	Skenario Uji	Hasil yang diharapkan	Hasil
Menu Data Sampel Penyakit	Tambah data sampel penyakit	Berhasil Menambahkan data sampel penyakit	sukses
	Screenshot:		
	Edit data sampel penyakit	Berhasil Mengubah sampel data penyakit	Sukses
	Screenshot:		
	Hapus Data sampel penyakit	Berhasil Menghapus data sampel penyakit	sukses
Screenshot:			

Tabel 4.47. Hasil Pengujian proses olah citra *GLCM*

Kasus Dan Hasil Uji																
Kasus	Skenario Uji	Hasil yang diharapkan	Hasil													
Olah citra <i>GLCM</i>	Melakukan ekstraksi ciri fitur <i>GLCM</i>	Mendapatkan nilai fitur energi, entropi, kontras dan homogeniti	sukses													
	Screenshot: <div style="border: 1px solid #ccc; padding: 10px; margin-top: 5px;"> <p style="text-align: center; margin: 0;">Detail Olah Citra:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 33%; text-align: center; border-bottom: 1px solid #ccc;">Citra Asli</th> <th style="width: 33%; text-align: center; border-bottom: 1px solid #ccc;">Citra Abu-Abu</th> <th style="width: 33%; text-align: center; border-bottom: 1px solid #ccc;">Nilai Fitur GLCM</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;"></td> <td style="text-align: center;"></td> <td> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Energi</td> <td style="padding: 2px;">:0.41221764749316</td> </tr> <tr> <td style="padding: 2px;">Entropi</td> <td style="padding: 2px;">:0.12407751189544</td> </tr> <tr> <td style="padding: 2px;">Kontras</td> <td style="padding: 2px;">:0.31672935153071</td> </tr> <tr> <td style="padding: 2px;">Homogeniti</td> <td style="padding: 2px;">:0.93585564628639</td> </tr> </table> </td> </tr> </tbody> </table> </div>			Citra Asli	Citra Abu-Abu	Nilai Fitur GLCM			<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Energi</td> <td style="padding: 2px;">:0.41221764749316</td> </tr> <tr> <td style="padding: 2px;">Entropi</td> <td style="padding: 2px;">:0.12407751189544</td> </tr> <tr> <td style="padding: 2px;">Kontras</td> <td style="padding: 2px;">:0.31672935153071</td> </tr> <tr> <td style="padding: 2px;">Homogeniti</td> <td style="padding: 2px;">:0.93585564628639</td> </tr> </table>	Energi	:0.41221764749316	Entropi	:0.12407751189544	Kontras	:0.31672935153071	Homogeniti
Citra Asli	Citra Abu-Abu	Nilai Fitur GLCM														
		<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Energi</td> <td style="padding: 2px;">:0.41221764749316</td> </tr> <tr> <td style="padding: 2px;">Entropi</td> <td style="padding: 2px;">:0.12407751189544</td> </tr> <tr> <td style="padding: 2px;">Kontras</td> <td style="padding: 2px;">:0.31672935153071</td> </tr> <tr> <td style="padding: 2px;">Homogeniti</td> <td style="padding: 2px;">:0.93585564628639</td> </tr> </table>	Energi	:0.41221764749316	Entropi	:0.12407751189544	Kontras	:0.31672935153071	Homogeniti	:0.93585564628639						
Energi	:0.41221764749316															
Entropi	:0.12407751189544															
Kontras	:0.31672935153071															
Homogeniti	:0.93585564628639															

Tabel 4.48. Hasil Pengujian proses klasifikasi *K-Means*

Kasus Dan Hasil Uji				
Kasus	Skenario Uji	Hasil yang diharapkan	Hasil	
Klasifikasi penyakit dengan algoritma <i>K-Means</i>	Proses klasifikasi dari nilai fitur <i>GLCM</i> yang didapat	Mendapatkan jenis penyakit	sukses	
	Screenshot: <div style="border: 1px solid #ccc; padding: 10px; margin-top: 5px;"> <p style="text-align: center; margin: 0;">Proses Klasifikasi Penyakit</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; vertical-align: top; padding: 5px;"> <p>Gambar Sampel Uji:</p>  <p>Nama Penyakit: Penyakit daun - <i>Cylindrocladium quinqueseptatum</i></p> </td> <td style="width: 70%; vertical-align: top; padding: 5px;"> <p>Ciri-Ciri Fisik: ditandai bercak-bercak merah atau merah kecoklatan yang biasanya dibagian tengahnya berwarna putih</p> <p>Penanganan: -</p> </td> </tr> </table> </div>			<p>Gambar Sampel Uji:</p>  <p>Nama Penyakit: Penyakit daun - <i>Cylindrocladium quinqueseptatum</i></p>
<p>Gambar Sampel Uji:</p>  <p>Nama Penyakit: Penyakit daun - <i>Cylindrocladium quinqueseptatum</i></p>	<p>Ciri-Ciri Fisik: ditandai bercak-bercak merah atau merah kecoklatan yang biasanya dibagian tengahnya berwarna putih</p> <p>Penanganan: -</p>			

BAB V

PENUTUP

A. Kesimpulan

Setelah melalui proses perancangan pengenalan pola citra dengan ekstraksi *GLCM* dan klasifikasi *K-Means* untuk mendeteksi penyakit pada daun cengkeh, adapun kesimpulan yang diberikan yaitu:

1. Sistem ini memberikan hasil berupa klasifikasi jenis penyakit pada daun cengkeh dengan menggunakan algoritma *K-Means* berdasarkan hasil olah citra *GLCM* dari gambar daun tanaman cengkeh yang terdampak.
2. Sistem ini dapat digunakan untuk membantu para petani/pengurus tanaman cengkeh dalam mendiagnosa penyakit apa yang menjangkit tanamannya hanya dengan mengunggah foto dari daun tanaman cengkeh yang terdampak.

B. Saran

Setelah melalui proses pengenalan pola citra dengan klasifikasi k-mens untuk mendeteksi penyakit pada daun cengkeh, adapun saran yang diberikan yaitu:

1. Sistem yang telah dibuat dikembangkan menjadi aplikasi *mobile* yang lebih *modern* lagi dan bisa digunakan secara maksimal di ponsel untuk membuat para petani/pengurus tanaman cengkeh dapat menggunakannya dengan lebih efisien.
2. Memperbanyak daftar penyakit/hama yang ada.
3. Perlunya tahap pengembangan dengan merubah konsep klasifikasi menjadi sistem pakar.

DAFTAR PUSTAKA

- Alvansga, E. (2019). *Texture Recognition Using Glcm Method*.
http://repository.usd.ac.id/35558/2/155114015_full.pdf
- Ardianto, I., & Nst, M. R. K. (2016). Perancangan aplikasi video streaming web menggunakan xampp di universitas al-washliyah labuhanbatu. *Infotek*, 1(3), 1–6.
- Ariyani, D. (2019). *Sistem Pakar Hama Dan Penyakit Tembakau*. 12(1).
- Bin Ladjamudin. (2013). Analisis dan Desain Sistem Informasi. *Analisis Dan Desain Sistem Informasi*, 53(9), 1689–1699.
- Firman, A., Wowor, H. F., Najoan, X., Teknik, J., Fakultas, E., & Unsrat, T. (2016). Sistem Informasi Perpustakaan Online Berbasis Web. *E-Journal Teknik Elektro Dan Komputer*, 5(2), 29–36.
- Ganis, Y. K., Santoso, I., & Isnanto, R. R. (2011). Klasifikasi Citra Dengan Matriks Ko-okurensi Aras Keabuan (Gray Level Co-occurrence Matrix - GLCM) Pada Lima Kelas Biji-Bijian. *Undergraduate Thesis, Jurusan Teknik Elektro Fakultas Teknik Univesitas Diponegoro*, 1–7.
- Iswari, L., & Ayu, E. G. (2015). Pemanfaatan Algoritma K-Means Untuk Pemetaan Hasil Klasterisasi Data Kecelakaan Lalu Lintas. *Teknoin*, 21(1).
<https://doi.org/10.20885/teknoin.vol21.iss1.art7>
- Johan Wahyudi, & Ihdahubbi Maulida. (2019). Pengenalan Pola Citra Kain

Tradisional Menggunakan Glcm Dan Knn. *Jurnal Teknologi Informasi Universitas Lambung Mangkurat (JTIULM)*, 4(2), 43–48. <https://doi.org/10.20527/jtiulm.v4i2.37>

Rahman, Y., & Wijayanto, H. (2015). Klasifikasi Batik Menggunakan Metode K-Nearest Neighbour Berdasarkan Gray Level Co-Occurrence Matrices (GLCM). *Jurusan Teknik Informatika FIK UDINUS*, 244(Ecpe), 1–7.

Setiawan, A., Hidayat, N., & Dewi, R. K. (2018). Sistem Pakar Diagnosis Penyakit Tanaman Cengkeh Menggunakan Metode Naive Bayes (Studi kasus Kecamatan Wonosalam , Jombang). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 2(10), 4034–4038.

Wahyuno, D., & Martini, E. (2015). Pedoman Budi daya Cengkeh di Kebun Campur. *Guidelines for Clove Cultivation in Mixed Gardens. Bogor, Indonesia: World Agroforestry Centre (ICRAF) Southeast Asia Regional Program*, 2017. <http://www.worldagroforestry.org/sea/Publications/files/booklet/BL0052-15.pdf>

Waliyansyah, R. R., Adi, K., & Suseno, J. E. (2018). Implementasi Metode Gray Level Co-occurrence Matrix dalam Identifikasi Jenis Daun Tengkawang. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi (JNTETI)*, 7(1), 50–56. <https://doi.org/10.22146/jnteti.v7i1.400>



PEMERINTAHAN KABUPATEN KEPULAUAN SELAYAR
DINAS PERTANIAN DAN KETAHANAN PANGAN

Jl. Dr. Shun Raiuiani No. 1? Benteng Sulawesi Selatan
Telpon (IH)4) 21058-21208 Faximile (tH4) 21055

Data produksi cengkeh kabupaten kepulauan selayar:

No	Tahun	Hasil Produksi
1	2013	126.00
2	2014	125,00
3	2015	125,50
4	2016	178,35
5	2017	102,49
6	20 18	535,09
7	2019	482,07

Bentong 19 Janiari 2021

Kepala Dinas,



Ir. Ismail

NIP. 19630526 199503 1 004

PENGUJIAN SISTEM

PENGENALAN POLA CITRA DENGAN EKSTRAKSI *GLCM* DAN KLASIFIKASI *K-MEANS* UNTUK MENDETEKSI PENYAKIT PADA DAUN CENGKEH

NAMA Odo Jamaluddin, SP

PROFESI Staf Kantor Dinas Pertanian dan Ketahanan

NO HP 085 756 923 774

HASIL PENGUJIAN kr'>if•\>i rt^{oo} /<<f ;*^*yur .ri gis|>t,I*@r<
berjalan penyakit dapat

SARAN Sebagai saran dari program yang telah saya
uji, mungkin kedepannya program tersebut
dapat ditakukan se
langung
memperm

Benteng, December 2020


(Odo Jamaluddin, Sp.)