

**Segmentasi K-means Citra Daun Tin Dengan Klasifikasi
Ekstraksi Ciri GLCM**



TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister
Ilmu Komputer (M.Kom)

Muhammad Qomaruddin

14002209

PROGRAM PASCASARJANA MAGISTER ILMU KOMPUTER
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
NUSA MANDIRI
JAKARTA
2020

SURAT PERNYATAAN ORISINILITAS DAN BEBAS PLAGIARISME

Yang bertanda tangan dibawah ini :

Nama : Muhammad Qomaruddin

Nim : 14002209

Program Studi : Magister Ilmu Komputer

Jenjang : Strata Dua (S2)

Konsentrasi : *Image Processing*

Dengan ini menyatakan bahwa tesis yang telah saya buat dengan judul "Segmentasi K-Means Citra Daun Tin Dengan Klasifikasi Ekstrasi Ciri GLCM" adalah hasil karya sendiri, dan semua sumber baik yang dikutip maupun yang dirujuk telah saya nyatakan dengan benar dan tesis belum pernah diterbitkan atau dipublikasikan dimanapun dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat dengan sebenar-benarnya. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa tesis saya yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari Program Pascasarjana Magister Ilmu Komputer Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri dicabut atau dibatalkan.

Jakarta, 6 Agustus 2020

Yang menyatakan,



Muhammad Qomaruddin

HALAMAN PERSETUJUAN DAN PENGESAHAN TESIS

Tesis ini diajukan oleh:

Nama : M. Qomaruddin
NIM : 14002209
Program Studi : Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : *Image Processing*
Judul Tesis : Segmentasi K-means Citra Daun Tin Dengan Klasifikasi Ekstraksi Ciri Glem

Telah dipertahankan pada periode 2020-1 dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri (STMIK Nusa Mandiri).

Jakarta, 28 Agustus 2020

PEMBIMBING TESIS

Pembimbing I : Dr. Dwiza Riana, S.Si, MM, M.Kom



Pembimbing II : Anton, M.Kom.



DEWAN PENGUJI

Penguji I : Dr. Agus Subekti, M.T



Penguji II : Dr. Windu Gata, M.Kom



Penguji III /
Pembimbing I : Dr. Dwiza Riana, S.Si, MM,
M.Kom



KATA PENGANTAR

Bismillahirromanirrohim, Alhamdulillahirobbil alamin, penulis panjatkan puji syukur kehadiran Allah SWT yang telah melimpahkan segala rahmat, hidayah dan nikmat sehat sehingga penulis dapat menyelesaikan penulisan tesis ini. Dimana tesis ini penulis sajikan dalam bentuk buku. Tesis ini berjudul Segmentasi K-means Citra Daun Tin Dengan Klasifikasi Ekstrasi Ciri GLCM.

Tujuan penulisan tesis ini dibuat sebagai salah satu syarat untuk mendapatkan gelar Magister Ilmu Komputer (M.Kom) pada Program Pascasarjana Magister Ilmu Komputer Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri (PPs MIK STMIK Nusa Mandiri).

Tesis ini diambil berdasarkan hasil penelitian mengenai segmentasi citra pada daun Tin. Dalam tahap penulisan tesis ini, penulis mencari sumber referensi dari jurnal ilmiah, buku dan internet yang berkaitan dengan pembahasan tesis ini.

Penulis menyadari bahwa tanpa bimbingan dan dukungan dari semua pihak dalam pembuatan tesis ini, maka penulis tidak dapat menyelesaikan tesis ini tepat pada waktunya. Untuk itu izinkanlah penulis dalam kesempatan ini untuk mengucapkan, terima kasih yang sebesar-besarnya kepada :

1. Ibu Dr. Dwiza Riana, S.Si, MM, M.Kom selaku Dosen Pembimbing tesis yang dengan kesabarannya membimbing penulis, menyediakan waktu, pikiran dan tenaga dalam membimbing penulis dalam menyelesaikan tesis ini.
2. Bapak Anton, M.Kom selaku Dosen Pembimbing tesis yang menyediakan waktu, pikiran dan tenaga dalam membimbing penulis dalam menyelesaikan tesis ini.
3. Orang tua tercinta Bapak Zaidun, doa untuk Ibu Alm Siti Asobah dan Istri tercinta Stephanie Dara Geraldine, anak Khaylilla Ashley Fatma, kakak Abdul Fatah S.pdi, Adik saya Muhammad Salafudin S.Kom, Muhmmad Lutfi Amin S.A yang telah memberikan dukungan material dan moral kepada penulis.
4. Seluruh staf pengajar (dosen) PPs MIK STMIK Nusa Mandiri yang telah memberikan pelajaran yang berarti bagi penulis selama menempuh studi.

5. Seluruh staf dan karyawan PPs MIK STMIK Nusa Mandiri yang telah melayani penulis dengan baik selama kuliah.
6. Rekan-rekan mahasiswa kelas 14.4A.01 yang telah membantu untuk memberikan masukan dan arahan serta dukungan untuk menyelesaikan tesis ini.

Serta semua pihak yang tidak dapat penulis sebutkan satu persatu sehingga terwujudnya penulisan tesis ini. Penulis menyadari bahwa penulisan tesis ini masih jauh sekali dari sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan karya ilmiah yang penulis hasilkan untuk yang akan datang.

Akhir kata semoga tesis ini dapat bermanfaat bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya

Jakarta, 6 Agustus 2020



Muhammad Qomaruddin
Penulis

**SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH
UNTUK KEPENTINGAN AKADEMIS**

Yang bertanda tangan dibawah ini :

Nama : Muhammad Qomaruddin
Nim : 14002209
Program Studi : Magister Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : *Image Processing*

Demi pengembangan ilmu pengetahuan, dengan ini menyetujui untuk memberikan ijin kepada pihak Program Studi Ilmu Komputer (S2) Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri (STMIK Nusa Mandiri) Hak Bebas Royalti Non-Eksklusif (Non-exclusive Royalti-Free Right) atas karya ilmiah saya yang berjudul : "Segmentasi K-Means Citra Daun Tin Dengan Klasifikasi Ekstraksi Ciri GLCM" beserta perangkat yang diperlukan (apabila ada). Dengan Hak Bebas Royalti Non-Eksklusif ini pihak STMIK Nusa Mandiri berhak menyimpan, mengalih-media atau bentuk-kan, mengelolanya dalam pangkalan data (*database*), mendistribusikannya dan menampilkan atau mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama kami sebagai penulis/pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak STMIK Nusa Mandiri, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini . Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 06 Agustus 2020

Yang menyatakan,



Muhammad Qomaruddin

ABSTRAK

Nama : Muhammad Qomaruddin
NIM : 14002209
Program Studi : Magister Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : Image Processing
Judul : “Segmentasi K-Means dan Klasifikasi Naive Bayes pada Citra Daun Tin Dengan Ekstrasi Ciri GLCM”

Tanaman Tin dengan nama latin *Ficus Caric* adalah sejenis tanaman buah dari sejenis pohon yang banyak tumbuh di kawasan daerah tropis dan subtropis. Tanaman Tin saat ini sudah banyak dibudidayakan di Indonesia. Tanaman Tin dapat tumbuh hingga 10 meter, dengan batang lunak berwarna abu-abu dimana daunnya cukup besar dan berlekuk dalam 3 atau 5 cuping. Buah Tin memiliki buah yang berwarna kuning kecoklatan, dengan rasa yang manis. *Cerotelium Fici* adalah jenis penyakit karat daun yang menyerang pada daun tin, dan menjadi ancaman terbesar terhadap produksi buah tin. Penyakit lain yang menyerang pada tanaman tin adalah kutu kebul dan virus mosaik. Virus mosaik ini pertama kali muncul di *California* dan menyebar ke sebagian besar wilayah Indonesia dan Amerika Serikat. Daun tin yang terinfeksi virus ini menjadi bintik-bintik cokelat menyebabkan pertumbuhan tanaman tin menjadi lambat dan cacat pada buah tin. Dalam perkembangan ilmu pengetahuan dan teknologi, cara untuk mendeteksi penyakit pada tanaman tin seperti karat daun, virus mosaik dan kutu kebul dapat dilakukan dengan bantuan pengolahan citra. Untuk itu penelitian ini bertujuan melakukan pengolahan citra berupa segmentasi ROI pada citra daun tin yang dianalisa dengan ekstrasi fitur GLCM dengan mengklasifikasikan Algoritma C4.5 dan *Naïve Bayes* untuk mendapatkan akurasi terbaik dalam klasifikasi penyakit citra daun tin . Setelah itu, dilakukan analisis tekstur menggunakan metode *Grey Level Co-Occurance Matrix* (GLCM) dan segmentasi k-means clustering dalam pengolahan citra daun tin.

Kata kunci : *K-Means*, GLCM, Algoritma *Naïve Bayes*, Segmentasi

ABSTRACT

Name : Muhmmad Qomaruddin
NIM : 14002209
Study of Program : Magister Ilmu Komputer
Levels : Program Study (S2)
Concentration : *Image Processing*
Titel : “Segmentasi K-Means dan Klasifikasi Naive Bayes pada Citra Daun Tin Dengan Ekstrasi Ciri GLCM”

The fig plant with the latin name Ficus Caric is a type of fruit plant from a type of tree that grows in many tropical and subtropical regions. Figs are now widely cultivated in Indonesia. Fig plants can grow up to 10 meters tall, with soft gray stems where the leaves are quite large and curved in 3 or 5 lobes. Fig has a brownish yellow fruit, with a sweet taste. Cerotelium Fici is a type of leaf rust that attacks the fig leaves, and is the biggest threat to the production of figs. Other diseases that attack the fig plant are whitefly and mosaic virus. This mosaic virus first appeared in California and spread to most parts of Indonesia and the United States. The fig leaves infected with this virus become brown spots causing the growth of fig plants to be slow and defective in figs. In the development of science and technology, ways to detect diseases in fig plants such as leaf rust, mosaic viruses and white lice can be done with the help of image processing. For this reason, this study aims to perform image processing in the form of ROI segmentation on fig leaf images analyzed by extracting the GLCM feature by classifying C4.5 and Naïve Bayes Algorithm to get the best accuracy in the classification of fig leaf image diseases. After that, texture analysis is done using the Gray Level Co-Occurance Matrix (GLCM) method and k-means clustring segmentation in fig leaf image processing.

Keywords: K-Means, GLCM, Algoritma Naïve Bayes, Segmentation

DAFTAR ISI

	Halaman
HALAMAN SAMPUL	i
HALAMAN JUDUL	ii
HALAMAN PERNYATAAN ORISINILITAS	iii
HALAMAN PENGESAHAN	iv
KATA PENGANTAR	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS	vii
ABSTRAK	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xiv
 BAB 1. PENDAHULUAN	
1.1. Latar Belakang.....	1
1.2. Identifikasi Masalah	2
1.3. Tujuan Penelitian.....	2
1.4. Ruang Lingkup Penelitian	3
1.5. Sistematika Penulisan	3
 BAB II. LANDASAN PEMIKIRAN	
2.1. Tinjauan Pustaka	5
2.1.1. Digitalisasi Citra.....	5
2.1.2. Resize	5
2.1.3. Konversi Citra	6
2.1.4. Segmentasi Citra	6
2.1.5. Segmentasi <i>K-Means</i>	9
2.1.6. <i>Region Of Interest</i>	9
2.1.7. Data Set Pelatihan dan Pengujian	9
2.1.8. Metode Analisis Tekstur GLCM.....	10
2.1.9. Algoritma <i>Naïve Bayes</i>	13
2.2. Tinjauan Studi.....	14
 BAB III. METODOLOGI PENELITIAN	
3.1. Digitalisasi Citra	21
3.2. <i>Resize</i>	22
3.3. <i>Bw Image</i>	22
3.4. <i>Thresholding</i>	23
3.5. Segmentasi <i>K-Means</i>	24
3.6. <i>Region Of Interest</i>	25
3.7. Data Set Pelatihan Dan Pengujian.....	26
3.8. Algoritma <i>Naïve Bayes</i> dan Algoritma C4.5	29
3.9. Ekstraksi Fitur GLCM	29

3.10. Evaluasi dan Validasi Hasil	29
3.11. Identifikasi Citra.....	30

BAB IV. HASIL PENELITIAN DAN PEMBAHASAN

4.1. Digitalisasi Citra	31
4.2. <i>Resize</i>	32
4.3. <i>Bw Image</i>	34
4.4. Segmentasi <i>Thresholding</i>	35
4.5. Segmentasi Algoritma <i>K-Means</i>	36
4.6. <i>Region Of Interest</i>	37
4.7. Ekstraksi Fitur <i>Gray Level Co-Occurrence Matrix</i>	39
4.8. Hasil Pelatihan Citra Daun Tin	39
4.9. Hasil Pengujian Citra Daun Tin.....	41
4.10. Hasil Komparasi Algoritma Naïve Bayes	43
4.11. Implementasi Dan Hasil	44

BAB V. PENUTUP

5.1. Kesimpulan	47
5.2. Saran.....	48

DAFTAR PUSTAKA

LEMBAR KONSULTASI

DAFTAR RIWAYAT HIDUP

LAMPIRAN-LAMPIRAN

DAFTAR TABEL

	Halaman
Tabel 2.1 Penelitian Terdahulu.....	17
Tabel 3.1 Contoh Citra Daun Tin Sakit	21
Tabel 3.2 Contoh Citra	Bw
<i>Image</i>	23
.....	
Tabel 3.2 Contoh Hasil <i>Thresolding</i>	24
Tabel 3.3 Contoh Citra Segmentasi K-Means.....	25
Tabel 3.4 Contoh Citra <i>Region Of Interes</i>	26
Tabel 3.6 Contoh Data Citra Pelatihan Daun Tin	26
Tabel 3.7 Data Citra Pengujian Daun Tin.....	28
Tabel 4.1 Hasil <i>Resize</i> Citra	33
Tabel 4.2 Hasil Konversi Citra <i>Bw Image</i>	34
Tabel 4.3 Hasil Segmentasi <i>Thresholding</i>	35
Tabel 4.4 Hasil Segmentasi <i>K-Means</i>	36
Tabel 4.5 Hasil <i>Region Of Interest</i>	38
Tabel 4.12 Hasil Pelatihan Algoritma <i>Naïve Bayes</i>	43
Tabel 4.13 Hasil Pengujian Algoritma <i>Naïve Bayes</i>	43

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Pemilihan <i>Thresholding</i> Analis Visual Histogram Bimodal	7
Gambar 2.2 (a) Citra Berukuran 5x5 Dengan Intensitas 0,1 Dan 2	12
Gambar 2.3 (b) Matriks Intensitas <i>Co-Occurance</i> Untuk $D=(1,1)$	12
Gambar 3.1 Tahapan Metode Penelitian.....	20
Gambar 3.2 Digitalisasi Citra	21
Gambar 4.1 Proses Digitalisasi Citra Daun Tin	31
Gambar 4.2 Data Set Citra Daun Tin Sehat Dan Sakit	32
Gambar 4.5 Hasil Pelatihan Klasifikasi Algoritma <i>Naïve Bayes</i>	41
Gambar 4.7 Hasil Pengujian Klasifikasi Algoritma <i>Naïve Bayes</i>	42
Gambar 4.9 Grafik Pengujian Algoritma <i>Naïve Bayes</i>	43
Gambar 4.8 Halaman Depan Gui Matlab	44
Gambar 4.12 Hasil Identifikasi Citra Daun Tin Sakit.....	46

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Data Set Daun Tin Sehat	67
Lampiran 2. Data Set Citra Daun Tin Sakit.....	68
.....	
Lampiran 3. Gambar Halaman Depan Gui Matlab	69
Lampiran 4. Gambar Pelatihan Klasifikasi <i>Naïve Bayes</i>	70
Lampiran 5. Gambar Pengujian Klasifikasi <i>Naïve Bayes</i>	71
Lampiran 6. Gambar Pengolahan Citra <i>Naïve Bayes</i>	72
Lampiran 7. Tabel Citra Daun Tin RGB Dan Bw Image	73
Lampiran 8. Tabel Citra Hasil <i>Thresholding</i>	74
Lampiran 9. Tabel Citra Hasil <i>K-Means</i>	75
Lampiran 10. Hasil <i>Region of Interest</i>	76
Lampiran 11. Source Code Aplikasi Matlab	77

BAB I

PENDAHULUAN

1.1 Latar Belakang Penulisan

Tanaman Tin (*Ficus Caric*) merupakan sejenis tumbuhan penghasil buah yang berasal dari Asia Barat. Nama Tin diambil dari bahasa Arab, yang berarti buah ara atau pohon ara. Di Indonesia, Tanaman Tin mulai banyak digemari dan dikembangkan dikarenakan memiliki banyak khasiat dalam tanaman tersebut. Berbagai penelitian menyebutkan bahwa kandungan senyawa polifenol yang terdapat pada daun dan buah Tin dapat digunakan guna menghambat pertumbuhan sel kanker[1] dan antikanker payudara[2].

Permintaan Tanaman Tin yang melonjak sering tidak dapat dipenuhi membuat Tanaman Tin menjadi langka di Indonesia. Kelangkaan tersebut disebabkan oleh penyakit dan virus yang menyerang pada Tanaman Tin. Misalkan saja penyakit *Cerotelium Fici* yang muncul pada musim hujan membuat daun Tin yang telah terinfeksi berubah warna menjadi cokelat kekuningan, daun rontok hingga membuat Tanaman Tin menjadi lemah yang mempengaruhi perkembangan buah[3].

Penyakit lain yang menyerang pada tanaman tin adalah jenis virus mosaik, penyakit ini pertama kali muncul di *California* dan menyebar ke sebagian besar wilayah Indonesia dan Amerika Serikat. Daun tin yang terinfeksi virus ini menjadi bintik-bintik cokelat menyebabkan pertumbuhan menjadi lambat dan cacat pada buah. Seringkali penyakit yang menyerang Tanaman Tin tidak segera bisa diatasi dikarenakan harus dilakukan pemeriksaan dan analisa jenis penyakitnya terlebih dahulu. Adapun dalam penanganannya dapat terjadi kesalahan seperti salah mendeteksi jenis penyakit dan salah melakukan pengobatannya[3].

Segmentasi Algoritma *K-Means* pada Daun Tin dapat digunakan untuk menganalisa penyakit yang ada pada daun Tin[4]. Melalui fitur *Gray level co-occurrence metrices* (GLCM) akan didapatkan tekstur *correlation*, *energy*, *homogeneity*, dan *contrast* dari daun Tin. Kemudian dengan menggunakan metode algoritma C4.5 dan Naïve Bayes dilakukan pembagian wilayah-wilayah homogen guna mendapatkan hasil analisa penyakit daun Tin dengan akurat.

Dengan melakukan Riset Segmentasi citra daun tin sakit dan sehat secara digitalisasi ini diharapkan dapat mendeteksi penyakit yang ada pada tanaman Tin dengan cepat sehingga mampu dilakukan penanganannya dengan tepat, efektif dan efisien.

1.2. Identifikasi Masalah

Identifikasi masalah yang terdapat pada penelitian ini adalah sebagai berikut:

1. Bagaimana hasil digitalisasi citra guna mendapatkan data set daun tin sehat dan sakit?
2. Apakah algoritma *K-Means* dapat mensegmentasi area daun tin sakit?
3. Bagaimana hasil analisis tekstur *Gray Level Co-occurrence Matrix* (GLCM) pada citra daun tin pada ekstraksi fitur *Contras*, *Correlation*, *Energy*, *Homogeneity* terhadap citra daun sehat dan sakit ?
4. Bagaimanakah hasil klasifikasi nilai yang didapat dari citra daun tin terhadap algoritma *Naïve Bayes* untuk tingkat keakurasiannya?

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Melakukan digitalisasi citra daun tin sehingga dapat digunakan sebagai data set penelitian.
2. Melakukan segmentasi *K-Means* pada citra daun tin sakit dan sehat sehingga dapat diketahui area daun yang terkena penyakit.
3. Mencari nilai tekstur dari *Contras*, *Correlation*, *Energy*, *Homogeneity* menggunakan metode analisa tekstur *Gray Level Co-occurrence Matrix* (GLCM) pada citra daun tin sakit dan sehat.
4. Mencari hasil terbaik perhitungan keakurasian nilai yang dihasilkan pada fitur GLCM dengan klasifikasi Algoritma *Naïve Bayes* dan algoritma C4.5.

Selain tujuan penelitian di atas, tujuan penulisan laporan tesis ini dibuat sebagai salah satu syarat untuk mendapatkan gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) STMIK Nusa Mandiri.

1.4. Ruang Lingkup Penelitian

Ruang lingkup masalah pada penelitian ini dalam membatasi pokok pembahasan agar terfokus dan memiliki tujuan guna dilakukannya segmentasi pada citra terhadap penyakit. Setelah dilakukan analisis tekstur, dan dilakukan perhitungan area, maka tahapan berikutnya dilakukan klasifikasi dalam citra daun tin. Data yang digunakan adalah citra daun tin sakit dan sehat masing sebanyak 60 citra total seluruh citra daun tin sebanyak 120 citra. Aplikasi dikembangkan menggunakan MATLAB. Algoritma segmentasi yang digunakan adalah Algoritma K-means Clustering, dan metode analisa tekstur yang digunakan adalah GLCM untuk mencari hasil terbaik yaitu algoritma *Naïve Bayes* dan Algoritma C4.5.

1.5. Sistematika Penulisan

Sistematika penulisan dalam penelitian ini berisi gambaran secara umum pada masing-masing bab, yang terdiri dari bab pendahuluan, landasan teori, metode penelitian, hasil penelitian dan pembahasan serta kesimpulan dan saran.

BAB I Pendahuluan

Bab pendahuluan membahas tentang latar belakang dari penelitian dan melakukan perumusan masalah yang diangkat dalam penelitian, selain itu dibahas juga mengenai tujuan dilakukannya penelitian, ruang lingkup, kontribusi penelitian, serta manfaat dari penelitian.

BAB II Landasan Teori

Bab landasan teori mengupas tentang tinjauan studi yaitu membahas penelitian sebelumnya yang digunakan untuk mengetahui sejauh mana penelitian sebelumnya dan membahas tentang landasan teori terkait penelitian yang dilakukan.

BAB III Metode Penelitian

Bab Metode Penelitian ini berisi tentang penjelasan mengenai perancangan penelitian diantaranya adalah metode pengumpulan data, metode yang diusulkan, eksperimen dan pengujian metode serta evaluasi dan validasi hasil.

BAB IV Hasil Penelitian dan Pembahasan

Bab hasil penelitian dan pembahasan membahas tentang hasil penelitian berupa hasil dari segmentasi, nilai tekstur yang dihasilkan oleh metode analisa tekstur,

implementasi algoritma C4.5 *dan algoritma Naïve Bayes* untuk akurasi awal yang dihasilkan oleh algoritma C4.5 *dan algoritma Naïve Bayes*, implementasi *rule* dengan program dan hasil keseluruhan.

BAB V Kesimpulan dan Saran

Bab kesimpulan dan saran berisi tentang ringkasan akhir dari penelitian serta saran-saran yang diberikan untuk mengembangkan penelitian.

BAB II

LANDASAN/KERANGKA PEMIKIRAN

2.1 Tinjauan Pustaka

Bagian ini menguraikan tentang teori-teori yang digunakan dalam penelitian ini mulai dari teknik pengolahan citra hingga metode-metode yang digunakan dalam penelitian segmentasi citra daun tin. Berikut beberapa teori dan metode yang digunakan :

2.1.1 Digitalisasi Citra

Digitalisasi citra yaitu suatu metode yang dilakukan untuk merubah citra dari citra analog menjadi citra digital, dalam bentuk piksel agar komputer dapat melakukan proses dalam manipulasi pada citra tersebut yang menghasilkan tampilan lebih baik dan indah. Digitalisasi citra sebagai penciptaan citra digital, seperti struktur bagian dari suatu objek. Digitalisasi citra diklasifikasikan berdasarkan jenis radiasi elektromagnetik yang disebut sebagai gelombang lain yang melewati atau memantulkan objek, sehingga dapat mengirimkan informasi berupa citra. Proses digitalisasi citra yaitu suatu teknologi memanipulasi bit-bit atau piksel untuk dapat meningkatkan kualitas citra atau membuat sudut pandang yang berbeda untuk mengekstrak nilai informasi dari suatu citra secara digital, dengan menggunakan algoritma komputer[5]. Peralatan penunjang dalam proses digitalisasi citra daun tin yaitu seperti *camera digital*, *Smartphone*, *USB Webcam*, *Scanner* dan mikroskop *digital*[6].

2.1.2 Resize

Proses melakukan perubahan ukuran luas citra dari lebih besar atau lebih kecil dari ukuran aslinya merupakan proses *resizing*. Dalam tahapan *preprocessing* merubah ukuran citra asli atau melakukan *resize*, hal ini dilakukan agar semua citra yang diinputkan memiliki ukuran citra yang sama, sehingga akan memudahkan dalam proses mengolah citra ke tahapan berikutnya[7].

2.1.3 Konversi Citra

Pada tahapan konversi Citra terdapat beberapa tahapan-tahapan pemrosesan yang dilakukan. Proses yang pertama melakukan proses resizing yang berfungsi untuk menyamakan ukuran citra input agar ukuraannya seragam. Kemudian dilakukan konversi citra RGB ke *Grayscale* dengan melakukan operasi komplemen agar objek berwarna putih dan background berwarna hitam. Selanjutnya dilakukan operasi morfologi untuk menyempurnakan hasil segmentasi[8].

2.1.4 Segmentasi Citra

Segmentasi membagi citra ke dalam sejumlah *region* atau obyek. *Level* untuk pembagian tergantung pada masalah yang diselesaikan. Maka, Segmentasi seharusnya berhenti ketika obyek yang diinginkan dalam aplikasi telah terisolasi. Misalnya, pemeriksaan otomatis pada rakitan produk elektronik. Yang diinginkan adalah analisis citra produk dengan tujuan untuk mengetahui ada atau tidaknya penyimpangan tertentu seperti salah komponen, atau lintasan hubungan yang terputus[9].

Algoritma segmentasi citra umumnya didasarkan pada satu dari dua properti nilai intensitas diskontinuitas dan similaritas. Dalam kategori pertama, pendekatannya adalah memecah atau memilah citra berdasarkan perubahan kasar dalam intensitas, seperti tepi dalam citra. Pendekatan utama kategori kedua didasarkan pada pemecahan citra ke dalam *region* yang sama menurut sejumlah kriteria yang didefinisikan, seperti *thresholding*, *region growing*, *region splitting and merging*[10].

a. Bw Image

Citra yang sudah dikonversi menjadi citra *Grayscale* yang hanya mempunyai nilai derajat keabuan. Selanjutnya diubah ke citra biner menjadi citra hitam dan putih. Citra biner direpresentasikan dengan dua intensitas warna pada tiap pikselnya yaitu 0 dan 1, dimana nilai 1 mewakili warna hitam dan nilai 0 mewakili warna putih. Pada Citra biner, piksel-piksel objek bernilai 1 dan piksel-piksel latar belakang bernilai 0[11].

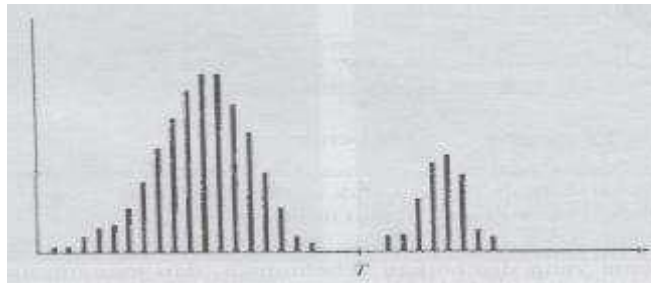
b. *Thresholding*

Karena properti intuitif dan kesederhanaanya dalam implementasi, *Thresholding* citra menjadi titik pusat dalam aplikasi segmentasi citra. Di sini akan dibahas mengenai cara pemilihan nilai *thresholding* secara otomatis dan memperhatikan metode untuk bermacam-macam *thresholding* menurut properti ketetapan citra lokal [12].

Andaikan bahwa histogram intensitas yang ditunjukkan pada gambar 2.1 yang berkaitan dengan citra $f(x,y)$, yang terdiri dari obyek terang pada background gelap, maka piksel obyek dan background mempunyai level intensitas yang dikelompokkan ke dalam dua mode dominan. Satu cara yang jelas untuk mengesktrak sembarang titik (x,y) untuk dimana $f(x,y) > T$ disebut object point. Sedangkan yang lain disebut background point. Dengan kata lain, citra yang di-*thresholding* $g(x,y)$ didefinisikan sebagai [13] :

$$g(x,y) = \begin{cases} 1 & \text{jika } f(x,y) \geq T \\ 0 & \text{jika } f(x,y) < T \end{cases}$$

Piksel yang diberi nilai 1 berkaitan dengan obyek sedangkan piksel yang diberi nilai 0 berkaitan dengan *background*. Ketika T adalah konstanta, pendekatan ini disebut *global thresholding*.



Gambar 2.1 Pemilihan *thresholding* secara analisis visual histogram bimodal

Sumber: [9].

c. *Region Growing*

Sebagaimana implikasi namanya, *region growing* adalah prosedur yang mengelompokkan piksel atau sub-region kedalam region yang lebih besar berdasarkan pada kriteria yang sudah ditentukan untuk pertumbuhannya. Pendekatan dasarnya adalah memulai dengan titik seed dan dari sini menumbuhkan *region* oleh

penambahan pada setiap seed piksel tetangga yang mempunyai properti sama dengan seed (seperti range spesifik dari *gray level* atau warna)[9].

Pemilihan sejumlah satu atau lebih starting point sering menjadi dasar masalah. Satu prosedur untuk menghitung pada setiap piksel kesamaan sekumpulan properti yang akhirnya akan digunakan untuk memberi nilai piksel pada *region* selama proses *growing*. Jika hasil dari komputasi ini menunjukkan cluster nilai, piksel yang tempat propertinya dekat *cluster centroid* dapat digunakan sebagai seed [13].

Pemilihan kesamaan kriteria tidak hanya tergantung pada masalah tersebut, tetapi juga pada jenis data citra yang tersedia. Ketika citranya monokrom analisis region harus dilakukan dengan sejumlah deskriptor yang didasarkan pada level intensitas (seperti momen atau tekstur yang dan properti spasial)[13].

Deskriptor itu sendiri dapat memberikan hasil yang tidak benar jika informasi konektivitas (*adjacency*) tidak digunakan dalam proses *region growing*. Misalnya, visualisasi susunan *random* piksel dengan hanya tiga perbedaan nilai intensitas. Pengelompokan piksel dengan *level* intensitas yang sama untuk membentuk “*region*” tanpa perhatian konektivitas akan memberikan hasil segmentasi yang kurang bermakna[9].

d. *region splitting and merging*.

Metode *split-and-merge* adalah salah satu metode segmentasi yang banyak digunakan. *Split-and-merge* dengan struktur *quadtree* adalah pemecahan populer pada segmentasi citra karena sederhana dan komputasinya efisien. Algoritma *split-and-merge*[14] dengan struktur *quadtree* tidak mampu menyesuaikan dengan semantik citra. Tepi-tepi segmen yang dibentuk oleh algoritma ini hanya mempunyai dua arah (vertikal dan horisontal) dan posisinya dibatasi oleh batas node *quadtree*. Kesalahan dalam arah dan posisi tepi akan mengakibatkan kesalahan dalam segmentasi[14].

Metode ini dimulai dengan mengasumsikan bahwa keseluruhan citra (region R) adalah homogen. Jika asumsi ini tidak terpenuhi, maka citra dibagi (*split*) menjadi empat subregion (R_1 , R_2 , R_3 , R_4) yang luasnya sama. Prosedur ini

dilaksanakan secara rekursif terhadap subregion-subregion yang dihasilkan sampai didapatkan subregion-subregion yang homogen. Subregion subregion yang bertetangga akan digabung (*merge*) jika memenuhi kriteria [14].

2.1.5 Segmentasi *K-Means*

Metode *clustering* non hirarki atau disebut dengan *K-means clustering*, merupakan pengelompokan suatu data menggunakan metode *K-Means* yang secara umum dapat dilakukan dengan algoritma dasar seperti menentukan jumlah cluster awal, menempatkan centroid sesuai dengan jumlah *cluster* secara acak, mengalokasikan *cluster* yang sesuai dengan centroid dan menggunakan perhitungan jarak terdekat. Data yang digunakan untuk dikluster diperoleh dengan membandingkan jarak (*distance*) untuk menentukan tingkat kesamaan atau ketidaksamaan dua vektor fitur [13].

Proses segmentasi menggunakan metode *K-Means* dengan menghitung jarak kemiripan warna dari hasil segmentasi menggunakan metode *K-Means*. Selanjutnya pendekatan warna menggunakan *Euclidean Distance* untuk mendeteksi penyakit tanaman daun jagung yaitu antara penyakit hawar daun dan bercak daun didapatkan presentase pengujian sistem sebesar 90%[15].

2.1.6 *Region Of Interest*

Region of Interest (ROI) adalah bagian salah satu proses teknik segmentasi untuk mengolah citra *digital* sehingga pengguna mampu mengolah citra yang mengandung nilai informasi dari data citra yang dikehendaki. ROI bekerja dalam pengkodean secara berbeda pada bagian wilayah tertentu dari citra *digital*, sehingga bagian wilayah citra yang lebih penting akan memiliki kualitas informasi citra yang lebih baik dari wilayah citra disekitarnya[4]. Untuk mendapatkan *region of interest* (ROI) suatu citra, sebelumnya dilakukan *region filling* terlebih dahulu guna memisahkan citra dengan background-nya[16].

Algoritma sederhana untuk *region filling* didasarkan pada sejumlah dilasi, komplementasi, dan interseksi dinyatakan himpunan yang berisi subset dimana elemen-elemennya adalah *boundary* titik-titik *8-connected* dari *region*. Dimulai dengan sebuah titik *p* didalam *boundary*, tujuannya adalah untuk mengisi semua

region dengan nilai 1. Jika mengikuti konvensi bahwa semua titik *non-boundary* (*background*) diberi nilai 0, maka p harus diberi nilai 1 untuk memulai. Prosedur dibawah ini adalah untuk mengisi *region* dengan nilai 1[16]:

$$X_k = (X_{k-1} + B) \cap A^c \quad K = 1, 2, 3 \dots$$

Dimana $X_0 = p$ dan B adalah sterl simetrik. Algoritma berhenti pada iterasi langkah ke k jika $X_k = X_{k-1}$ himpunan union dari X dan A mengisi daerah isian dan boundary-nya[16].

Proses dilasi akan mengisi semua daerah jika belum diperiksa. Interseksi pada setiap langkah dengan A^c membatasi hasil agar selalu didalam region yang ditentukan. Hal tersebut adalah proses morfologi yang dapat diisyaratkan untuk mencapai properti yang diharapkan[16].

2.1.7 Data Set Pelatihan dan Pengujian

Klasifikasi algoritma C4.5 dan algoritma *Naïve Bayes* pada data set citra pelatihan dan pengujian ditujukan untuk mencari hasil terbaik dalam tingkat akurasi terbaik dari ekstrasi fitur *contrast*, *correlation*, *energy*, *homogeneity*. Akurasi terbaik yang didapat dari pengklasifikasian algoritma C4.5 dan algoritma *Naïve Bayes* seperti jumlah data benar, jumlah data salah, dan nilai akurasi terbaik dari kedua algoritma seperti C4.5 dan algoritma *Naïve* dalam mengklasifikasi citra pelatihan dan pengujian citra daun tin[17].

2.1.8 Metode Analisis Tekstur GLCM

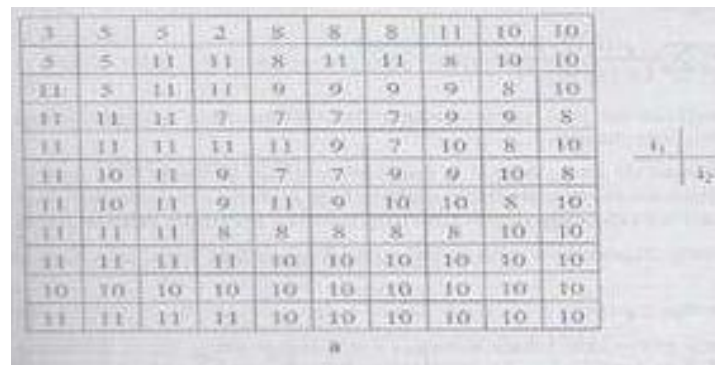
Gray level Co-occurrence Matrix (GLCM) adalah metodologi pemrosesan gambar yang digunakan untuk menggambarkan hubungan spasial antara nilai-nilai abu-abu dalam citra 2 dimensi. Sejak itu, GLCM telah digunakan secara luas dalam berbagai aplikasi. Dari semua teknik analisis tekstur, saat ini mungkin yang paling banyak digunakan adalah yang didasarkan pada algoritma GLCM[18]. Metode GLCM sebagai cara untuk mengklasifikasikan citra menggunakan pengukuran statistik urutan kedua[19].

Matrik intensitas *co-occurrence* merupakan suatu matrik yang menggambarkan frekuensi munculnya pasangan dua piksel dengan intensitas tertentu dalam jarak dan arah tertentu dalam citra. Matriks intensitas

co-occurrence $p(i_1, i_2)$ didefinisikan dengan dua langkah sederhana sebagai berikut. Langkah pertama adalah menentukan lebih dulu jarak antara dua titik dalam arah vertikal dan horizontal (vektor $d=(dx,dy)$), dimana besaran dx dan dy dinyatakan dalam piksel sebagai unit terkecil dalam citra digital. Langkah kedua adalah menghitung pasangan piksel-piksel yang mempunyai nilai intensitas i_1 dan i_2 dan berjarak di piksel dalam citra. Kemudian hasil setiap pasangan nilai intensitas diletakkan pada matriks sesuai dengan koordinatnya, dimana absis untuk nilai intensitas i_1 dan kordinat untuk nilai intensitas i_2 . Sebagai contoh misalnya citra berukuran 5x5 piksel dan mempunyai intensitas 0,1 dan 2 seperti terlihat pada gambar 2.2, akan dianalisis teksturnya. Karena hanya ada tiga nilai intensitas (0,1 dan 2) maka matriks $p(i_1, i_2)$ akan berukuran 3x3. Bila jarak antar piksel ditentukan (1,1), yang berarti satu piksel ke kanan dan satu piksel ke bawah maka pasangan piksel yang harus dihitung adalah berjarak satu piksel dengan arah sudut 135° dari sumbu vertikal. Dalam citra berukuran 5x5 ada 16 pasangan yang memenuhi syarat ini[9].

Kemudian pasangan piksel, dimana piksel pertama mempunyai nilai intensitas i_1 dan pasangan yang berjarak d mempunyai nilai intensitas i_2 , dihitung dan dimasukkan kedalam kolom ke- i_1 dan baris ke i_2 , pada matriks $p(i_1, i_2)$, menggunakan contoh pada gambar 2.2, ada tiga pasangan piksel yang mempunyai nilai intensitas (2,1) dan terpisah dengan jarak d seperti yang ditetapkan semula, maka nilai kordinat yang bersangkutan pada matriks $p(i_1, i_2)$ adalah 3. Matriks yang sudah lengkap diisi terlihat pada gambar 2.3[20].

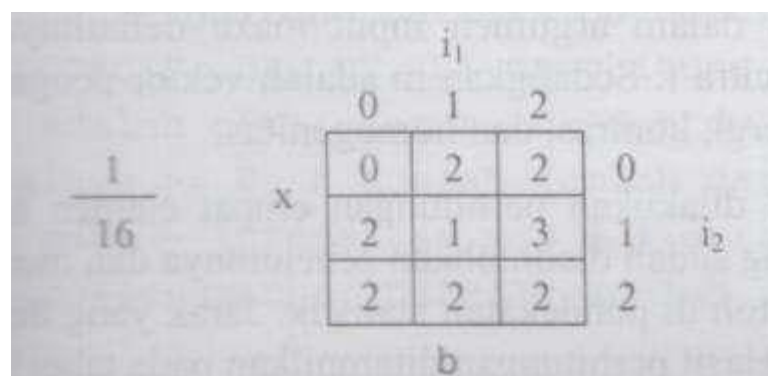
Selanjutnya tiap elemen matriks $p(i_1, i_2)$ perlu dinormalisasikan dengan cara membaginya dengan bilangan yang merupakan jumlah total dari pasangan piksel. Pada contoh, tiap elemen dibagi dengan 16 karena jumlah tiap pasangan dalam gambar 2.3 adalah 16. Nilai-nilai elemen matriks setelah dinormalisasi kemudian dapat diperlakukan sebagai fungsi probabilitas karena jumlah seluruh elemen sekarang bernilai 1[20].



3	5	5	2	8	8	8	11	10	10
5	5	11	11	8	11	11	8	10	10
11	5	11	11	9	9	9	9	8	10
11	11	11	7	7	7	7	9	9	8
11	11	11	11	11	9	7	10	8	10
11	10	11	9	7	7	9	9	10	8
11	10	11	9	11	9	10	10	8	10
11	11	11	8	8	8	8	8	10	10
11	11	11	11	10	10	10	10	10	10
10	10	10	10	10	10	10	10	10	10
11	11	11	11	10	10	10	10	10	10

Gambar 2.2 Citra Berukuran 5x5 Dengan Intensitas 0,1 Dan 2

Sumber:[13]



		i_1			
		0	1	2	
i_2	0	0	2	2	0
	1	2	1	3	1
	2	2	2	2	2

b

Gambar 2.3 Matriks Intensitas Co-Occurance Untuk $D=(1,1)$

Sumber:[13]

Salah satu sifat tekstur dapat diambil dari statistik nilai intensitas abu-abu dalam citra, yaitu rata-rata (*mean*). Nilai *mean* dari suatu sebaran nilai intensitas citra abu-abu dapat dicari tanpa bantuan matriks *co-occurrence*, karena sifatnya sederhana. Tetapi untuk mengesktrak sifat-sifat lain dalam analisis tekstur, matriks *co-occurece* sangat diperlukan untuk membantu perhitungan sifat-sifat yang akan diekstrak dari citra tersebut.

Sebuah fitur untuk mengukur keteracakan dari distribusi intensitas disebut entropi dan didefinisikan sebagai berikut[20]:

$$Entropi = - \sum_{i_1} \sum_{i_2} p(i_1, i_2) \log p(i_1, i_2) \dots\dots\dots (2.1)$$

Nilai *entropi* maksimum jika semua elemen $p(i_1, i_2)$ sama, yaitu matriks yang berhubungan dengan citra dimana tidak terdapat susunan tertentu dalam pasangan intensitas dengan jarak vektor tertentu (daerah konstan).

Fitur yang lain adalah energi, yaitu fitur untuk mengukur konsentrasi pasangan intensitas pada matriks co-occurance, dan didefinisikan dengan :

$$Energi = \sum_{i_1} \sum_{i_2} p^2(i_1, i_2) \dots\dots\dots(2.2)$$

Nilai energi makin membesar jika pasangan piksel yang memenuhi syarat matriks intensitas co-occurrence terkonsentrasi pada beberapa koordinat dan mengecil jika letaknya menyebar. Fitur lainnya yaitu kontras yang digunakan mengukur kekuatan perbedaan intensitas dalam citra dan dinyatakan dengan :

$$Kontras = \sum_{i_1} \sum_{i_2} (i_1, i_2)^2 p(i_1, i_2) \dots\dots\dots(2.3)$$

Nilai kontras membesar jika variasi intensitas dalam citra tinggi, dan menurun jika variansinya rendah. Kebalikan dari kontras adalah homogenitas, yaitu untuk mengukur kehomogenan variasi intensitas dalam citra, dan didefinisikan dengan :

$$Homogenitas = \sum_{i_1} \sum_{i_2} \frac{p(i_1, i_2)}{1 + |i_1 - i_2|} \dots\dots\dots(2.4)$$

Nilai homogenitas membesar jika variansi intensitas dalam citra mengecil dan sebaliknya mengecil jika variansi dalam citra membesar.

2.1.9 Algoritma Naïve Bayes

Algoritma *Naïve Bayes* merupakan suatu metode yang tidak memiliki aturan. Algoritma *Naïve Bayes* merupakan suatu cabang algoritma matematika yang dikenal dengan teori probabilitas untuk mencari peluang terbesar dari kemungkinan klasifikasi, dengan cara melihat frekuensi tiap klasifikasi pada data training. Algoritma *Naïve Bayes* termasuk merupakan suatu metode klasifikasi yang populer, dan masuk dalam 10 algoritma terbaik dalam *data mining*. Algoritma ini dikenal dengan nama *Idiot's Bayes*, *Simple Bayes* dan *Independence Bayes*[17].

Klasifikasi algoritma *Naive Bayes* merupakan pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class. Klasifikasi Bayesian didasarkan pada teorema *Bayes*, yang diambil dari nama seorang ahli matematika yang juga menteri *Prebysterian* Inggris, *Thomas Bayes* (1702-1761). Klasifikasi bayesian memiliki kemampuan klasifikasi serupa dengan *decision tree* dan *neural network*. *Bayes rule* digunakan untuk menghitung probabilitas suatu *class*. Algoritma *Naive Bayes* memberikan suatu cara mengkombinasikan peluang terdahulu dengan syarat kemungkinan menjadi sebuah formula yang dapat digunakan untuk menghitung peluang dari tiap kemungkinan yang terjadi[17].

2.1.10 Algoritma Decision Tree (C4.5)

Perkembangan dari Algoritma ID3, adalah Algoritma C4.5 yang diperkenalkan oleh J. Ros Quinlan, yang diperuntukan untuk membentuk pohon keputusan atau biasa dikenal dengan *decision tree*. Dan dianggap sebagai salah satu pendekatan yang paling populer dalam pengklasifikasian. Dalam klasifikasi *decision tree* atau pohon keputusan terdiri dari sebuah node yang membentuk akar. Node akar tidak memiliki input, nod lain yang bukan sebagai akar tetapi tepat satu input disebut sebagai node internal atau test node, sedangkan node lainnya dinamakan daun. Daun mewakili nilai target yang paling tepat dari satu *class*[17].

Pohon keputusan dapat dibuat dengan melakukan pembagian nilai-nilai pada atribut menjadi cabang untuk setiap kemungkinan. Pohon keputusan bekerja dengan melakukan penelusuran dari akar hingga ke cabang sampai class suatu objek ditemukan. *Instance* diklasifikasikan dengan mengarahkan dari akar pohon sampai ke daun sesuai dengan hasil tes melalui node internal[17].

Konsep dasar dari algoritma C4.5 yaitu mengubah data menjadi pohon keputusan (*decision tree*) dan aturan-aturan keputusan (*rule*). C4.5 adalah algoritma yang cocok untuk masalah klasifikasi dan *data mining*. C4.5 memetakan nilai atribut menjadi *class* yang dapat diterapkan untuk klasifikasi baru[17].

2.2 Tinjauan Studi

Penelitian tentang citra penyakit pada daun sudah banyak dilakukan, terutama dalam penelitian tentang segmentasi penyakit citra daun dengan region of interest (ROI). Untuk mendapatkan area citra daun sakit dan tidak sakit, melalui analisa tekstur *gray level co-occurrence matrix* (GLCM). Berikut ini beberapa penelitian terkait dengan segmentasi ROI pada citra daun sakit dan tidak sakit melalui analisa tekstur GLCM :

1. Penelitian yang dilakukan oleh (Juee P. Gaikwad et al. 2017) yang berjudul *Region of Interest of Diseased Leaf using Overlapping Window Technique* pada penelitian ini mengusulkan tentang Teknik *Overlapping Window* yang digunakan untuk mencari *Region Of Interest* (ROI) pada daun yang sakit. Dari

- hasil penelitian di dapatkan bahwa penggunaan tehnik ini dipengaruhi oleh berbagai ukuran window dan inputan gambar yang telah kita ambil sebelumnya.
2. Penelitian yang dilakukan oleh (Aditya Sinha et al 2020) yang berjudul *Olive Spot Disease Detection and Classification using Analysis of Leaf Image Textures* penelitian ini menganalisa Penyakit bercak daun pada Tanaman Zaitun. Dengan melakukan analisa tekstur bercak pada Daun Zaitun, ditemukan penyakit bercak daun neofabrea berkorelasi tinggi pada energi ($r = 0,97$) dan sifat entropi ($r = 0,92$). Parameter ini membantu mengklasifikasikan jenis penyakit yang ada pada daun Zaitun.
 3. Penelitian yang dilakukan oleh (J. Praveen Kumar,&S. Domnic 2018) yang berjudul *Image based leaf segmentation and counting in rosette plants*. Dalam metode ini menggunakan perhitungan daun (*Leaf Counting Method*), dengan objek yang digunakan tanaman rosette. Metode penghitungan daun merupakan teknik peningkatan citra dan model grafik untuk segmentasi citra tanaman. Metode ini bergantung pada distribusi kecerahan, algoritma grafik, dan fitur warna hasil yang akurasi yang dicapai Yang dengan Menerapkan CHT akurasi segmentasi 95,4% dan juga mencapai penghitungan akurasi 0,7.
 4. Penelitian yang dilakukan oleh (Kankamon Phookronghin et.al 2018) yang berjudul *2 Level Simplified Fuzzy ARTMAP for Grape Leaf Disease System Using Color Imagery and Gray Level Co-Occurrence Matrix*. Penelitian ini menggunakan sistem ini SOFM dan SFAM untuk mengekstraksi daerah penyakit dari daun anggur. Dan mengklasifikasikan atribut penyakit dengan menggunakan pencitraan Warna dan data statistik dari GLCM. Manfaat 2L-SFAM adalah reorganisasi terus-menerus tanpa proses pelatihan ulang, yang dapat mengurangi waktu untuk mempelajari dan mengklasifikasikan proses secara efisien. Hal itu akan membuat algoritma ini bekerja dengan kinerja terbaik.
 5. Penelitian yang dilakukan oleh (Žarko Ivanović ,et al..2018), yang berjudul *Leaf spot disease on Philodendron scandens, Ficus carica and Actinidia deliciosa caused by Pseudomonas syringae pv. syringae in Serbia*. Objek penelitian melakukan Uji biokimia dan molekuler, penyakit bercak daun pada tanaman *Philodendron scandens*, tanaman *Ficus carica* dan tanaman *Actinidia deliciosa*.

Berdasarkan uji biokimia dan molekuler *Pseudomonas syringae* pv. *syringae* diidentifikasi sebagai agen penyebab bercak daun bakteri philodendron daun-hati dan berada dalam musim lembab yang bertanggung jawab atas munculnya penyakit. Keragaman genetik dalam patogen ini diisolasi dari inang yang berbeda di Serbia tengah, dipelajari untuk memperoleh wawasan tentang etologi penyakit

6. Penelitian yang dilakukan oleh (A. K. Khambampati et al., 2018), penelitian yang berjudul *An automatic detection of the ROI using Otsu thresholding in nonlinear difference EIT imaging*. Penelitian melakukan Metode berbasis pencitraan *diferensial nonlinier* diusulkan untuk merekonstruksi distribusi resistivitas internal menggunakan EIT. Metode Otsu digunakan untuk mengklasifikasikan daerah sebagai target dan latar belakang. Dari pengujian didapatkan bahwa metode NDIWO yang diusulkan telah meningkatkan resolusi spasial dengan latar belakang yang seragam dan memiliki lebih sedikit kesalahan gambar dan koefisien korelasi yang lebih tinggi dibandingkan dengan ABI dan LDI konvensional.
7. Penelitian yang dilakukan oleh (Namrata R. Bhimte & V. R. Thool 2018) penelitian ini berjudul *Diseases Detection of Cotton Leaf Spot using Image Processing and SVM Classifier*. Metode yang digunakan GLCM SVM, dengan objek penelitian daun kapas hasil yang dicapai Multi Class SVM memiliki keakuratan yang tinggi mencapai 98,46% dalam hal mendeteksi penyakit yang ada pada daun kapas (*cotton*).

Tabel 2.1 Penelitian Terdahulu

Judul Penelitian	Tahun	Kelebihan	Perbedaan
<i>Region of Interest of Diseased Leaf using Overlapping Window Technique</i>	2017	Tentang Teknik <i>Overlapping Window</i> yang digunakan untuk mencari <i>Region Of Interest</i> (ROI) pada daun yang sakit. Dari hasil penelitian di	Mensegmentasi area Roi dengan klasifikasi algoritma C4.5 dan Naïve untuk mendapatkan tingkat akurasi yang lebih baik dalam klasifikasi. Objek

		dapatkan bahwa penggunaan tehnik ini dipengaruhi oleh berbagai ukuran window dan inputan gambar yang telah kita ambil sebelumnya. Objek yang digunakan daun Anggur	daun Tin (Ara)
<i>Image based leaf segmentation and counting in rosette plants</i>	2018	Dalam metode ini menggunakan perhitungan daun (<i>Leaf Counting Method</i>), dengan objek yang digunakan tanaman rosette. Metode penghitungan daun merupakan teknik peningkatan citra dan model grafik untuk segmentasi citra tanaman. Metode ini bergantung pada distribusi kecerahan, algoritma grafik, dan fitur warna hasil yang akurasi yang dicapai Yang dengan Menerapkan CHT akurasi segmentasi 95,4% dan juga mencapai	Mensegmentasi area Roi dengan klasifikasi algoritma C4.5 dan Naïve untuk mendapatkan tingkat akurasi yang lebih baik dalam klasifikasi. Objek daun Tin (Ara)

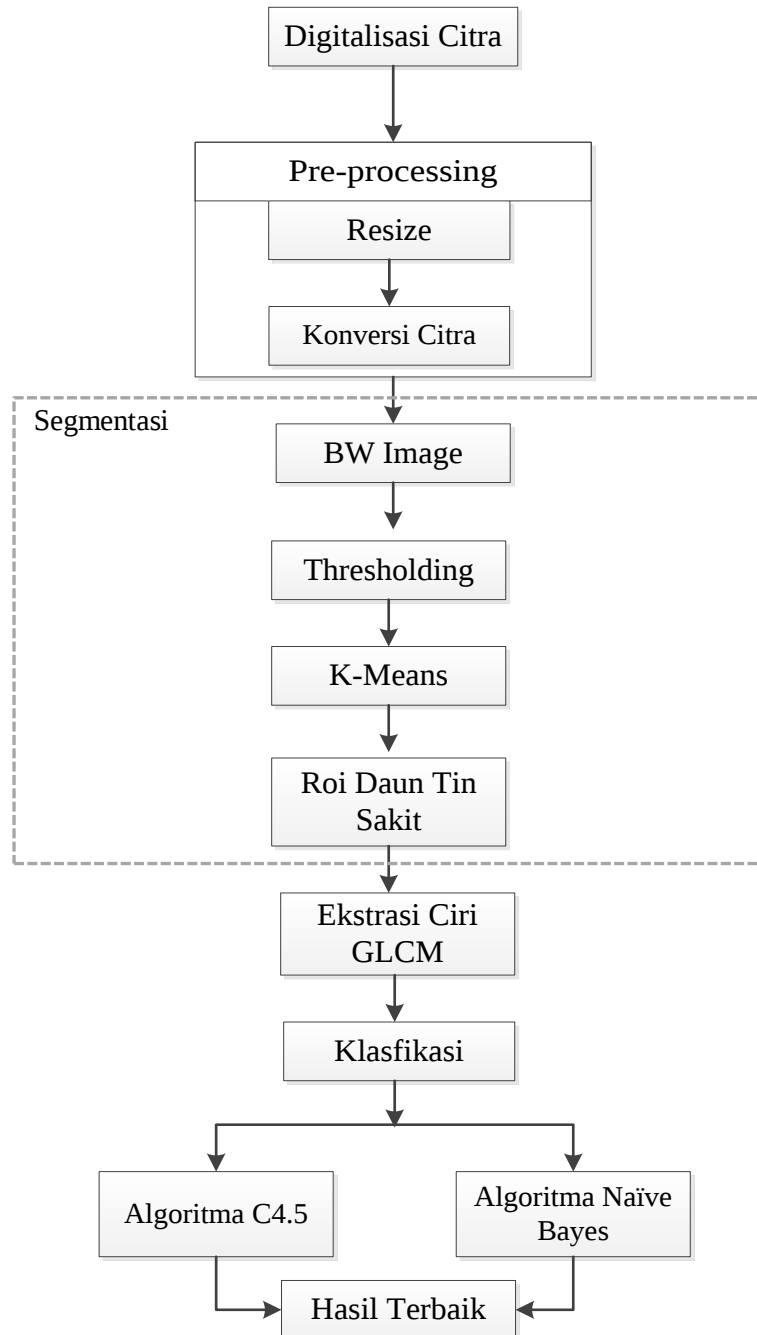
		penghitungan akurasi 0,7.	
<i>2 Level Simplified Fuzzy ARTMAP for Grape Leaf Disease System Using Color Imagery and Gray Level Co-Occurrence Matrix</i>	2018	Penelitian ini menggunakan sistem ini SOFM dan SFAM untuk mengekstraksi daerah penyakit dari daun anggur. Dan mengklasifikasikan atribut penyakit dengan menggunakan pencitraan Warna dan data statistik dari GLCM Objek yang dipakai daun Anggur	Mensegmentasi area Roi dengan ekstraksi fitur glcm menggunakan klasifikasi algoritma C4.5 dan Naïve untuk mendapatkan tingkat akurasi yang lebih baik dalam klasifikasi. Objek daun Tin (Ara)
<i>An automatic detection of the ROI using Otsu thresholding in nonlinear difference EIT imaging</i>	2018	Metode Otsu digunakan untuk mengklasifikasikan daerah sebagai target dan latar belakang. Dari pengujian di dapatkan bahwa metode NDIWO yang diusulkan telah meningkatkan resolusi spasial dengan latar belakang yang seragam dan memiliki lebih sedikit kesalahan gambar dan koefisien	melakukan operasi komplemen agar objek berwarna putih (1) dan background berwarna hitam (0) dan melakukan operasi morfologi untuk menyempurnakan hasil segmentasi mengubah nilai <i>background</i> pada citra grayscale menjadi nol sehingga diperoleh region of interest (ROI) melakukan subtraksi untuk mendapatkan <i>region</i> penyakit pada

		korelasi yang lebih tinggi dibandingkan dengan ABI dan LDI konvensional	daun dengan menandai <i>region</i> penyakit dan membentuk matriks <i>Co-oCurensi</i>
<i>Diseases Detection of Cotton Leaf Spot using Image Processing and SVM Classifier</i>	2018	Metode yang digunakan GLCM SVM, dengan objek penelitian daun kapas hasil yang dicapai Multi Class SVM memiliki keakurasian yang tinggi mencapai 98,46% dalam hal mendeteksi penyakit yang ada pada daun kapas (<i>cotton</i>).	Hasil ekstraksi fitur GLCM digunakan untuk mengklasifikasikan algoritma C4.5 dan Naïve Bayes

BAB III

METODOLOGI PENELITIAN

Tahapan – tahapan metodologi penelitian yang dilakukan dalam penelitian dapat dilihat pada Gambar 3.1. dan dilanjutkan dengan penjelasan berikut :



Sumber: preprocessing[5]

Sumber: hasil penelitian (2020)

Gambar 3.1 Tahapan Metode Penelitian

3.1 Digitalisasi Citra

Pada tahapan ini dilakukan pengolahan data awal yaitu melakukan proses digitalisasi citra. Dimana data analog citra daun tin diambil menggunakan kamera *Canon Eos 1200D* yang dihubungkan dengan laptop. Untuk dilakukan proses perubahan menjadi citra *digital* menggunakan kamera *Canon Eos 1200D*.



Tanaman tin, pengambilan daun tin untuk diproses menjadi citra digital



Proses pengambilan citra daun tin menggunakan kamera *Canon Eos 1200D* yang dihubungkan ke laptop

Sumber: hasil penelitian(2020)

Gambar 3.2 Digitalisasi Citra

Tabel 3.1 Contoh Citra Daun Tin Sakit

Jenis Penyakit	Contoh Citra
<i>Cerotelium Ficidan</i> (jenis penyakit karat daun yang menyerang pada daun tin)	
Virus Mosaik daun tin yang terinfeksi virus ini menjadi bintik-bintik cokelat menyebabkan pertumbuhan tanaman tin menjadi lambat	
Kutu Kebul hama berbintik putih yang biasa menyerang pada tanaman diarea terbuka, menyebabkan pertumbuhan tanaman terhambat	

Pada tabel 3.1 ditampilkan contoh penyakit yang menyerang pada tanaman daun tin, diantaranya:

Cerotelium Fici adalah jenis penyakit karat daun yang menyerang pada daun tin, dan menjadi ancaman terbesar terhadap produksi buah tin. Tingkat keparahan pada penyakit karat pada daun tin ini muncul pada saat musim hujan. Daun yang telah terinfeksi berubah warna menjadi coklat kekuningan, daun menjadi rontok dan pohon menjadi lemah menyebabkan terganggunya ke perkembangan tanaman tin.

Virus Mosaik, Daun tin yang terinfeksi virus ini menjadi bintik-bintik coklat menyebabkan pertumbuhan tanaman tin menjadi lambat.

Kutu Kebul hama berbintik putih yang biasa menyerang pada tanaman diarea terbuka, menyebabkan pertumbuhan tanaman terhambat

3.2 *Resize*

Untuk mempercepat waktu proses pengolahan citra pada saat dilakukan maka dilakukan perubahan ukuran citra. Karena citra terlalu besar ukurannya maka dilakukan *resize* dengan mengubah ukuran citra semula 3000x2216, maka dilakukan *resize* menjadi 10% dari ukuran semula.

3.3 *Bw Image*

Citra *bw image* atau biner direpresentasikan dengan dua intensitas warna yaitu hitam dan putih. Pada tiap pikselnya yaitu 0 dan 1, dimana nilai 1 mewakili warna hitam dan nilai 0 mewakili warna putih. Pada Citra biner, piksel-piksel objek bernilai 1 dan piksel-piksel latar belakang bernilai 0. Pada tabel 3.2 contoh hasil konversi warna RGB citra daun tin ke *bw image* kemudian disegmentasi dengan menggunakan metode segmentasi *Thresholding* untuk memisahkan antara objek dengan *background* sehingga didapatkan area daun tin yang selanjutnya disegmentasi dengan Algoritma *K-Means* kluster. Berikut contoh hasil algoritma *Thresholding* setelah dilakukan proses *bw image* :

Tabel 3.2 Contoh Konversi Citra Bw *Image*

No	Nama File	Citra RGB	Citra Bw <i>Image</i>
1	Sehat01.jpg		
2	Sehat02.jpg		
3	Sehat03.jpg		

Pada Tabel 3.2 contoh hasil konversi warna RGB citra daun tin *bw image* tampak citra daun tin yang berubah warna menjadi hitam dan putih setelah dilakukan proses *bw image*. Dari proses *bw image* selanjutnya dilakukan segmentasi dengan algoritma *Thresholding* untuk memisahkan antara objek citra daun tin dengan *background*.

4.4 *Thresholding*

Setelah dilakukan proses *bw image* terhadap citra daun tin, maka tahapan selanjutnya yang peneliti lakukan adalah mensegmentasi citra menggunakan algoritma *Thresholding*. Dimana tahapan ini adalah untuk memisahkan antara background dengan objek yang akan diteliti. Berikut contoh pada tabel 3.2 merupakan perubahan setelah dilakukannya *bw image* ke *Thresholding*.

Tabel 3.3 Contoh Hasil *Thresholding*

No	Nama File	Citra RGB	Bw Image	Thresholding
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			

Tabel 3.2 merupakan hasil dari algoritma *Thresholding* bisa kita lihat perbedaan warna yang tampak pada *bw image* hitam dan putih sedangkan pada algoritma *Thresholding* menunjukkan hitam artinya nilai pada algoritma *Thresholding* lebih besar sehingga didapatkan warna dominan hitam.

3.5 Segmentasi *K-Means*

Proses segmentasi klustering dilakukan dengan cara mengkonversi ruang warna citra RGB menjadi ruang warna L^*a^*b . Selanjutnya komponen a dan b dari citra L^*a^*b digunakan sebagai nilai masukan dalam algoritma *K-means*. Memilih kluster yang terdapat objek citra daun tin. Pemilihan didasarkan pada kluster yang memiliki luasan objek paling kecil. Pada tabel 3.3 Pemilihan didasarkan pada kluster yang memiliki luasan objek paling kecil. Setelah diperoleh citra daun tin saja, selanjutnya dilakukan operasi morfologi untuk menyempurnakan hasil segmentasi.

Tabel 3.4 Contoh Citra Segmentasi K-Means

No	Nama File	Citra RGB	Thresholding	Segmentasi K-Means
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			

Tabel 3.3 merupakan hasil contoh hasil segmentasi dengan algoritma K-Means.

3.6 Region Of Interest

Pencarian *Region Of Interest* (ROI) pada objek citra daun tin dilakukan dengan cara mengkonversi menjadi citra *bw image* terlebih dahulu. Setelah diperoleh citra *bw image*, maka misalnya luas dan keliling dapat dihitung. Untuk menyempurnakan segmentasi hasil citra *bw image* tersebut selanjutnya dilakukan operasi *filling holes*, agar citra biner hasil segmentasi tepat merepresentasikan objek *region* daun tin. Selanjutnya dilakukan dengan segmentasi *K-Means* dan mensubstraksi sehingga didapatkan region penyakit citra daun tin.

Tabel 3.5 Contoh Citra *Region Of Interes*

No	Nama File	Citra RGB	Segmentasi <i>K-Means</i>	<i>Region Of Interes</i>
1	Sakit01.jpg			
2	Sakit04.jpg			
3	Sakit27.jpg			

Tabel 3.4 merupakan contoh hasil *Region of Interest* citra daun tin yang disegmentasi dengan algoritma *K-Means*

3.7 Data Set Pelatihan Dan Pengujian

Citra yang sudah dikumpulkan sebanyak 120 citra diantaranya daun tin terkena karat daun 47, daun tin terkena karat daun kutu kebul 4 citra terkena virus mosaik 5 citra dan daun tin sehat sebanyak 60 citra. Data set dibagi dalam 60% data pelatihan dan 40% data pengujian, yang terdiri dari 80 citra pelatihan dan 40 citra pengujian .Tabel 3.3. berikut contoh data set *testing* dapat dilihat pada tabel dibawah ini :

Tabel 3.6 Contoh Data Citra Pelatihan Daun Tin

No.	Nama File	Kondisi Citra	Citra Daun
1	Sakit01.jpg	Karat Daun	

No.	Nama File	Kondisi Citra	Citra Daun
2	Sakit02.jpg	Karat Daun	
3	Sakit03.jpg	Karat Daun	
4	Sakit04.jpg	Karat Daun	
5	Sakit05.jpg	Karat Daun	
6	Sakit06.jpg	Karat Daun	
7	Sakit07.jpg	Karat Daun	
8	Sakit08.jpg	Karat Daun	
9	Sakit09.jpg	Karat Daun	
10	Sakit27.jpg	Virus Mosaik	

Pada tabel 3.4 untuk citra pelatihan daun tin sebanyak 80, diantaranya 40 citra daun tin sehat, daun tin karat 34 citra, daun tin virus mosaik 4 citra dan daun tin kutu kebul 3 citra.

Tabel 3.7 Data Citra Pengujian Daun Tin

No.	Nama File	Kondisi Citra	Citra Daun
1	Sakit41.Jpg	Karat Daun	
2	Sakit42.Jpg	Karat Daun	
3	Sakit43.Jpg	Karat Daun	
4	Sakit44.Jpg	Karat Daun	
5	Sakit45.Jpg	Karat Daun+Kutu Kebul	
6	Sakit46.Jpg	Karat Daun	
7	Sakit47.Jpg	Karat Daun	
8	Sakit48.Jpg	Karat Daun	
9	Sakit49.Jpg	Karat Daun	
10	Sakit50.Jpg	Karat Daun	

Pada tabel 3.4 untuk citra pengujian diambil sebanyak 40 citra daun tin, yang terdiri dari citra daun tin karat daun 18 citra, daun tin virus mosaik 1 citra, daun tin kutu kebul berjumlah 1 citra dan daun tin sehat 20 citra.

3.8 Algoritma Naïve Bayes Dan Algoritma Decision Tree (C4.5)

Tahapan Pelatihan dan Pengujian dengan klasifikasi Algoritma C4.5 dan Algoritma Naïve Bayes pada pengolahan ekstraksi fitur seperti *contrast*, *correlation*, *energy*, dan *homogeneity* didapatkan hasil akurasi pengujian dan pelatihan, dengan klasifikasi algoritma c4.5 dan algoritma *naïve bayes* seperti jumlah data, jumlah data benar, jumlah data salah dan hasil akurasi dari pelatihan dan pengujian pada ekstraksi fitur GLCM akan didapatkan klasifikasi terbaik dari algoritma c4.5 dan algoritma *naïve bayes*.

3.9 Ekstraksi Fitur GLCM

Tahap ekstraksi fitur pada dataset pelatihan dan pengujian menggunakan metode GLCM (*GrayLevel Cooccurrence Matrix*) seperti pada nilai *contras*, *correlation*, *energy*, dan *homogeneity* pada citra daun tin dalam data citra Pelatihan dan citra daun tin Pengujian dengan klasifikasi Algoritma C4.5 dan Algoritma *Naïve Bayes*. Sehingga didapatkan hasil akurasi pelatihan dan pengujian citra daun tin seperti pada akurasi pelatihan dan pengujian terdapat jumlah data, jumlah data benar, jumlah data salah dan hasil akurasi Algoritma C4.5 dan Algoritma *Naïve Bayes*. pada ekstraksi fitur GLCM pengujian dan pelatihan. Sehingga terbentuklah *database* pelatihan dan pengujian keakurasian terbaik hasil klasifikasi algoritma C4.5 dan *Naïve Bayes*.

3.10 Evaluasi dan Validasi Hasil

Hasil pengolahan ekstraksi fitur GLCM yang diklasifikasikan dengan Algoritma C4.5 dan Algoritma *Naïve Bayes*, dengan pembagian kelas pelatihan dan pengujian didapatkan hasil nilai *Accuracy* pelatihan, pengujian .

3.11 Identifikasi Citra

Identifikasi citra daun tin sakit dan tidak sakit dilakukan dengan melakukan operasi komplemen agar objek citra daun dan melakukan operasi morfologi untuk menyempurnakan hasil segmentasi mengubah nilai *background* pada citra daun tin *Grayscale* menjadi nol sehingga diperoleh *region of interest* (ROI). Melakukan subtraksi untuk mendapatkan area penyakit pada daun tin dengan menandai area penyakit dan membentuk matriks *Co-oCurensi* disinilah didapatkan hasil identifikasi citra daun tin didapatkan seperti identifikasi sakit.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab hasil dan pembahasan ini merupakan tahapan yang telah dilakukan peneliti dalam proses segmentasi citra daun tin sehat dan sakit dengan menggunakan algoritma *Naive Bayes*.

4.1 Digitalisasi Citra

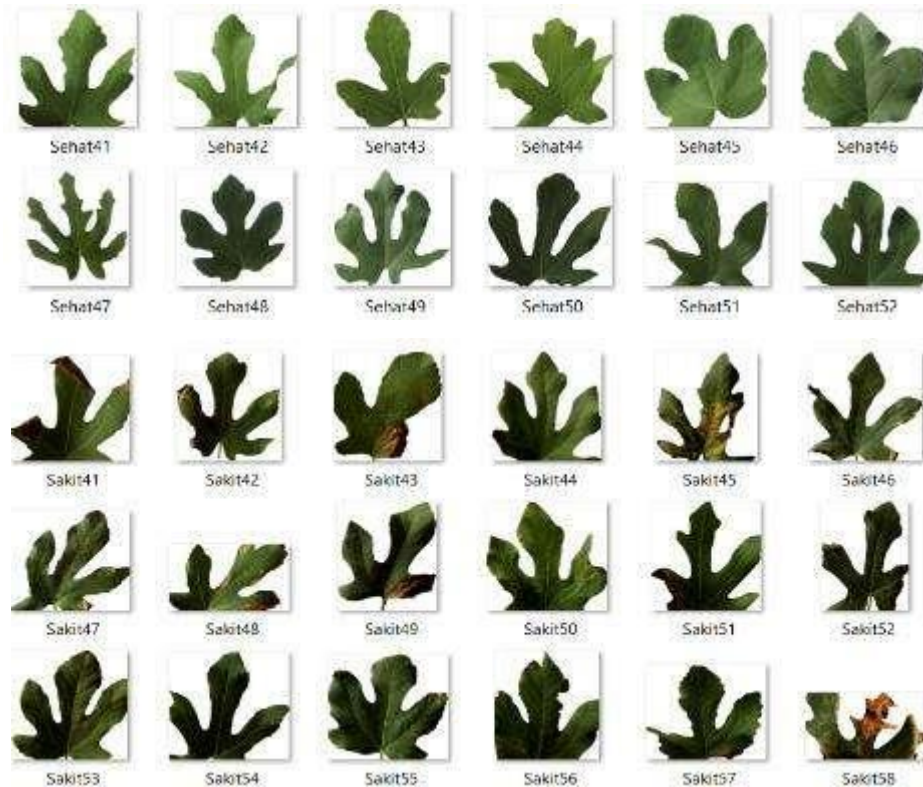
Tahapan ini merupakan proses digitalisasi citra daun tin yang dilakukan dengan menggunakan kamera *Canon Eos 1200D* dihubungkan ke dalam laptop guna memastikan bahwa citra berada di posisi yang tepat, seperti terlihat pada Gambar 4.1.



Sumber: hasil penelitian(2020)

Gambar 4.1 Proses Akuisisi Citra Daun Tin

Hasil dari digitalisasi citra tersebut, di dapatkan data set daun tin sejumlah 120 citra yang terdiri 60 citra daun tin sehat dan 60 citra daun tin sakit.



Sumber: hasil penelitian(2020)

Gambar 4.2 Data Set Citra Daun Tin Sehat dan Sakit

Selanjutnya dari 120 citra data set tersebut nantinya akan dibagi menjadi 60% untuk data pelatihan (80 citra pelatihan) dan 40% untuk data pengujian (40 citra pengujian).

4.2 *Resize*

Untuk mempercepat waktu proses pengolahan citra daun tin, maka dilakukan perubahan ukuran citra. Citra daun tin akan dilakukan *resize* 10% dari ukuran citra aslinya. Citra yang sebelumnya berukuran 2872x2016 diubah menjadi 288x202. Berikut Tabel 4.1 merupakan contoh citra dengan ukuran awal dan setelah dilakukan proses *resize*.

Tabel 4.1 Hasil *Resize* Citra

No.	Citra Daun Tin Sehat dan Sakit	Ukuran Citra Asli	Citra Resize
1		2872x2016	288x202
2		3080x1672	308x168
3		3000x2216	300x222
4		2368x1808	237x181
5		3272x2832	328x284
6		2264x2368	227x237
7		2712x2400	272x240
8		5184x3456	519x346
9		2008x2136	201x214
10		5184x3456	519x46

4.3 Bw Image

Pada tahap *bw image*, citra daun tin diubah ke dalam citra *bw image*. Hasil konversi citra *bw image* dapat dilihat pada tabel 4.2 sebagai berikut:

Tabel 4.2 Hasil Konversi Citra Bw Image

No	Nama File	Citra RGB	Bw Image
1	Sehat01.jpg		
2	Sehat02.jpg		
3	Sehat03.jpg		
4	Sehat04.jpg		
5	Sehat05.jpg		
6	Sakit01.jpg		
7	Sakit02.jpg		
8	Sakit07.jpg		
9	Sakit39.jpg		

10	Sakit27.jpg		

4.4 Segmentasi *Thresholding*

Pada tahap Segmentasi *Thresholding* citra daun tin yang sebelumnya sudah dilakukan perubahan menjadi citra bw *image* kemudian dilakukan pemisahan antara objek dengan *background*. Dapat dilihat pada Tabel 4.3 citra hasil *Thresolding* lebih dominan hitam. Berikut hasil segmentasi *Thresholding* :

Tabel 4.3 Hasil Segmentasi *Thresholding*

No	Nama File	Citra RGB	Bw Image	Thresholding
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			
4	Sehat04.jpg			
5	Sehat05.jpg			
6	Sakit01.jpg			

7	Sakit02.jpg			
8	Sakit07.jpg			
9	Sakit39.jpg			
10	Sakit27.jpg			

4.5 Segmentasi Algoritma *K-Means*

Pada tahap ini dilakukan proses klustering citra daun tin dengan cara mengkonversi ruang warna citra menjadi ruang warna L^*a^*b . Selanjutnya komponen a dan b dari citra L^*a^*b digunakan sebagai nilai masukan dalam Algoritma *K-Means*. Hasil segmentasi algoritma *K-Means* dapat dilihat pada tabel 4.4 sebagai berikut:

Tabel 4.4 Hasil Segmentasi *K-Means*

No	Nama File	Citra RGB	Bw Image	Thresholding
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			

4	Sehat04.jpg			
5	Sehat05.jpg			
6	Sakit01.jpg			
7	Sakit02.jpg			
8	Sakit07.jpg			
9	Sakit39.jpg			
10	Sakit27.jpg			

4.6 Region Of Interest

Setelah segmentasi dengan Algoritma *K-Means*, untuk menandai area citra daun tin sehat dan sakit maka dilakukan *Region Of Interest*. Hasil dari *Region Of Interest* seperti terlihat pada tabel 4.5 dibawah ini selanjutnya akan dihitung nilai ekstrasi cirinya dengan menggunakan GLCM (*Gray Level Co-Occurrence Matrix*).

Tabel 4.5 Hasil *Region Of Interest*

No	Nama File	Thresholding	Segmentasi K-Means	Region Of Interest
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			
4	Sehat04.jpg			
5	Sehat05.jpg			
6	Sakit01.jpg			
7	Sakit02.jpg			
8	Sakit07.jpg			
9	Sakit39.jpg			

10	Sakit27.jpg			
----	-------------	---	--	---

4.7 Ekstraksi Fitur *Gray Level Co-Occurrence Matrix*

Nilai dari ekstraksi fitur *Gray Level Co-Occurrence Matrix* merupakan nilai-nilai yang didapatkan dari metode yang dilatihkan dan diujikan pada 120 citra daun tin sehat dan sakit. Dengan komposisi data citra pelatihan daun tin sehat dan sakit sebesar 60%, sedangkan untuk citra daun tin sehat dan sakit data citra pengujian sebesar 40%. Pada penelitian ini digunakan analisis ciri *Gray Level Co-Occurrence Matrix* seperti : *contrast, correlation, energy dan homogeneity*.

Selanjutnya hasil dari ekstraksi fitur *Gray Level Co-Occurrence Matrix* pada data pelatihan dan pengujian dari citra daun tin sehat dan sakit diklasifikasikan menggunakan Algoritma C4.5 dan Algoritma *Naïve Bayes* guna mendapatkan hasil akurasi pelatihan dan pengujian untuk kedua algoritma tersebut.

4.8 Hasil Pelatihan Citra Daun Tin

Pada tahapan ini merupakan hasil dari pelatihan citra daun tin dengan menggunakan beberapa algoritma pengujian. Berikut hasil pelatihan:

4.8.1 Hasil Pelatihan Klasifikasi Algoritma C4.5

Hasil pelatihan 80 citra daun tin pada ekstraksi ciri *Gray Level Co-Occurrence Matrix* dengan klasifikasi Algoritma C4.5 ditemukan data benar sejumlah 73 citra dan data salah sejumlah 7 citra dengan akurasi sebesar 90%. Hasil akurasi pelatihan dengan klasifikasi Algoritma C4.5 citra daun tin dapat dilihat pada Gambar 4.3.

Section C4.5

Parameter Model: C4.5, Laplace, Laplace Smoothing

Membuat Folder Citra Daun Tin

Sakit

Sehat

Preview

Training

Stopper Model

Test

Pelatihan Ekstraksi Ciri GLCM Dengan Klasifikasi Algoritma C4.5

	Nama File	Contrast	Contrast	Energy	Homogeneity	Klasifikasi	Kelas Sebenarnya
1	Sakit01.jpg	0.24693	0.03792	0.11673	0.93963	Sakit	Sakit
2	Sakit02.jpg	0.20955	0.06387	0.23962	0.92119	Sakit	Sakit
3	Sakit03.jpg	0.2221	0.07623	0.30175	0.93590	Sakit	Sakit
4	Sakit04.jpg	0.25595	0.04798	0.39752	0.9305	Sakit	Sakit
5	Sakit05.jpg	0.3384	0.05499	0.28941	0.92469	Sakit	Sakit
6	Sakit06.jpg	0.27979	0.07930	0.28945	0.93504	Sakit	Sakit
7	Sakit07.jpg	0.37632	0.06898	0.23202	0.918	Sakit	Sakit
8	Sakit08.jpg	0.32282	0.02152	0.3927	0.93067	Sakit	Sakit
9	Sakit09.jpg	0.21873	0.9151	0.19745	0.95115	Sakit	Sakit
10	Sakit10.jpg	0.27483	0.84208	0.25719	0.92881	Sakit	Sakit
11	Sakit11.jpg	0.36621	0.17064	0.09805	0.91759	Sakit	Sakit
12	Sakit12.jpg	0.25209	0.07565	0.31589	0.94096	Sakit	Sakit
13	Sakit13.jpg	0.24199	0.03884	0.39200	0.91602	Sakit	Sakit
14	Sakit14.jpg	0.19226	0.01037	0.37551	0.93805	Sakit	Sakit

Algoritma Pelatihan

Jumlah Data Total: 14

Jumlah Data Sehat: 7

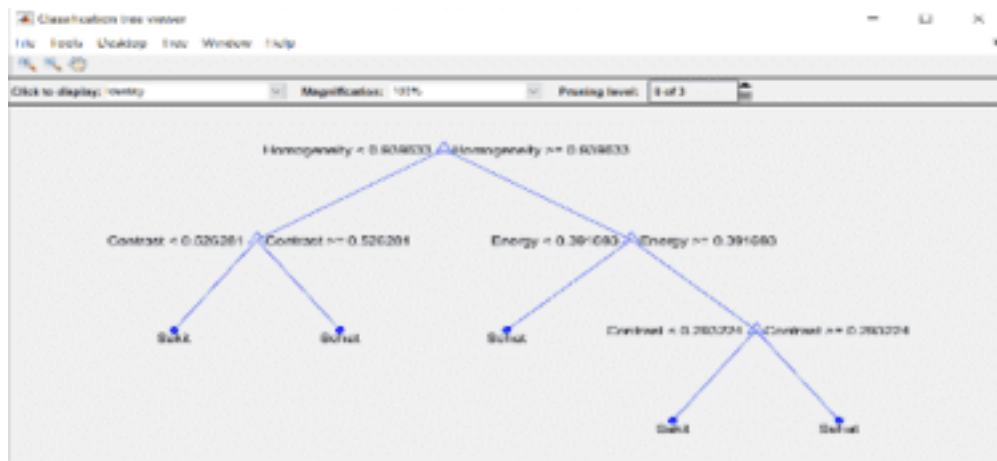
Jumlah Data Sakit: 7

Akurasi: 90%

Sumber: hasil penelitian(2020)

Gambar 4.3 Hasil Klasifikasi Algoritma C4.5

4.8.2 Hasil Klasifikasi *Decision Tree*



Sumber: hasil penelitian(2020)

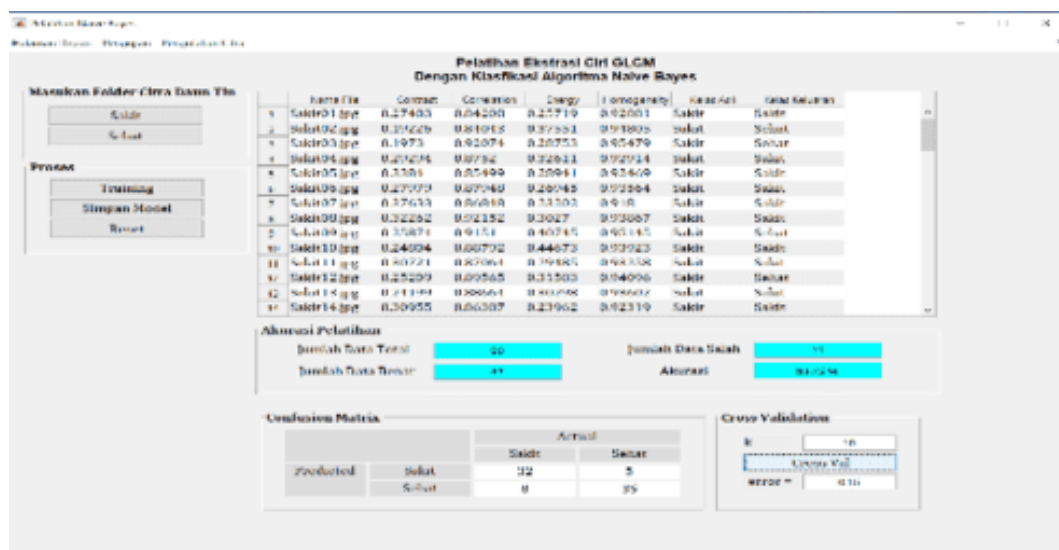
Gambar 4.4 Klasifikasi *Decision Tree*

Pada Gambar 4.4 diatas menggambarkan alur dari pemrosesan *classification tree*. Jika nilai *homogeneity* kurang dari 0.939633 dan nilai *contrast* kurang dari 0.526281 maka citra daun tin akan dikatakan sakit. Jika nilai *homogeneity* kurang dari 0.939633 dan nilai *contrast* lebih besar atau sama dengan 0.526281 maka citra daun tin akan dikatakan sehat. Jika *homogeneity* lebih besar sama dengan 0.939633 dan nilai *energy* kurang dari 0.391693 maka citra daun tin akan dikatakan sehat. Jika *homogeneity* lebih besar sama dengan 0.939633 dan nilai *energy* lebih lebih besar sama dengan 0.391693 maka citra daun tin akan dikatakan

sehat. Jika *homogeneity* lebih besar sama dengan 0.939633 dan nilai *contrast* kurang dari 0.293224 maka citra daun tin akan dikatakan sakit. Dan jika *homogeneity* lebih besar sama dengan 0.939633 dan nilai *contrast* lebih besar sama dengan 0.293224 maka citra daun tin dikatakan sehat.

4.8.3 Hasil Pelatihan Klasifikasi Algoritma Naïve Bayes

Hasil pelatihan 80 citra daun tin pada ekstraksi ciri *Gray Level Co-Occurrence Matrix* dengan klasifikasi Algoritma Naïve Bayes ditemukan data benar sejumlah 67 citra dan data salah sejumlah 13 citra. Akurasi klasifikasi dengan Algoritma Naïve Bayes didapatkan sebesar 83.75%. pada menu hasil confusion matrix didapatkan hasil prediksi hasil daun tin sakit terdeteksi sehat sebanyak 5 citra daun dan pada daun tin sehat terdeteksi sakit sebanyak 8 citra daun. Hasil yang didapatkan pada cross validation dengan pengujian jumlah K-fold 10 didapatkan hasil tingkat error sebesar 0,1875. Berikut hasil pelatihan dengan klasifikasi Algoritma Naïve Bayes pada Gambar 4.5 dibawah. Berikut hasil pelatihan dengan klasifikasi Algoritma Naïve Bayes pada Gambar 4.5.



Sumber: hasil penelitian(2020)

Gambar 4.5 Hasil Klasifikasi Algoritma Naïve Bayes

4.9 Hasil Pengujian Citra Daun Tin Sehat Dan Sakit

Pada tahapan ini merupakan hasil dari pengujian citra daun tin dengan menggunakan beberapa algoritma pengujian. Berikut hasil pengujian:

4.9.1 Hasil Pengujian Klasifikasi Algoritma C4.5

Hasil pengujian 40 citra daun tin pada ekstraksi ciri *Gray Level Co-Occurrence Matrix* dengan klasifikasi Algoritma C4.5 ditemukan data benar sejumlah 35 citra dan data salah sejumlah 5 citra. Akurasi klasifikasi dengan Algoritma C4.5 didapatkan sebesar 87.5%. Hasil akurasi pengujian dengan klasifikasi Algoritma C4.5 dapat dilihat pada Gambar 4.6.

No. File	File	Entropy	Contrast	Energy	Homogeneity	Kelas Pred	Kelas Sebenarnya
1	SN0001.jpg	0.22642	0.20611	0.30962	0.41670	Sakit	Sakit
2	SN0017.jpg	0.19438	0.36109	0.59115	0.95517	Sakit	Sakit
3	SN0041.jpg	0.21714	0.775	0.27705	0.92712	Sakit	Sakit
4	SN0049.jpg	0.2542	0.55192	0.3343	0.93172	Sakit	Sakit
5	SN0016.jpg	0.21695	0.39195	0.39095	0.92952	Sakit	Sakit
6	SN0047.jpg	0.25049	0.5611	0.32441	0.9304	Sakit	Sakit
7	SN0008.jpg	0.26141	0.8070	0.70144	0.91219	Sakit	Sakit
8	SN0042.jpg	0.23793	0.63322	0.40612	0.91392	Sakit	Sakit
9	SN0050.jpg	0.24039	0.57477	0.3111	0.93412	Sakit	Sakit
10	SN0001.jpg	0.2149	0.30912	0.32254	0.90102	Sakit	Sakit
11	SN0024.jpg	0.27151	0.67065	0.4477	0.91592	Sakit	Sakit
12	SN0013.jpg	0.27542	0.52111	0.3422	0.92271	Sakit	Sakit
13	SN0001.jpg	0.25940	0.71596	0.38714	0.90545	Sakit	Sakit
14	SN0053.jpg	0.24751	0.66159	0.31127	0.92705	Sakit	Sakit

Jumlah Data Total	40	Jumlah Data Salah	5
Jumlah Data Benar	35	Akurasi	87.5%

Sumber: hasil penelitian(2020)

Gambar 4.6 Hasil Pengujian Klasifikasi Algoritma C4.5

4.9.2 Hasil Pengujian Klasifikasi Naïve Bayes

Hasil pengujian 40 citra daun tin pada ekstraksi ciri *Gray Level Co-Occurrence Matrix* ditemukan data benar sejumlah 36 citra dan data salah sejumlah 4 citra. Untuk hasil akurasi klasifikasi dengan Algoritma C4.5 pada citra daun tin didapatkan sebesar 90%. Hasil akurasi pengujian dengan klasifikasi Algoritma *Naïve Bayes* citra daun tin dapat dilihat pada Gambar 4.7.

No. File	File	Entropy	Contrast	Energy	Homogeneity	Kelas Pred	Kelas Sebenarnya
1	SN0001.jpg	0.22642	0.20611	0.30962	0.41670	Sakit	Sakit
2	SN0017.jpg	0.19438	0.36109	0.59115	0.95517	Sakit	Sakit
3	SN0041.jpg	0.21714	0.775	0.27705	0.92712	Sakit	Sakit
4	SN0049.jpg	0.2542	0.55192	0.3343	0.93172	Sakit	Sakit
5	SN0016.jpg	0.21695	0.39195	0.39095	0.92952	Sakit	Sakit
6	SN0047.jpg	0.25049	0.5611	0.32441	0.9304	Sakit	Sakit
7	SN0008.jpg	0.26141	0.8070	0.70144	0.91219	Sakit	Sakit
8	SN0042.jpg	0.23793	0.63322	0.40612	0.91392	Sakit	Sakit
9	SN0050.jpg	0.24039	0.57477	0.3111	0.93412	Sakit	Sakit
10	SN0001.jpg	0.2149	0.30912	0.32254	0.90102	Sakit	Sakit

Jumlah Data Total	40	Jumlah Data Salah	4
Jumlah Data Benar	36	Akurasi	90%

Predicted \ Actual	Actual	
	Sakit	Sehat
Sakit	36	2
Sehat	2	10

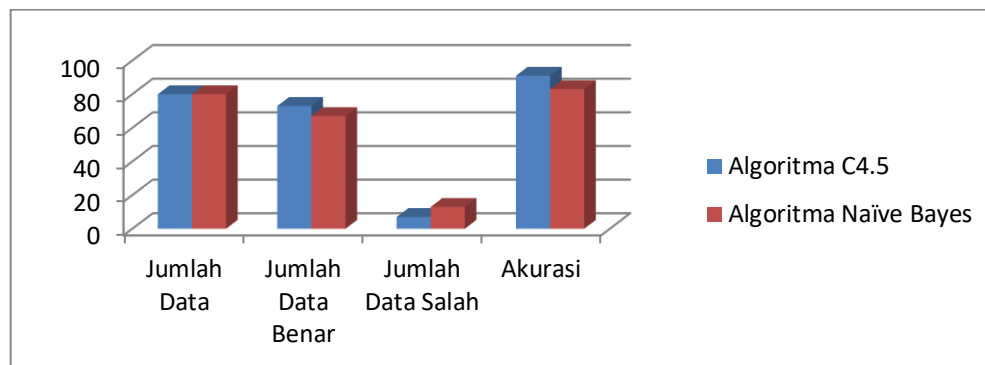
Sumber: hasil penelitian(2020)

Gambar 4.7 Hasil Pengujian Klasifikasi Algoritma Naïve Bayes

4.10 Hasil Komparasi Algoritma C4.5 Dan Algoritma *Naïve Bayes*

Tabel 4.6 Hasil Pelatihan Algoritma C4.5 Dan Algoritma *Naïve Bayes*

Hasil Klasifikasi Pelatihan Citra Daun Tin							
Algoritma C4.5				Algoritma <i>Naïve Bayes</i>			
Kelas Asli	Kelas Keluaran	Jumlah Data Salah	Akurasi	Kelas Asli	Kelas Keluaran	Jumlah Data Salah	Akurasi
80	73	7	90%	80	67	13	83.75%

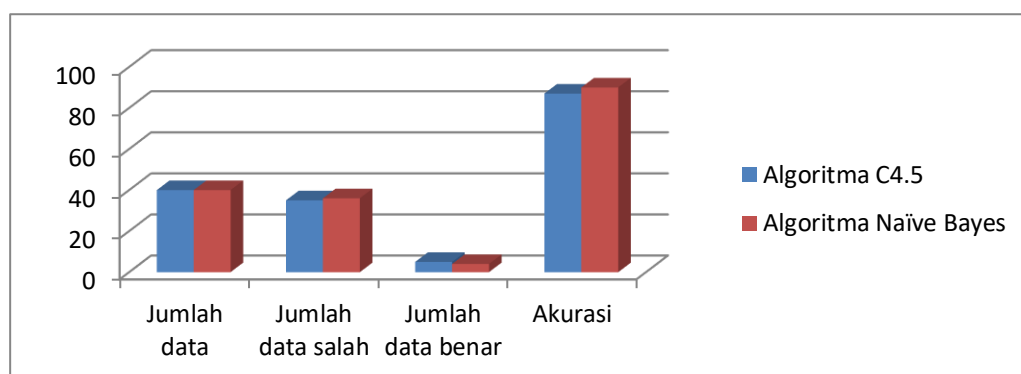


Gambar 4.8 Grafik Pelatihan Algoritma C4.5 dan Algoritma *Naïve Bayes*

Hasil pelatihan citra daun tin pada klasifikasi Algoritma C4.5 dan Algoritma *Naïve Bayes*, didapatkan nilai akurasi tertinggi pada algoritma C4.5 sebesar 91.25%. Sedangkan pada pelatihan Algoritma *Naïve Bayes* citra akurasi yang didapatkan 83.75%.

Tabel 4.7 Hasil Pengujian Algoritma C4.5 dan Algoritma *Naïve Bayes*

Hasil Klasifikasi Pengujian Citra Daun Tin							
Algoritma C4.5				Algoritma <i>Naïve Bayes</i>			
Kelas Asli	Kelas Keluaran	Jumlah Data Salah	Akurasi	Kelas Asli	Kelas Keluaran	Jumlah Data Salah	Akurasi
40	35	5	87.5%	40	36	4	90%



Gambar 4.9 Grafik Pengujian Algoritma C4.5 dan Algoritma *Naïve Bayes*

Hasil pengujian citra daun tin pada klasifikasi algoritma C4.5 dan algoritma *Naïve Bayes* didapatkan nilai akurasi tertinggi pada algoritma *Naïve Bayes* sebesar 90%, sedangkan pada pengujian algoritma C4.5 citra akurasi yang didapatkan 87.5%.

4.11 Implementasi Dan Hasil

Berikut merupakan hasil implementasi penelitian yang telah dibuat dalam bentuk GUI Matlab.



Sumber: hasil penelitian(2020)

Gambar 4.10 Halaman Depan Gui Matlab

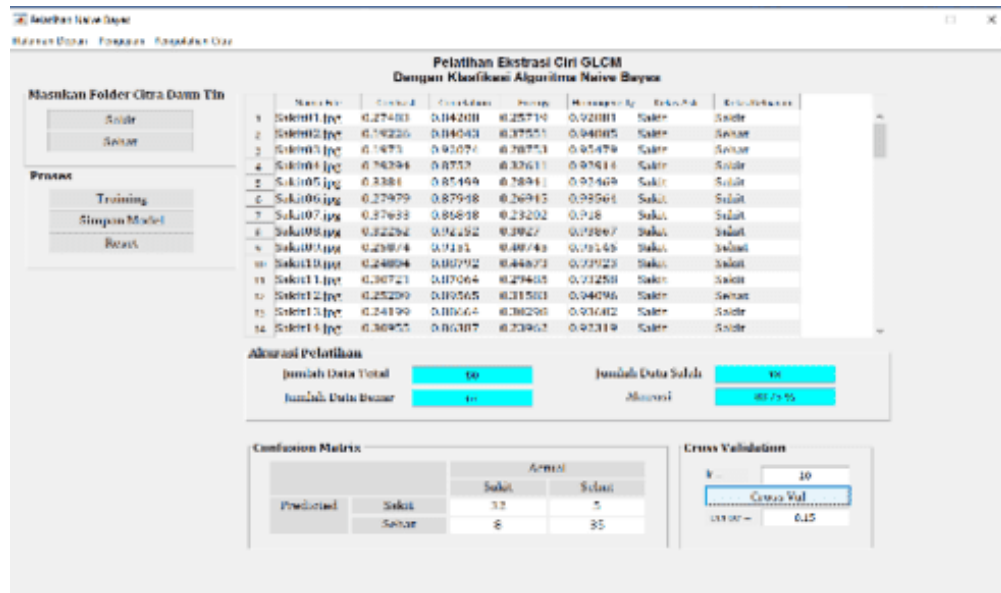


Sumber: hasil penelitian(2020)

Gambar 4.11 Menu dari Algoritma C4.5 Dan Algoritma *Naïve Bayes*

Pada Gambar 4.11 merupakan tampilan menu dari Algoritma C4.5 dan Algoritma *Naïve Bayes*. Menu pelatihan dan pengujian digunakan untuk

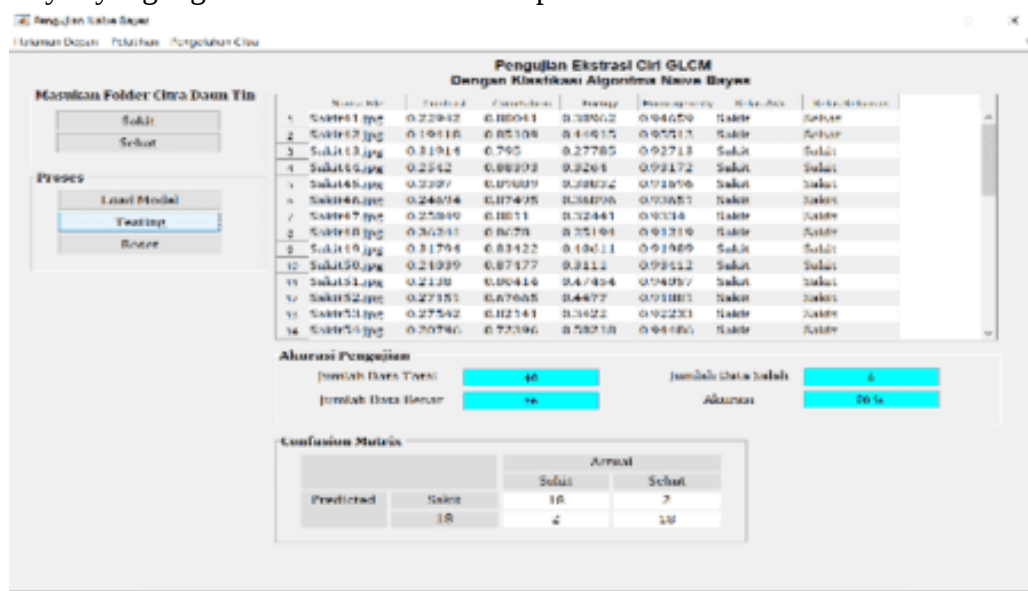
mengekstraksi ciri *Gray Level Co-Occurrence Matrix* dengan klasifikasi Algoritma C4.5 dan Algoritma *Naïve Bayes*. Pada menu pengolahan citra digunakan untuk melakukan pengolahan citra daun tin.



Sumber: hasil penelitian(2020)

Gambar 4.12 Menu Pelatihan Algoritma Naïve Bayes

Pada Gambar 4.12 merupakan tampilan menu pelatihan algoritma Naïve Bayes yang digunakan untuk melakukan pelatihan citra daun tin.



Sumber: hasil penelitian(2020)

Gambar 4.13 Pengujian Klasifikasi Algoritma Naïve Bayes

Pada Gambar 4.13 tampilan menu pengujian terdapat menu masukan folder citra daun tin, lalu pada menu proses terdapat *load* model digunakan untuk memasukkan hasil pelatihan untuk dilakukan pengujian citra daun tin.



Sumber: hasil penelitian(2020)

Gambar 4.14 Hasil Identifikasi Citra Daun Tin Sakit

Salah satu contoh hasil identifikasi citra daun tin yang teridentifikasi sakit dapat dilihat pada gambar 4.14 diatas dimana hasil dari ekstrasi ciri nilai *homogeneity* sebesar 0.90559 mengacu pada hasil *classification tree viewer (decision)* jika kurang dari 0.939633 maka citra dinyatakan sakit.

BAB V

KESIMPULAN DAN SARAN

Dalam bagian ini kesimpulan yang diambil berdasarkan hasil dari penelitian disajikan dengan padat dan singkat, serta pada bagian akhir akan dikemukakan saran yang dapat dijadikan masukan untuk penelitian selanjutnya.

5.1. Kesimpulan

Bertitik tolak dari pembahasan yang telah dijelaskan pada bab sebelumnya, dapat ditarik kesimpulan beberapa poin sebagai berikut:

1. Dengan *Region Of Interest* dapat digunakan untuk mengoptimisasi area pada citra daun tin menggunakan segmentasi *K-Means clustering*.
2. Perbandingan antara algoritma C4.5 dan algoritma *Naïve Bayes* dengan cara mencari akurasi terbaik berhasil didapatkan melalui metode dari nilai GLCM dan didapatkan akurasi terbaik pada pengujian citra daun Tin adalah algoritma *Naïve Bayes* dengan nilai akurasi sebesar 90% dibandingkan algoritma C4.5 hanya 87.5% saja.
3. Rule dari pohon keputusan *decision tree classifier* berhasil diimplementasikan pada GUI Matlab guna mencari klaster citra daun tin sehat dan sakit dengan tepat.

5.2. Saran

Berikut merupakan beberapa saran yang diharapkan dapat memberikan perbaikan pada penelitian selanjutnya.

1. Dalam segmentasi citra daun tin sehat dan sakit menggunakan Algoritma C4.5 dan algoritma *Naïve Bayes*, ada beberapa citra yang tidak dapat dilakukan segmentasi dengan sempurna, sehingga pada penelitian selanjutnya diharapkan seluruh citra yang diujikan dapat tersegmentasi semua dengan baik.
2. Dataset daun tin sehat dan sakit yang digunakan yang merupakan hasil digitalisasi atas tanaman tin sendiri terbatas. Diharapkan pada penelitian selanjutnya dapat memperkaya dengan memperbanyak dataset meliputi seluruh jenis penyakit yang ada pada daun tin.
3. Melakukan ujicoba dengan menggunakan metode lain dalam melakukan segmentasi dan identifikasi sehingga mendapatkan akurasi yang lebih baik lagi.

DAFTAR PUSTAKA

- [1] R. Purnamasari, D. Winarni, A. A. Permanasari, E. Agustina, S. Hayaza, and W. Darmanto, "Anticancer Activity of Methanol Extract of Ficus carica Leaves and Fruits Against Proliferation, Apoptosis, and Necrosis in Huh7it Cells," *Cancer Inform.*, vol. 18, pp. 0–6, 2019, doi: 10.1177/1176935119842576.
- [2] Y. Zhang, Y. Wan, B. Huo, B. Li, Y. Jin, and X. Hu, "Extracts and components of ficus carica leaves suppress survival, cell cycle, and migration of triple-negative breast cancer MDA-MB-231 cells," *Onco. Targets. Ther.*, vol. 11, pp. 4377–4386, 2018, doi: 10.2147/OTT.S171601.
- [3] J. Kamas, M. Nesbitt, and L. Stein, "Texas Fruit and Nut Production," pp. 1–7.
- [4] J. P. Gaikwad, "Region of Interest of Diseased Leaf using Overlapping Window Technique," vol. 10, no. 1, pp. 865–868, 2017.
- [5] S. Hadiani and D. Riana, "Segmentation and Image Analysis for Image Microscopic Pap Smear," vol. xx, no. x, 2017.
- [6] V. K. Mishra, S. Kumar, and N. Shukla, "Image Acquisition and Techniques to Perform Image Acquisition," *SAMRIDDHI A J. Phys. Sci. Eng. Technol.*, vol. 9, no. 01, pp. 3–6, 2017, doi: 10.18090/samriddhi.v9i01.8333.
- [7] F. R. Lestari, J. Y. Sari, Sutardi, and I. Purwanti, "Deteksi Penyakit Tanaman Jeruk Siam Berdasarkan Citra Daun," no. December, pp. 276–283, 2018.
- [8] F. G. Febrinanto, C. Dewi, and A. T. Wiratno, "Implementasi Algoritme K-Means Sebagai Metode Segmentasi Citra Dalam Identifikasi Penyakit Daun Jeruk," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 11, pp. 5375–5383, 2018.
- [9] Eko Prasetyo, "Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab," F. S. Suyantoro, Ed. Yogyakarta: Andi Yogyakarta, 2011, p. 404.
- [10] S. Informasi, U. B. Insani, P. Diplodia, E. Fitur, N. Network, and D. Tree, "Deteksi Penyakit Pada Tanaman Mangga Dengan Citra Digital : Tinjauan Literatur Sistematis (SLR)," vol. 7, no. 1, pp. 63–72, 2020.
- [11] S. Zahrah, R. Saptono, and E. Suryani, "Identifikasi Gejala Penyakit Padi Menggunakan Operasi Morfologi Citra," no. Snik, pp. 100–106, 2016.
- [12] R. V. Nahari *et al.*, "Cow Weight Estimation Using Local Adaptive Thresholding Method And Connected Component Labelling," vol. 1, no. Icst, pp. 148–152, 2018, doi: 10.2991/icst-18.2018.32.
- [13] E. Prasetyo and A. M. Matlab, "Bab 8 SEGMENTASI E. Prasetyo,

Pengolahan Citra digital dan Aplikasinya Menggunakan Matlab,
Yogyakarta: Penerbit ANDI, 2011,” 2011.

- [14] W. F. Mahmudy, M. A. Rahman, J. Matematika, and U. Brawijaya, “Segmentasi Citra Digital Menggunakan Metode Adaptive Split-and-Merge yang Dimodifikasi,” no. 2, pp. 127–137.
- [15] Ulla Delfana Rosiani, Cahya Rahmad, Marcelina Alifia Rahmawati, and Frangky Tupamahu, “Segmentasi Berbasis K-Means Pada Deteksi Citra Penyakit Daun Tanaman Jagung,” *J. Inform. Polinema*, vol. 6, no. 3, pp. 37–42, 2020, doi: 10.33795/jip.v6i3.331.
- [16] S. S. FI, Ed., “BAB 6 Pengolahan Citra digital dan Aplikasinya Menggunakan Matlab,” Yogyakarta: CV.Andi Offset, 2011, p. 155.
- [17] A. Mukminin and D. Riana, “Komparasi Algoritma C4 . 5 , Naïve Bayes Dan Neural Network Untuk Klasifikasi Tanah,” *J. Inform.*, vol. 4, no. 1, pp. 21–31, 2017.
- [18] E. Kamilah, R. Venantius, H. Ginardi, and C. Fatichah, “Klasifikasi penyakit noda pada citra daun tebu berdasarkan ciri tekstur dan warna menggunakan segmentation-based gray level cooccurrence matrix dan LAB color moments,” vol. 3, no. 1, pp. 1–10, 2017.
- [19] S. C. Madiwalar and M. V. Wyawahare, “Plant disease identification: A comparative study,” *2017 Int. Conf. Data Manag. Anal. Innov. ICDMAI 2017*, pp. 13–18, 2017, doi: 10.1109/ICDMAI.2017.8073478.
- [20] E. Prasetyo and A. M. Matlab, “BAB 9Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab,” *Pengolahan.*, vol. 26, FI. Sigit Suyantoro, Ed. Yogyakarta: CV.Andi Offset, 2011, pp. 283–284.



LEMBAR BIMBINGAN TESIS

STMIK NUSA MANDIRI

NIM : 14002209
Nama Lengkap : Muhammad Qomaruddin
Dosen Pembimbing I : Dr. Dwiza Riana, S.Si, MM, M.Kom
Judul Tesis : "Segmentasi K-Means Citra Daun Tin Dengan Klasifikasi Ekstraksi Ciri GLCM"

No	Tanggal Bimbingan	Pokok Bahasan	Paraf Dosen Pembimbing
1.	1 April 2020	Pembahasan Paper Rujukan	<i>Dwiza Riana</i>
2.	9 Mei 2020	Pembahasan Objek Penelitian Via Zoom	<i>Dwiza Riana</i>
3.	27 Juni 2020	Pembahasan Metodologi Penelitian	<i>Dwiza Riana</i>
4.	12 Juli 2020	Pengajuan Bab I dan Bab III	<i>Dwiza Riana</i>
5.	24 Juli 2020	Pengajuan Bab II, III dan IV	<i>Dwiza Riana</i>
6.	29 Juli 2020	ACC Bab II dan III, dan Pengajuan Bab IV	<i>Dwiza Riana</i>
7.	5 Agustus	Acc Bab IV dan Pengajuan Bab V	<i>Dwiza Riana</i>
8.	6 Agustus 2020	Acc Keseluruhan	<i>Dwiza Riana</i>

Catatan untuk Dosen Pembimbing.

Bimbingan Tugas Akhir

- Dimulai pada tanggal : 1 April 2020
- Diakhiri pada tanggal : 6 Agustus 2020
- Jumlah pertemuan bimbingan : 8 Kali Pertemuan

Disetujui oleh,
Dosen Pembimbing I

(Dr. Dwiza Riana, S.Si, MM, M.Kom)



LEMBAR BIMBINGAN TESIS

STMIK NUSA MANDIRI

NIM : 14002209
Nama Lengkap : Muhammad Qomaruddin
Dosen Pembimbing II : Anton, M.Kom
Judul Tesis : "Segmentasi K-Means Citra Daun Tin Dengan Klasifikasi Ekstraksi Ciri GLCM"

No	Tanggal Bimbingan	Pokok Bahasan	Paraf Dosen Pembimbing
1.	1 April 2020	Pembahasan Paper Rujukan	
2.	9 Mei 2020	Pembahasan Objek Penelitian Via Zoom	
3.	27 Juni 2020	Pembahasan Metodologi Penelitian	
4.	12 Juli 2020	Pengajuan Bab I dan Bab III	
5.	24 Juli 2020	Pengajuan Bab II, III dan IV	
6.	29 Juli 2020	ACC Bab II dan III, dan Pengajuan Bab IV	
7.	5 Agustus	Acc Bab IV dan Pengajuan Bab V	
8.	6 Agustus 2020	Acc Keseluruhan	

Catatan untuk Dosen Pembimbing.

Bimbingan Tugas Akhir

- Dimulai pada tanggal : 1 April 2020
- Diakhiri pada tanggal : 6 Agustus 2020
- Jumlah pertemuan bimbingan : 8 Kali Pertemuan

Disetujui oleh,
Dosen Pembimbing II

(Anton, M.Kom)

DAFTAR RIWAYAT HIDUP

A. Biodata Mahasiswa :

Nim : 14002209
Nama : Muhammad Qomaruddin
Tempat Tanggal Lahir : Pekalongan, 04 April 1982
Alamat : Jl. Melati II Blok I Rt.002/011 No.383
Kel. Jatimulya Kec.Tambun Selatan
Bekasi Timur
Hp. : 081513125359
Email : 14002209@nusamandiri.ac.id

B. Riwayat Pendidikan Formal

1. SD Negeri Rawa Tembaga I, Perumnas I Bekasi 1994
2. SMP Islam Al-Kholidin, Jakarta lulus tahun 1998
3. SMU Hasyim Asy'ari, lulus tahun 2001
4. Diploma 3 (D3) Akademik Bina Sarana Informatika, lulus tahun 2012
5. Strata 1 (S1) STMIK Nusa Mandiri, lulus tahun 2018
6. Pascasarjana (S2) STMIK Nusa Mandiri Jakarta 2020

C. Riwayat Pendidikan Non Formal

1. Kursus Komputer LPK Suadata Paket Windows 98 tahun 2002
2. Kursus Komputer LPK Suadata Paket Teknisi 2002

D. Riwayat Pekerjaan

1. Staff BTI Technical Support UBSI Group 2014 – Sekarang
2. Magang BTI Technical Support UBSI Group 2010 - 2014
3. Mengajar LPK Komputer Suadata 2003 – 2005
4. Mangajar LPK Komputer BBC 2006 – 2007



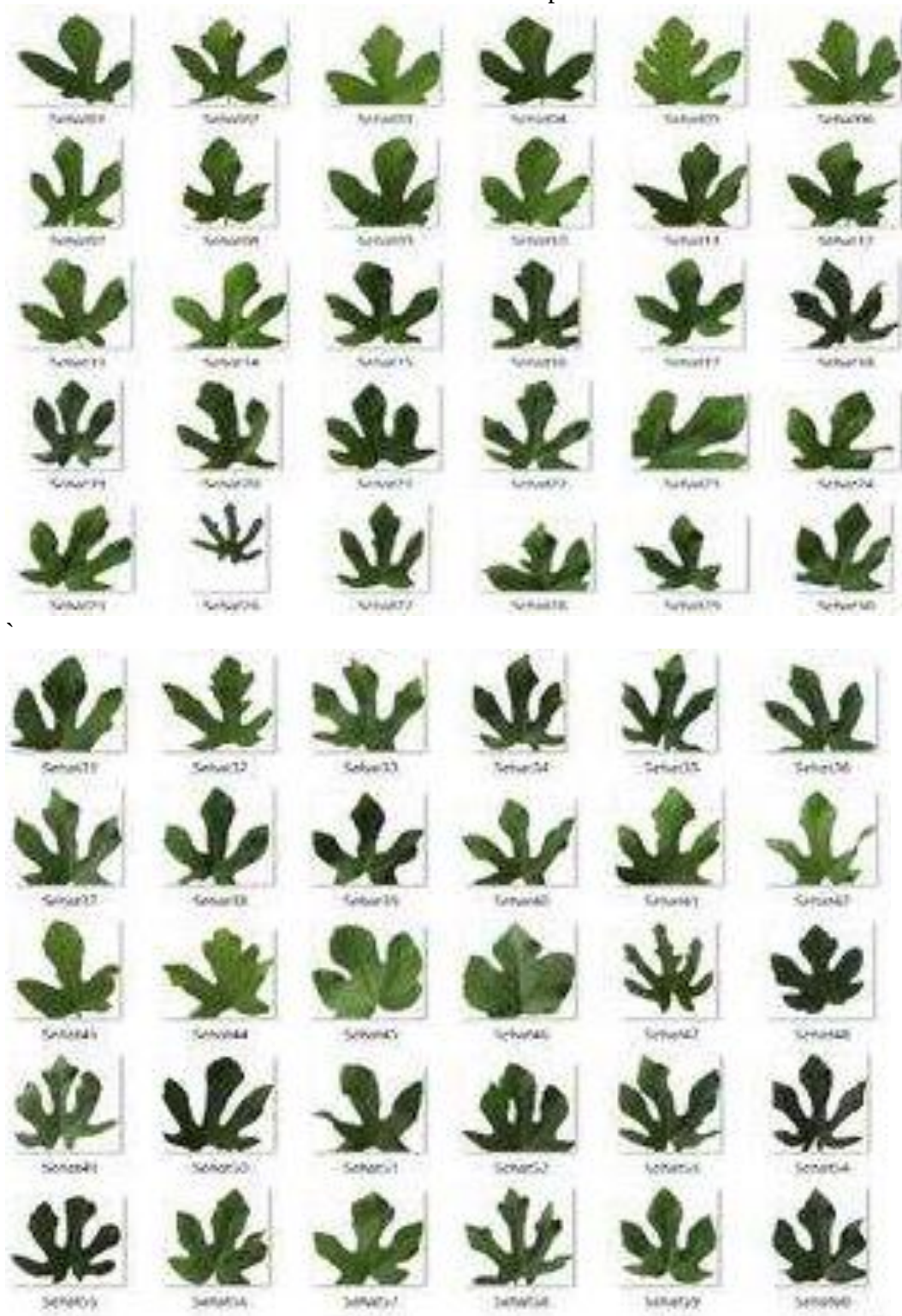
Jakarta, 06 Agustus 2020



Muhammad Qomaruddin

LAMPIRAN-LAMPIRAN

Lampiran 1. Data Set Daun Tin Sehat



Lampiran 2. Data Set Citra Daun Tin Sakit



Lampiran 3. Gambar Halaman Depan Gui Matlab



Lampiran 4. Gambar Pelatihan Klasifikasi *Naïve Bayes*

Perhitungan Naïve Bayes

File | View | Options | Help | Pengaturan Data

Pelatihan Ekstraksi Ciri GLCM Dengan Klasifikasi Algoritma Naïve Bayes

Masukan Folder Citra Dari Tin

Sakit
Sehat

Proses

Uyotung
Simpan Model
Reset

No	Nama File	Contrast	Correlation	Energy	Homogeneity	Kelas Asli	Kelas Prediksi
1	Sakit01.jpg	0.27189	0.03200	0.17719	0.92991	Sakit	Sakit
2	Sakit02.jpg	0.19226	0.04043	0.37551	0.94005	Sakit	Sakit
3	Sakit03.jpg	0.1973	0.92074	0.28758	0.95179	Sakit	Sakit
4	Sakit04.jpg	0.29296	0.0752	0.32011	0.92914	Sakit	Sakit
5	Sakit05.jpg	0.3384	0.05499	0.20541	0.92469	Sakit	Sakit
6	Sakit06.jpg	0.27979	0.07930	0.20942	0.93561	Sakit	Sakit
7	Sakit07.jpg	0.37633	0.06040	0.23202	0.910	Sakit	Sakit
8	Sakit08.jpg	0.32262	0.92152	0.3027	0.93867	Sakit	Sakit
9	Sakit09.jpg	0.25071	0.9151	0.40745	0.95115	Sakit	Sakit
10	Sakit10.jpg	0.24804	0.00792	0.44673	0.93923	Sakit	Sakit
11	Sakit11.jpg	0.30721	0.07061	0.29403	0.93258	Sakit	Sakit
12	Sakit12.jpg	0.25209	0.09565	0.31503	0.94096	Sakit	Sakit
13	Sakit13.jpg	0.24199	0.88664	0.30298	0.93602	Sakit	Sakit
14	Sakit14.jpg	0.30955	0.06307	0.23902	0.92319	Sakit	Sakit

Akurasi Pelatihan

Jumlah Data Total: 14
Jumlah Data Benar: 13
Jumlah Data Salah: 1
Akurasi: 93.75%

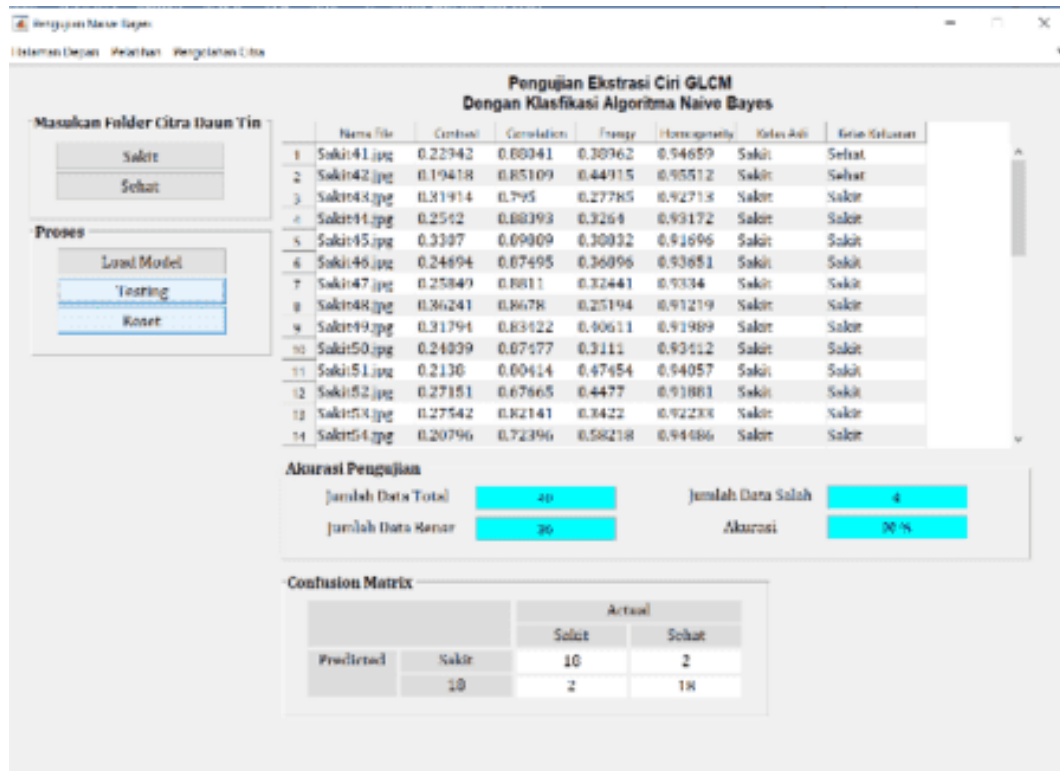
Confusion Matrix

		Actual	
		Sakit	Sehat
Predicted	Sakit	12	5
	Sehat	0	35

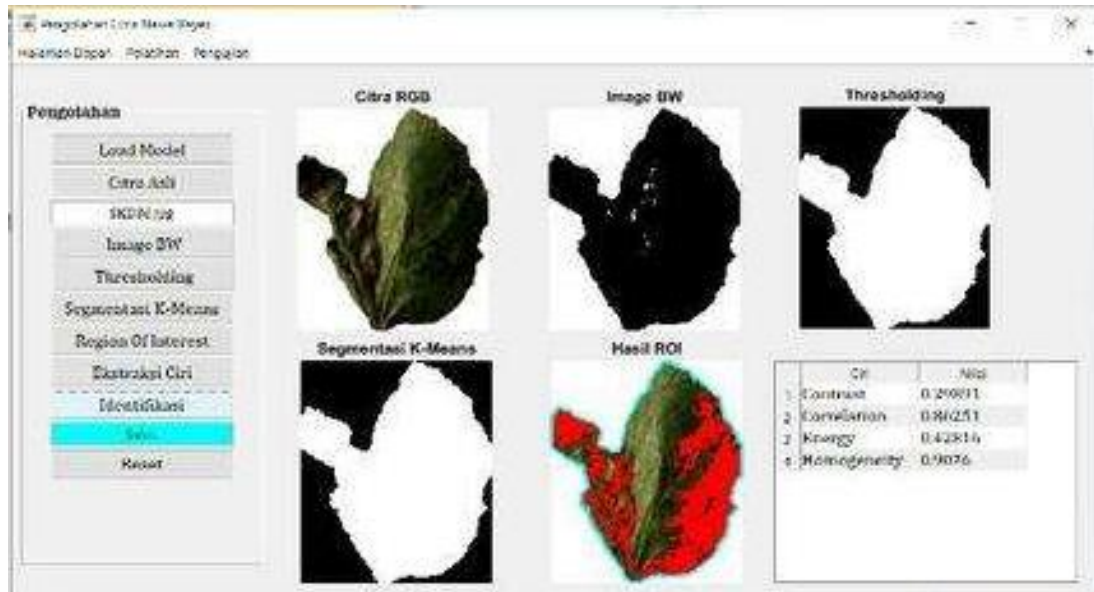
Cross Validation

k = 10
Cross Val
error = 0.2

Lampiran 5. Gambar Pengujian Klasifikasi Naïve Bayes



Lampiran 6. Gambar Pengolahan Citra *Naïve Bayes*



Lampiran 7. Tabel Citra Daun Tin RGB Dan Bw Image

No	Nama File	Citra RGB	<i>Bw Image</i>
1	Sehat01.jpg		
2	Sehat02.jpg		
3	Sehat03.jpg		
4	Sehat04.jpg		
5	Sehat05.jpg		
6	Sakit01.jpg		
7	Sakit02.jpg		
8	Sakit07.jpg		
9	Sakit39.jpg		
10	Sakit27.jpg		

Lampiran 8. Tabel Citra Hasil *Thresholding*

No	Nama File	Citra RGB	Bw Image	Thresholding
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			
4	Sehat04.jpg			
5	Sehat05.jpg			
6	Sakit01.jpg			
7	Sakit02.jpg			
8	Sakit07.jpg			
9	Sakit39.jpg			
10	Sakit27.jpg			

Lampiran 9. Tabel Citra Hasil *K-Means*

No	Nama File	Bw Image	Thresholding	Segmengasi <i>K-Means</i>
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			
4	Sehat04.jpg			
5	Sehat05.jpg			
6	Sakit01.jpg			
7	Sakit02.jpg			
8	Sakit07.jpg			
9	Sakit39.jpg			
10	Sakit27.jpg			

Lampiran 10. Hasil Region Of Interest

No	Nama File	Thresholding	Segmentasi K-Means	Region Of Interest
1	Sehat01.jpg			
2	Sehat02.jpg			
3	Sehat03.jpg			
4	Sehat04.jpg			
5	Sehat05.jpg			
6	Sakit01.jpg			
7	Sakit02.jpg			
8	Sakit07.jpg			
9	Sakit39.jpg			
10	Sakit27.jpg			

A. Mengambil Folder Citra Daun Tin Pelatihan Folder 1

```
% untuk semua file yg dibaca
for n = 1:jumlah_file1
    % membaca file citra
    Img = imread(fullfile(nama_folder1,
nama_file1(n).name));
    % meresize ukuran citra menjadi 10%
    ukuran semula
    Img_rsz = imresize(Img,0.1);
    % mengkonversi citra rgb menjadi
    grayscale
    Img_gray = rgb2gray(Img_rsz);
    % melakukan segmentasi cira
    bw =
    im2bw(Img_gray,graythresh(Img_gray));
    % melakukan operasi komplemen agar
    objek berwarna putih (1) dan background
    % berwarna hitam (0)
    bw = imcomplement(bw);
    % melakukan operasi morfologi untuk
    menyempurnakan hasil segmentasi
    bw = imfill(bw,'holes');
    % mengubah nilai background pada citra
    grayscale menjadi nol sehingga
    % diperoleh region of interest (ROI)
    Img_gray(~bw) = 0;
    % Segmentasi menggunakan algoritma K-
    Means Clustering
    cform = makecform('srgb2lab');
    lab = applycform(Img_rsz,cform);
    ab = double(lab(:, :, 2:3));
    nrows = size(ab,1);
    ncols = size(ab,2);
    ab = reshape(ab,nrows*ncols,2);
    nColors = 2;
    [cluster_idx, ~] =
    kmeans(ab,nColors,'distance','sqEuclidean', ...
    'Replicates',3);
    pixel_labels =
    reshape(cluster_idx,nrows,ncols);
    segmented_images = cell(1,3);
    rgb_label = repmat(pixel_labels,[1 1
3]);
```

```

meanRGB = zeros(2,1);

for k = 1:nColors
    color = Img_rsz;
    color(rgb_label ~= k) = 0;
    segmented_images{k} = color;
    R = color(:, :, 1);
    G = color(:, :, 2);
    B = color(:, :, 3);
    meanR = mean2(R);
    meanG = mean2(G);
    meanB = mean2(B);
    meanRGB(k) =
mean([meanR, meanG, meanB]);
end

[~, cluster] = max(meanRGB);
area_cluster = (pixel_labels==cluster);
I_gray = rgb2gray(Img_rsz);
bw = im2bw(I_gray);
bw = imcomplement(bw);
bw = imfill(bw, 'holes');
bw = bwareaopen(bw, 50);

% melakukan subtraksi untuk mendapatkan
region penyakit pada daun
Img_subtract =
logical(imsubtract(bw, area_cluster));
Img_subtract2 =
imsubtract(bw, Img_subtract);
Img_subtract2 = Img_subtract2>0;
Img_subtract2 =
bwareaopen(Img_subtract2, 50);

% menandai region penyakit dengan warna
merah
R = Img_rsz(:, :, 1);
G = Img_rsz(:, :, 2);
B = Img_rsz(:, :, 3);
R(Img_subtract2) = 255;
G(Img_subtract2) = 0;
B(Img_subtract2) = 0;
RGB = cat(3, R, G, B);
% membentuk matriks kookurensi
pixel_dist = 1;

```

```

        GLCM =
graycomatrix(Img_gray,'Offset',...
        [0 pixel_dist; -pixel_dist
pixel_dist;...
        -pixel_dist 0; -pixel_dist -
pixel_dist]);
        % menghitung ciri glcm
        stats = graycoprops(GLCM);
        Contrast = mean(stats.Contrast);
% Contrast
        Correlation = mean(stats.Correlation);
% Correlation
        Energy = mean(stats.Energy);
% Energy
        Homogeneity = mean(stats.Homogeneity);
% Homogeneity
        % mengisi nilai2 pada data_sakit
        data_sakit{n,1} = nama_file1(n).name;
        data_sakit{n,2} = num2str(Contrast);
        data_sakit{n,3} = num2str(Correlation);
        data_sakit{n,4} = num2str(Energy);
        data_sakit{n,5} = num2str(Homogeneity);
        data_sakit{n,6} = 'Sakit';
        % mengisi nilai2 pada ciri_sakit
        ciri_sakit(n,1) = Contrast;
        ciri_sakit(n,2) = Correlation;
        ciri_sakit(n,3) = Energy;
        ciri_sakit(n,4) = Homogeneity;
        % mengisi nilai pada target_sakit
        target_sakit{n} = 'Sakit';
end

% menampilkan data_daun_tin pada tabel
data_daun_tin = data_sakit;

set(handles.uitable1,'Data',data_daun_tin,'RowName',1:size(data_daun_tin,1))

% menyimpan variabel2 pada lokasi handles
(lokasi penyimpanan variabel)
% agar dapat dipanggil oleh pushbutton yg
lain
handles.data_sakit = data_sakit;
handles.ciri_sakit = ciri_sakit;
handles.target_sakit = target_sakit;

```



```

        % meresize ukuran citra menjadi 10%
ukuran semula
        Img_rsz = imresize(Img,0.1);
        % mengkonversi citra rgb menjadi
grayscale
        Img_gray = rgb2gray(Img_rsz);
        % melakukan segmentasi cira
        bw =
im2bw(Img_gray,graythresh(Img_gray));
        % melakukan operasi komplemen agar
objek berwarna putih (1) dan background
        % berwarna hitam (0)
        bw = imcomplement(bw);
        % melakukan operasi morfologi untuk
menyempurnakan hasil segmentasi
        bw = imfill(bw,'holes');
        % mengubah nilai background pada citra
grayscale menjadi nol sehingga
        % diperoleh region of interest (ROI)
        Img_gray(~bw) = 0;
        % Segmentasi menggunakan algoritma K-
Means Clustering
        cform = makecform('srgb2lab');
        lab = applycform(Img_rsz,cform);
        ab = double(lab(:, :, 2:3));
        nrows = size(ab,1);
        ncols = size(ab,2);
        ab = reshape(ab,nrows*ncols,2);
        nColors = 2;
        [cluster_idx, ~] =
kmeans(ab,nColors,'distance','sqEuclidean', ...
        'Replicates',3);
        pixel_labels =
reshape(cluster_idx,nrows,ncols);
        segmented_images = cell(1,3);
        rgb_label = repmat(pixel_labels,[1 1
3]);
        meanRGB = zeros(2,1);

for k = 1:nColors
        color = Img_rsz;
        color(rgb_label ~= k) = 0;
        segmented_images{k} = color;
        R = color(:, :, 1);
        G = color(:, :, 2);

```

```

        B = color(:,:,3);
        meanR = mean2(R);
        meanG = mean2(G);
        meanB = mean2(B);
        meanRGB(k) =
mean([meanR,meanG,meanB]);
    end

    [~,cluster] = max(meanRGB);
    area_cluster = (pixel_labels==cluster);
    I_gray = rgb2gray(Img_rsz);
    bw = im2bw(I_gray);
    bw = imcomplement(bw);
    bw = imfill(bw,'holes');
    bw = bwareaopen(bw,50);

    % melakukan subtraksi untuk mendapatkan
    region penyakit pada daun
    Img_subtract =
logical(imsubtract(bw,area_cluster));
    Img_subtract2 =
imsubtract(bw,Img_subtract);
    Img_subtract2 = Img_subtract2>0;
    Img_subtract2 =
bwareaopen(Img_subtract2,50);

    % menandai region penyakit dengan warna
    merah
    R = Img_rsz(:,:,1);
    G = Img_rsz(:,:,2);
    B = Img_rsz(:,:,3);
    R(Img_subtract2) = 255;
    G(Img_subtract2) = 0;
    B(Img_subtract2) = 0;
    RGB = cat(3,R,G,B);
    % membentuk matriks kookurensi
    pixel_dist = 1;
    GLCM =
graycomatrix(Img_gray,'Offset',...
    [0 pixel_dist; -pixel_dist
    pixel_dist;...
    -pixel_dist 0; -pixel_dist -
    pixel_dist]);
    % menghitung ciri glcm
    stats = graycoprops(GLCM);

```

```

        Contrast = mean(stats.Contrast);
% Contrast
        Correlation = mean(stats.Correlation);
% Correlation
        Energy = mean(stats.Energy);
% Energy
        Homogeneity = mean(stats.Homogeneity);
% Homogeneity
        % mengisi nilai2 pada data_sehat
        data_sehat{n,1} = nama_file2(n).name;
        data_sehat{n,2} = num2str(Contrast);
        data_sehat{n,3} = num2str(Correlation);
        data_sehat{n,4} = num2str(Energy);
        data_sehat{n,5} = num2str(Homogeneity);
        data_sehat{n,6} = 'Sehat';
        % mengisi nilai2 pada ciri_sehat
        ciri_sehat(n,1) = Contrast;
        ciri_sehat(n,2) = Correlation;
        ciri_sehat(n,3) = Energy;
        ciri_sehat(n,4) = Homogeneity;
        % mengisi nilai pada target_sehat
        target_sehat{n} = 'Sehat';
    end

    % menampilkan data_daun_tin pada tabel
    data_daun_tin =
[handles.data_sakit;data_sehat];

set(handles.uitable1,'Data',data_daun_tin,'RowName',1:size(data_daun_tin,1))

    % menyimpan variabel2 pada lokasi handles
    (lokasi penyimpanan variabel)
    % agar dapat dipanggil oleh pushbutton yg
    lain
    handles.data_sehat = data_sehat;
    handles.ciri_sehat = ciri_sehat;
    handles.target_sehat = target_sehat;
    guidata(hObject, handles);

    % mereset button2
    set(handles.pushbutton3,'Enable','on')
    set(handles.pushbutton4,'Enable','off')
else

```

```

        % jika tidak ada folder yg dipilih maka
        akan kembali
        return
    end

```

Menampilkan hasil ekstrasi ciri

```

% memanggil variabel2 yg ada pada lokasi
handles
ciri_sakit = handles.ciri_sakit;
ciri_sehat = handles.ciri_sehat;

target_sakit = handles.target_sakit;
target_sehat = handles.target_sehat;

% menyusun data latih dan target latih
data_latih = [ciri_sakit;ciri_sehat];
target_latih = [target_sakit;target_sehat];

```

Melakukan Training Algoritma Naïve Bayes

```

Mdl =
    fitcnb(data_latih,target_latih,'ClassNames',{'S
akit','Sehat'});

```

Membaca kelas keluaran berdasarkan Mdl yg telah dilatihkan

```

kelas_keluaran = predict(Mdl,data_latih);

```

Menghitung nilai2 akurasi

```

kelas_keluaran = predict(Mdl,data_latih);

% menghitung nilai2 akurasi
jumlah_data = numel(kelas_keluaran);
nilai_keluaran = zeros(jumlah_data,1);
jumlah_benar = 0;
for k = 1:jumlah_data
    if isequal(target_latih{k},kelas_keluaran{k})
        jumlah_benar = jumlah_benar+1;
    end
    if isequal(kelas_keluaran{k},'Sakit')
        nilai_keluaran(k) = 1;
    end
end
nilai_target = zeros(jumlah_data,1);
jumlah_salah = 0;
for k = 1:jumlah_data
    if ~isequal(target_latih{k},kelas_keluaran{k})

```

```

        jumlah_salah = jumlah_salah+1;
    end
    if isequal(target_latih{k}, 'Sakit')
        nilai_target(k) = 1;
    end
end
akurasi = (jumlah_benar/jumlah_data)*100;

% menampilkan nilai2 akurasi pada edit text
set(handles.edit1, 'String', jumlah_data)
set(handles.edit2, 'String', jumlah_benar)
set(handles.edit3, 'String', jumlah_salah)
set(handles.edit4, 'String', [num2str(akurasi), '
%'])

% menampilkan data_daun_tin pada tabel
data_daun_tin =
[handles.data_sakit;handles.data_sehat];
data_daun_tin = [data_daun_tin,kelas_keluaran];
set(handles.uitable1, 'Data', data_daun_tin, 'RowName
',1:size(data_daun_tin,1))

% menampilkan confusion matrix pada edit text
confmat =
confusionmat(nilai_target,nilai_keluaran);
set(handles.text16, 'String', confmat(2,2))
set(handles.text17, 'String', confmat(1,2))
set(handles.text18, 'String', confmat(2,1))
set(handles.text19, 'String', confmat(1,1))

```

Menampilkan nilai2 akurasi pada edit text

```

set(handles.edit1, 'String', jumlah_data)
set(handles.edit2, 'String', jumlah_benar)
set(handles.edit3, 'String', jumlah_salah)
set(handles.edit4, 'String', [num2str(akurasi), '
%'])

```

Menampilkan data hasil confusion matrix dan cross validation

```

confmat =
confusionmat(nilai_target,nilai_keluaran);
set(handles.text16, 'String', confmat(2,2))
set(handles.text17, 'String', confmat(1,2))
set(handles.text18, 'String', confmat(2,1))
set(handles.text19, 'String', confmat(1,1))

```


Memasukan Citra Model Pengolahan

```
[nama_file,nama_path] = uigetfile({'*.mat','MAT
File (*.mat)');
    '*..*','All file type (*.*)'},'Browse
File');

% jika ada file yg dipilih maka akan
mengeksekusi perintah yg ada di bawahnya
if ~isequal(nama_file,0)
    % mereset button2
    axes(handles.axes1)
    cla reset
    set(gca,'XTick',[])
    set(gca,'YTick',[])

    axes(handles.axes2)
    cla reset
    set(gca,'XTick',[])
    set(gca,'YTick',[])

    axes(handles.axes3)
    cla reset
    set(gca,'XTick',[])
    set(gca,'YTick',[])

    axes(handles.axes4)
    cla reset
    set(gca,'XTick',[])
    set(gca,'YTick',[])

set(handles.uitable1,'Data',[],'RowName',{' ' ' '
' ' ' '})
set(handles.edit1,'String','')
set(handles.edit2,'String','')
set(handles.pushbutton2,'Enable','on')
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton5,'Enable','off')

% menyimpan variabel nama_path & nama_file
pada lokasi handles agar dapat
```

```

        % dipanggil oleh pushbutton yg lain
        handles.nama_path = nama_path;
        handles.nama_file = nama_file;
        guidata(hObject,handles);
        set(handles.pushbutton2,'Enable','on')
    else
        % jika tidak ada file yg dipilih maka akan
        kembali
        return
    end

```

Menampilkan Citra RGB

```

% membaca file citra
Img = imread(fullfile(pathname,filename));

% memperkecil ukuran citra menjadi 0,1x
semula untuk
% mempercepat waktu proses
Img_rsz = imresize(Img,0.1);
% menampilkan citra pada axes
axes(handles.axes1)
imshow(Img_rsz)
title('Citra RGB')

```

Menampilkan Bw Image

```

set(handles.uitable1,'Data',[],'RowName',{' ' ' '
' ' ' '})
set(handles.edit2,'String','')
set(handles.pushbutton11,'Enable','on')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton5,'Enable','off')
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton12,'Enable','off')
% memanggil variabel Img_rsz yg ada pada lokasi
handles
Img_rsz = handles.Img_rsz;
% mengkonversi citra rgb menjadi grayscale
Img_gray = rgb2gray(Img_rsz);
% melakukan segmentasi cira
bw = im2bw(Img_gray,graythresh(Img_gray));
% melakukan operasi komplemen agar objek
berwarna putih (1) dan background

```

```

% berwarna hitam (0)
axes(handles.axes2)
imshow(bw)
title('Image BW')
handles.bw = bw;
guidata(hObject, handles)

```

Menampilkan Citra Thresholding

```

bw = handles.bw;
set(handles.pushbutton3, 'Enable', 'on')
bwt = imcomplement(bw);
bwtres = imfill(bwt, 'holes');
% menampilkan citra biner hasil segmentasi pada
axes
axes(handles.axes3)
imshow(bwtres)
title('Thresholding')

handles.bwtres = bwtres;
guidata(hObject, handles)

```

Menampilkan Citra K-Means

```

set(handles.pushbutton12, 'Enable', 'on')

% memanggil variabel Img_rsz yg ada pada lokasi
handles
Img_rsz = handles.Img_rsz
bwtres = handles.bwtres;
% mengubah nilai background pada citra
grayscale menjadi nol sehingga
% diperoleh region of interest (ROI)
Img_gray(~Img_rsz) = 0;
% Segmentasi menggunakan algoritma K-Means
Clustering
cform = makecform('srgb2lab');
lab = applycform(Img_rsz, cform);
ab = double(lab(:, :, 2:3));
nrows = size(ab, 1);
ncols = size(ab, 2);
ab = reshape(ab, nrows*ncols, 2);
nColors = 2;

```

```

[cluster_idx, ~] =
kmeans(ab,nColors,'distance','sqEuclidean', ...
'Replicates',3);
pixel_labels =
reshape(cluster_idx,nrows,ncols);
segmented_images = cell(1,3);
rgb_label = repmat(pixel_labels,[1 1 3]);
meanRGB = zeros(2,1);

for k = 1:nColors
    color = Img_rsz;
    color(rgb_label ~= k) = 0;
    segmented_images{k} = color;
    R = color(:, :, 1);
    G = color(:, :, 2);
    B = color(:, :, 3);
    meanR = mean2(R);
    meanG = mean2(G);
    meanB = mean2(B);
    meanRGB(k) = mean([meanR,meanG,meanB]);
end

[~,cluster] = max(meanRGB);
area_cluster = (pixel_labels==cluster);
I_gray = rgb2gray(Img_rsz);
bw = im2bw(I_gray);
bw = imcomplement(bw);
bw1 = imfill(bw,'holes');
bw1 = bwareaopen(bw1,50);
axes(handles.axes4)
imshow(bw1)
title('Segmentasi K-Means')

handles.bw1 = bw1;
guidata(hObject, handles)

```

Menampilkan Hasil ROI

```

bw1 = handles.bw1

% mereset button2
set(handles.pushbutton4,'Enable','on')

% memanggil variabel Img_rsz yg ada pada lokasi
handles
Img_rsz = handles.Img_rsz

```

```

bwtres = handles.bwtres;
% mengubah nilai background pada citra
grayscale menjadi nol sehingga
% diperoleh region of interest (ROI)
Img_gray = rgb2gray(Img_rsz);
Img_gray(~bw1) = 0;
% Segmentasi menggunakan algoritma K-Means
Clustering
cform = makecform('srgb2lab');
lab = applycform(Img_rsz,cform);
ab = double(lab(:,:,2:3));
nrows = size(ab,1);
ncols = size(ab,2);
ab = reshape(ab,nrows*ncols,2);
nColors = 2;
[cluster_idx, ~] =
kmeans(ab,nColors,'distance','sqEuclidean', ...
'Replicates',3);
pixel_labels =
reshape(cluster_idx,nrows,ncols);
segmented_images = cell(1,3);
rgb_label = repmat(pixel_labels,[1 1 3]);
meanRGB = zeros(2,1);

for k = 1:nColors
    color = Img_rsz;
    color(rgb_label ~= k) = 0;
    segmented_images{k} = color;
    R = color(:,:,1);
    G = color(:,:,2);
    B = color(:,:,3);
    meanR = mean2(R);
    meanG = mean2(G);
    meanB = mean2(B);
    meanRGB(k) = mean([meanR,meanG,meanB]);
end

[~,cluster] = max(meanRGB);
area_cluster = (pixel_labels==cluster);
I_gray = rgb2gray(Img_rsz);
bw = im2bw(I_gray);
bw = imcomplement(bw);
bw1 = imfill(bw,'holes');
bw1 = bwareaopen(bw1,50);

```

```

% melakukan subtraksi untuk mendapatkan region
penyakit pada daun
Img_subtract =
logical(imsubtract(bw1,area_cluster));
Img_subtract2 = imsubtract(bw1,Img_subtract);
Img_subtract2 = Img_subtract2>0;
Img_subtract2 = bwareaopen(Img_subtract2,50);

% menandai region penyakit dengan warna merah
R = Img_rsz(:, :,1);
G = Img_rsz(:, :,2);
B = Img_rsz(:, :,3);
R(Img_subtract2) = 255;
G(Img_subtract2) = 0;
B(Img_subtract2) = 0;
RGB = cat(3,R,G,B);

% menampilkan citra ROI hasil segmentasi pada
axes
axes(handles.axes8)
imshow(RGB)
title('Hasil ROI')
[B,~] = bwboundaries(bw1,'noholes');
hold on
for k = 1:length(B)
    boundary = B{k};
    plot(boundary(:,2), boundary(:,1), 'c',
'LineWidth', 2)
end
hold off

% menyimpan variabel2 pada lokasi handles agar
% dapat dipanggil oleh pushbutton yg lain
handles.Img_gray = Img_gray;
guidata(hObject, handles)

```

Menampilkan Ekstraksi Ciri GLCM

```

set(handles.uitable1,'Data',[],'RowName',{' ' ' '
' ' ' '})
set(handles.edit2,'String','')
set(handles.pushbutton5,'Enable','on')

% memanggil variabel2 yang ada di lokasi
handles

```

```

Img_gray = handles.Img_gray;

% membentuk matriks kookurensi
pixel_dist = 1;
GLCM = graycomatrix(Img_gray, 'Offset', ...
    [0 pixel_dist; -pixel_dist pixel_dist; ...
    -pixel_dist 0; -pixel_dist -pixel_dist]);
% menghitung ciri glcm
stats = graycoprops(GLCM);
Contrast = mean(stats.Contrast); %
Contrast
Correlation = mean(stats.Correlation); %
Correlation
Energy = mean(stats.Energy); %
Energy
Homogeneity = mean(stats.Homogeneity); %
Homogeneity
% mengisi nilai2 pada data_daun_tin
data_daun_tin = cell(4,2);
data_daun_tin{1,1} = 'Contrast';
data_daun_tin{2,1} = 'Correlation';
data_daun_tin{3,1} = 'Energy';
data_daun_tin{4,1} = 'Homogeneity';
data_daun_tin{1,2} = num2str(Contrast);
data_daun_tin{2,2} = num2str(Correlation);
data_daun_tin{3,2} = num2str(Energy);
data_daun_tin{4,2} = num2str(Homogeneity);

% mengisi nilai2 pada ciri_daun_tin
ciri_daun_tin = zeros(1,4);
ciri_daun_tin(1,1) = Contrast;
ciri_daun_tin(1,2) = Correlation;
ciri_daun_tin(1,3) = Energy;
ciri_daun_tin(1,4) = Homogeneity;

% menampilkan hasil ekstraksi ciri pada tabel
set(handles.uitable1, 'Data', data_daun_tin, 'RowName', 1:4)

% menyimpan variabel2 pada lokasi handles
(lokasi penyimpanan variabel)
% agar dapat dipanggil oleh pushbutton yg lain
handles.ciri_daun_tin = ciri_daun_tin;
guidata(hObject, handles)

% --- Executes on button press in pushbutton5.

```

```

function pushbutton5_Callback(hObject,
 eventdata, handles)
% hObject      handle to pushbutton5 (see GCBO)
% eventdata    reserved - to be defined in a
future version of MATLAB
% handles      structure with handles and user
data (see GUIDATA)

% memanggil variabel2 yang ada di lokasi
handles
nama_path = handles.nama_path;
nama_file = handles.nama_file;
ciri_daun_tin = handles.ciri_daun_tin;

% memanggil Mdl hasil pelatihan
load(fullfile(nama_path,nama_file))

% membaca kelas keluaran berdasarkan Mdl yg
telah dilatihkan
kelas_keluaran = predict(Mdl,ciri_daun_tin);

% menampilkan kelas keluaran pada edit text
set(handles.edit2,'String',kelas_keluaran)

```