

**Pengenalan Tingkat Kematangan Buah
Mangga Menggunakan Metode *GLCM* dan
*KNN***

Oleh

MOHAMMAD REZKY B. PAKAYA

T3116101

SKRIPSI

**Untuk memenuhi salah satu syarat ujian
guna memperoleh gelar Sarjana**



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS ICHSAN GORONTALO
2020**

**PENGENALAN TINGKAT KEMATANGAN BUAH
MANGGA MENGGUNAKAN METODE *GLCM* DAN
*KNN***

Oleh

MOHAMMAD REZKY B. PAKAYA

T3116101

SKRIPSI

**Untuk memenuhi salah satu syarat ujian
guna memperoleh gelar Sarjana**



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS ICHSAN GORONTALO
2020**

PENGESAHAN SKRIPSI

**Pengenalan Tingkat Kematangan Buah
Mangga Menggunakan Metode GLCM dan
KNN**

Oleh

MOHAMMAD REZKY B. PAKAYA

T3116101

SKRIPSI

Untuk memenuhi salah satu syarat ujian guna memperoleh gelar Sarjana Program
Studi Teknik Informatika, ini telah disetujui oleh Tim Pembimbing

Gorontalo, 3 Desember 2020

Pembimbing I



Rezqiwati Ishak, M.Kom

NIDN.0903087901

Pembimbing II



Sudirman S. Panna, M.Kom

NIDN.9909912852

PERSETUJUAN SKRIPSI

Pengenalan Tingkat Kematangan Buah Mangga Menggunakan Metode *GLCM* dan *KNN*

Oleh

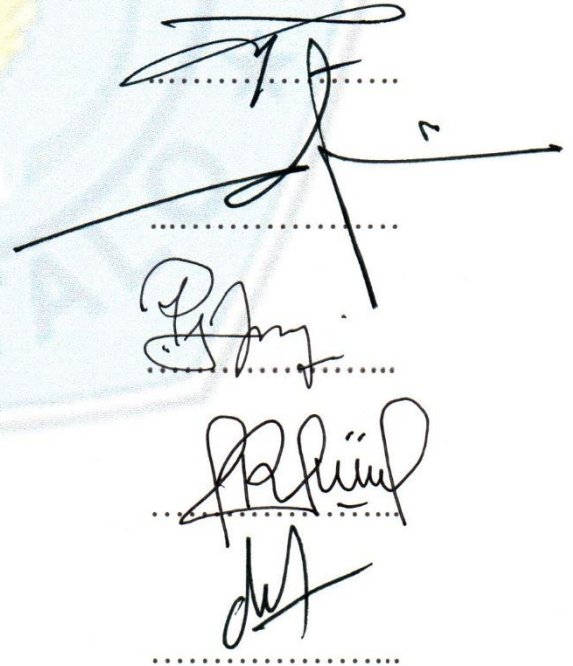
MOHAMMAD REZKY B. PAKAYA

T3116101

Di Periksa oleh Panitia Ujian Strata Satu (S1)

Universitas Ichsan Gorontalo

1. Ketua Penguji
Zohrahayaty, M.Kom
2. Anggota Penguji
Budy Santoso, S.Kom, M.Eng
3. Anggota Penguji
Kartika Chandra Pelangi, M.Kom
4. Anggota Penguji
Hj. Rezqiwati Ishak, M.Kom
5. Anggota Penguji
Sudirman S.Panna, M.Kom



PERNYATAAN SKRIPSI

Dengan ini saya menyatakan bahwa :

1. Karya tulis (Skripsi) saya ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di perguruan tinggi lainnya.
2. Karya tulis (Skripsi) saya ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali dari arahan Tim Pembimbing.
3. Dalam karya tulis (Skripsi) saya ini tidak terdapat karya atau pendapat yang telah dipublikasi orang lain, kecuali serta tertulis dicantumkan sebagai acuan/ sitasi dalam naskah dan dicantumkan pula dalam daftar pustaka.
4. Pernyataan isi saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidak benaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini serta sanksi lainnya sesuai dengan norma-norma yang berlaku di Universitas Ichsan Gorontalo.

Gorontalo, 3 Desember 2020

Yang Membuat Pernyataan



Mohammad Rezky B. Pakaya

ABSTRACT

The introduction of mango ripeness levels in this study was divided into four groups, namely Raw, Mengkal, Cooked, and Rot. This study aims to classify the ripeness of the fruit based on the level of accuracy using the feature extraction method, namely the gray level co-occurrence matrix. The research method consists of: conversion of rgb data to grayscale, image normalization, mango ripeness using color textures, feature extraction and classification. In this study, 80 mango fruit images were used as training data, which are real data. The data consists of 20 raw mangoes, 20 mangoes, 20 ripe mangoes, and 20 rotten mangoes. The image data is processed into a grayscale image, then the mango ripeness level is carried out using a color texture. After the maturity level is obtained, the characteristic calculation is carried out using the gray level co-occurrence matrix. The algorithm used for the classification process is the k-nearest neighbor algorithm. The final results of the test show that the proposed method has been able to recognize the level of maturity of the mango fruit using a color texture with accuracy results calculated using the confusion matrix, it can be concluded that in the K-NN model experiment, with the number of $K = 1$, and the GLCM direction = 0o, with distance = 1, then get the highest accuracy value of = 36.66%, with training data = 80, and Data Testing = 30. Whereas in the K-NN model experiment, with the number of $K = 2$, and GLCM direction = 135o, with distance = 2, then get the highest accuracy value of = 50%, with training data = 80, and Data Testing = 30. In the K-NN model experiment, with the number of $K = 7$, and the direction of GLCM = 135o, with distance = 5, then get the highest accuracy value of = 83.33%, with training data = 80, and Data Testing = 30. Thus the application of the gray level co-occurrence matrix and k-nearest neighbor method on the problem of mango ripeness level uses the w texture. arna needs to be developed again accurate results.

Keywords: Maturity Level, Gray Level Co-Occurrence Matrix, K-Nearest Neighbor

ABSTRAK

Pengenalan tingkat kematangan buah mangga dalam penelitian ini dibagi menjadi empat kelompok yaitu Mentah, Mangkal, Matang, dan Busuk. Penelitian ini bertujuan untuk mengklasifikasi kematangan buah berdasarkan tingkat akurasi dengan menggunakan metode fitur ekstraksi yaitu gray level co-occurrence matrix. Metode penelitian ini terdiri dari : konversi data rgb ke grayscale, normalisasi citra, tingkat kematangan buah mangga menggunakan tekstur warna, ekstraksi fitur dan klasifikasi. Dalam penelitian ini digunakan data latih sebanyak 80 citra buah mangga yang merupakan data real. Data tersebut terdiri dari buah mangga mentah 20, mangga mangkal 20, mangga matang 20, dan mangga busuk 20. Data citra tersebut diolah menjadi citra grayscale yang kemudian dilakukan tingkat kematangan buah mangga menggunakan tekstur warna. Setelah didapat tingkat kematangannya, kemudian dilakukan perhitungan ciri menggunakan gray level co-occurrence matrix. Algoritma yang digunakan untuk proses klasifikasi adalah algoritma k-nearest neighbor. Hasil akhir pengujian menunjukkan bahwa metode yang diusulkan telah mampu mengenali tingkat kematangan buah mangga menggunakan tekstur warna dengan hasil akurasi yang dihitung menggunakan confusion matrix, maka dapat ditarik suatu kesimpulan bahwa Pada percobaan model K-NN, dengan jumlah $K = 1$, dan Arah GLCM = 0° , dengan jarak = 1, maka mendapatkan nilai akurasi tertinggi Sebesar = 36,66%, dengan data training = 80, dan Data Testing = 30. Sedangkan pada percobaan model K-NN, dengan jumlah $K = 2$, dan Arah GLCM = 135° , dengan jarak = 2, maka mendapatkan nilai akurasi tertinggi sebesar = 50%, dengan data training = 80, dan Data Testing = 30. Pada percobaan model K-NN, dengan jumlah $K = 7$, dan Arah GLCM = 135° , dengan jarak = 5, maka mendapatkan nilai akurasi tertinggi Sebesar = 83,33%, dengan data training = 80, dan Data Testing = 30. Dengan demikian penerapan metode gray level co-occurrence matrix dan k-nearest neighbor pada masalah tingkat kematangan buah mangga menggunakan tekstur warna perlu dikembangkan lagi hasil akurat.

Kata Kunci : Tingkat Kematangan, Gray Level Co-Occurrence Matrix, K-Nearest Neighbor

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamualaikumwarahmatullahiwabarakatuh.

Alhamdulillah, penulis dapat menyelesaikan Skripsi ini dengan judul “**Pengenalan Tingkat Kematangan Buah Mangga Menggunakan Metode GLCM Dan KNN**”, untuk memenuhi salah satu syarat menyelesaikan studi dan memperoleh gelar sarjana Strata Satu pada Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Penulis menyadari sepenuhnya bahwa skripsi ini tidak mungkin terwujud tanpa bantuan dan dorongan dari berbagai pihak, baik bantuan moril maupun materil. Untuk itu, dengan segala keikhlasan dan kerendahan hati, penulis mengucapkan banyak terima kasih dan penghargaan yang setinggi-tingginya kepada:

1. Bapak Muhamad Ichsan Gaffar, SE, M.AK, Selaku Ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo.
2. Bapak Abdul Gaffar Latjokke, M.Si, Selaku Rektor Universitas Ichsan Gorontalo.
3. Ibu Zohrahayaty, S.Kom, M.Kom, selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
4. Bapak Sudirman S. Panna, M.Kom, selaku Pembantu Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Ichsan Gorontalo dan selaku Pembimbing II yang telah berkenan meberikan tambahan ilmu dan solusi pada setiap permasalahan atas kesulitan dalam penulisan skripsi ini.
5. Ibu Irma Surya Kumala Idris, M.Kom, selaku Pembantu Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

6. Bapak Sudirman Melangi, M.Kom, selaku Pembantu Dekan III Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
7. Bapak Irfan A. Salihi, S.Kom, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
8. Ibu Rezqiwati Ishak, M.Kom, selaku Pembimbing I yang telah membimbing penulis dengan sepenuh hati, sabar dan memberikan banyak ilmu hingga penulis bisa menyelesaikan skripsi.
9. Bapak dan Ibu Dosen Universitas Ichsan Gorontalo yang telah mendidik dan mengajarkan disiplin ilmu kepada penulis.
10. Kepada orang tua saya tercinta, atas segala kasih sayang, jerih payah dan doa restunya dalam membesarkan dan mendidik penulis.
11. Rekan-rekan seperjuangan yang telah telah banyak memberikan bantuan dan dukungan moril yang sangat besar kepada penulis.
12. Keluarga Skipoy yang tidak henti-hentinya memberi dukungan yang besar kepada penulis.
13. JDS Complex yang telah membantu penulis dalam penyusunan skripsi.
14. Dan semua pihak yang ikut membantu dalam penyelesaian Skripsi ini yang tak sempat penulis sebutkan satu-persatu.

Semoga Allah SWT melimpahkan balasan atas jasa-jasa mereka kepada kami. Penulis menyadari bahwa sepenuhnya bahwa apa yang telah dicapai masih jauh dari kata kesempurnaan dan masih banyak terdapat kekurangan. Oleh karena itu, penulis sangat mengharapkan adanya kritik dan saran yang konstruktif. Akhirnya penulis berharap semoga hasil yang telah dicapai ini dapat bermanfaat bagi kita semua, Aamiin.

Gorontalo, 3 Desember 2020

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
PERSETUJUAN SKRIPSI.....	ii
PERSETUJUAN SKRIPSI.....	iii
PERNYATAAN SKRIPSI.....	iv
ABSTRACT	v
ABSTRAK	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xii
BAB I PENDAHULUAN.....	1
1. 1. Latar Belakang Masalah.....	1
1. 2. Batasan Masalah.....	3
1. 3. Identifikasi Masalah	3
1. 4. Rumusan Penelitian.....	3
1. 5. Tujuan Penelitian	3
1. 6. Manfaat Penelitian	3
1. Manfaat Teoritis.....	3
2. Manfaat Praktis	3
BAB II LANDASAN TEORI	4
2. 1. Tinjauan Studi	4
2. 2. Tinjauan Pustaka	5
2.2. 1. Tingkat Kematangan Mangga.....	5
2.2. 2. Jenis-Jenis Tingkat Kematangan Mangga.....	5
1. Mangga Mentah.....	5
2. Mangga Mengkal.....	5
3. Mangga Matang.....	6
4. Mangga Busuk.....	6

2.2. 3. Pengolahan Citra Digital	6
2.2. 4. Jenis Citra Digital.....	7
2.2. 5. GLCM (<i>Gray Level Co-occurrence Matrix</i>).....	7
2.2. 6. KNN (<i>K-Nearst Neighbor</i>).....	10
2.2. 7. IMPLEMENTASI GLCM DAN KNN	13
2.2.7. 1. Data Citra.....	13
2.2.7. 2. Pra Proses.....	13
2.2.7. 3. Ekstraksi Ciri	13
2.2.7. 4. Pembagian Data	13
2.2.7. 5. Pelatihan Dan Pengujian.....	14
2.2.7. 6. Evaluasi.....	17
2. 3. Kerangka Pikir	18
BAB III METODE PENELITIAN	19
3. 1. Jenis, Metode, Subjek, Objek, Waktu Dan Lokasi Penelitian	19
3. 2. Pengumpulan Data	20
3. 3. Pemodelan.....	20
3.3.1. Pra Pengolahan.....	21
3.3.2. Ekstraksi Ciri	21
3.3.3. Data <i>Training</i>	21
3.3.4. <i>Training</i> Menggunakan KNN	21
3.3.5. Data <i>Testing</i>	21
3.3.6. Model	22
3.3.7. Hasil Klasifikasi	22
3.3.8. Evaluasi	22
BAB IV HASIL PENELITIAN.....	23
4. 1. Hasil Pengumpulan Data.....	23
4. 2. Hasil Pemodelan.....	25
4.2.1 Pra Pengolahan	25

4.2.2 Ekstraksi Fitur	29
4.2.3 Klasifikasi.....	36
4. 3. Evaluasi	44
4.3.1 Confusion Matrix.....	44
BAB V PEMBAHASAN	45
5. 1. Pembahasan Model	45
5. 2. Alat Penelitian.....	46
5. 3. Pembahasan Sistem.....	46
BAB VI PENUTUP	48
6. 1. Kesimpulan	48
6. 2. Saran.....	48
DAFTAR PUSTAKA	49
CODE PROGRAM	51
DAFTAR RIWAYAT HIDUP	64
GAMBAR DATA TRAINING.....	65
GAMBAR DATA TESTING	66

DAFTAR GAMBAR

Gambar 2.1 (Mangga Mentah).....	5
Gambar 2.2 (Mangga Mengkal).....	5
Gambar 2.3 (Mangga Matang).....	6
Gambar 2.4 (Mangga Busuk).....	6
Gambar 2.5 (Orientasi Sudut dan Jarak).....	8
Gambar 2.6 (Contoh Penentuan Awal GLCM).....	8
Gambar 2.7 (Matriks framework menjadi matriks simetris).....	8
Gambar 2.8 (Normalisasi matriks GLCM)	9
Gambar 2.9 (Contoh Klasifikasi pada <i>K nearest neighbor</i>).....	11
Gambar 2.10 (Penerapan KNN).....	12
Gambar 2.11 (Pembagian data latih dan data uji)	14
Gambar 2.12 (Perbandingan akurasi dari setiap sudut)	15
Gambar 2.13 (Perbandingan akurasi citra RGB dan <i>grayscale</i>)	15
Gambar 2.14 (Hasil akurasi 90° <i>grayscale</i>)	16
Gambar 2.15 (Penurunan Klasifikasi pada KNN).....	17
Gambar 3.1 (Desain Penelitian)	19
Gambar 3.2 (Pemodelan)	20
Gambar 5.1 (Tampilan Menu Utama).....	46
Gambar 5.2 (Hasil Klasifikasi)	47

DAFTAR TABEL

Tabel 2.1 (Tinjauan Studi)	4
Tabel 2.2 (Perbandingan Citra <i>grayscale</i> dan RGB)	14
Tabel 2.3 (<i>Confusion Matirks</i>)	17
Tabel 4.1 (Dataset Mangga).....	23
Tabel 4.2 (Proses Perubahan Citra Asli Menjadi <i>Grayscale</i>).....	26
Tabel 4.3 (Proses Normalisasi Citra)	27
Tabel 4.4 (Hasil Segmentasi Otsu).....	28
Tabel 4.5 (Nilai dari suatu citra)	29
Tabel 4.6 (Pencarian nilai kesaamaan berdasarkan jarak 1 dan arah 0).....	30
Tabel 4.7 (Matrix GLCM).....	31
Tabel 4.8 (Perhitungan matrix normalisasi).....	32
Tabel 4.9 (Matrix Ternormalisasi)	33
Tabel 4.10 (Nilai Contras,Homogenety,Entropy,Energy Data Training)	35
Tabel 4.11 (Data Training).....	37
Tabel 4.12 (Perbandingan Nilai)	38
Tabel 4.13 (Hasil Pengurutan dan Nilai Terkecil sampai Terbesar).....	41
Tabel 4.14 (Hasil Jumlah K yang Sesuai Data)	41
Tabel 4.15 (Hasil Perhitungan Data Testing).....	42
Tabel 4.16 (Hasil Data Testing)	42

BAB I

PENDAHULUAN

1.1 Latar Belakang

Salah satu buah musiman yang sudah banyak di ketahui oleh masyarakat indonesia adalah buah mangga. Di dataran rendah dan berhawa panas tanaman mangga ini dapat tumbuh dengan baik. Tidak hanya di dataran rendah, tanaman mangga ini juga bisa tumbuh di ketinggian 600 meter dpl [1].

Dalam kasus kematangan buah mangga ini terkadang ada buah mangga yang memiliki warna yang cukup matang tetapi masih terasa asam, begitu pun sebaliknya [2]. Sehingga di perlukan proses grading untuk dapat mengetahui tingkat kematangan dari buah Mangga. Proses grading ini bertujuan untuk memilih mangga yang terbaik dan berkualitas untuk di distributorkan ke konsumen. Hanya saja proses grading membutuhkan cost yang besar apabila menggunakan tenaga manusia ketika dilakukan dalam jumlah yang besar.

Febry Yuni Mulato (2016) pernah melakukan penelitian tentang bagaimana kematangan pada buah Jambu Biji Merah bisa di ketahui dengan menggunakan metode Fuzzy. Metode yang di gunakan oleh mulato yaitu dengan mengubah tipe gambar jambu biji merah dari RGB ke tipe *grayscale* dan setelah itu di ekstraksi menggunakan aplikasi MATLAB untuk mendapatkan informasi gambar. Hasil yang di dapatkan dari proses ini yaitu tingkat kematangan dari buah jambu biji merah yang dibagi menjadi empat yaitu merah, setengah matang, matang, dan busuk.

Atas Sudrajat Qodri (2016) juga pernah melakukan penelitian pada buah Mangga Manalagi dengan menggunakan pengolahan citra untuk dapat mengetahui mutu buah mangga tersebut. Dengan mengambil citra pada buah mangga manalagi dan mencari segmentasi citra menjadi salah satu metode dalam penelitian ini. Dari metode ini akurasi yang di peroleh adalah kelas mutu A sebesar 100%, kemudian kelas mutu B dan mutu D yang memiliki akurasi produksi yaitu sebesar 80%, dan yang terakhir kelas mutu C yaitu memiliki akurasi produksi sebesar 60%.

Pengolahan citra adalah bagian dari perkembangan teknologi yang di harapkan mesin komputer dapat mengetahui citra seperti layaknya penglihatan manusia. Pada saat ini teknologi telah melibatkan salah satu sistem pengenalan secara komputerisasi dengan menggunakan konsep ataupun metode untuk salah satu pemecahan masalah [3]. Setiap buah memiliki karakteristik sendiri dalam tingkat kematangan yang berbeda-beda dan salah satu aspek yang dapat membedakannya yaitu berdasarkan warna buah tersebut. Seperti pada buah mangga yang memiliki berbagai macam jenis dan setiap jenisnya memiliki ciri warna kematangan yang berbeda-beda. Dalam penelitian ini buah mangga akan di ambil sebagai salah satu objek citra buah yang di kenali dalam tiga kategori yaitu mentah, mengkal, atau matang. Salah satu metode yang akan di gunakan pada pengenalan buah mangga adalah *Gray Level Co-occurrence Matrix* (GLCM).

GLCM sendiri adalah salah satu metode statistika untuk membentuk sebuah fitur/ciri yang tidak bisa di dasarkan pada nilai piksel semata dan berhubungan dengan piksel. Penciri yang akan digunakan adalah tekstur, yang merupakan pembeda objek buah mangga satu dengan yang lain.

Metode algoritma KNN (*K-nearest neighbor*) sudah banyak di gunakan untuk melakukan penelitian sebelumnya. KNN (*K-nearest neighbor*) adalah pengenalan pola, kategori teks ataupun pengolahan objek yang di lakukan dengan tehnik yang sederhana, efisien dan juga efektif. karena kesederhanaan pengolahannya serta mampu melakukan traning data dalam jumlah yang besar.

Berdasarkan penjelasan di atas, penulis akan menggunakan GLCM untuk fitur ekstrasi pada tingkat kematangan buah mangga dan KNN (*K-nearest neighbor*) sebagai metode algoritmanya.

1.2 Batasan Masalah

Dalam penelitian ini hanya akan meneliti salah satu jenis mangga yaitu mangga Macan Indramayu.

1.3 Identifikasi Masalah

Berdasarkan latar belakang yang telah disusun di atas maka dapat diketahui bahwa pokok permasalahannya adalah proses grading pada kematangan buah mangga yang belum tepat dan konsisten.

1.4 Rumusan Masalah

Bagaimana kinerja metode KNN dan GLCM dalam tingkat kematangan buah mangga ?

1.5 Tujuan Penelitian

Untuk mengetahui kinerja metode KNN dan GLCM dalam mengenali kematangan buah mangga.

1.6 Manfaat Penelitian

1. Manfaat Teoritis

Manfaat teoritis dalam penelitian adalah:

1. Diharapkan penelitian ini bisa menjadi menambah ilmu pengetahuan di sistem komputer dalam hal pengenalan tingkat kematangan buah mangga.
2. Bisa membantu proses grading yang masih keliru dalam hal kematangan buah mangga.

2. Manfaat Praktis

Manfaat penelitian Tingkat Kematangan Buah Mangga Menggunakan Metode KNN dapat digunakan sebagai salah satu sistem yang bisa diambil dalam mengetahui tingkat kematangan buah mangga.

BAB II

LANDASAN TEORI

2.1. Tinjauan Studi

Tabel 2.1 : Tinjauan Studi

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
1	Atas Sudajrat Qodri	Pemutuan Buah Mangga(<i>Mangifera Indica L</i>) dengan menggunakan Pengolahan Citra (<i>Image Processing</i>)	2016	Pengolahan Citra	80%
2	Febry Yuni Mulato	Klasifikasi Kematangan Buah Jambu Biji Merah (<i>Psidium Guajava</i>) Dengan Menggunakan Model Fuzzy	2015	Fuzzy	94.67%
3	Arif Patriot Sri Pamungkas, Nur Nafi'iyah, Nur Qomariyah Nawafilah	KNN Klasifikasi Kematangan Buah Mangga Manalagi Menggunakan L*A*B dan Fitur Statistik	2019	KNN,L*A*B dan Fitur Statistik	62,5%

2.2 Tinjauan Pustaka

2.2.1 Tingkat Kematangan Mangga

Sifat manusia dalam memilih kematangan buah mangga terlalu subjektif dan tidak konsisten sehingga antara penilai satu dan penilai lainnya hasilnya akan berbeda [2]. Dengan adanya perkembangan teknologi informasi, memungkinkan untuk mengenali kematangan buah pada mangga dengan bantuan komputer.

2.2.2 Jenis-Jenis Tingkat Kematangan Mangga

1. Mangga Mentah

Warna kulit pada mangga yang masih mentah lebih kehijauan dan juga tekstur yang masih sangat keras.



Gambar 2.1 : buah mangga mentah

2. Mangga Mengkal

Pada mangga yang mengkal kulitnya berwarna hijau ke kuningan.



Gambar 2.2 : buah mangga mengkal

3. **Mangga Matang**

Pada mangga yang matang warna kulitnya lebih ke kuningan dan juga teksturnya yang sudah begitu lembut.



Gambar 2.3 : buah mangga matang

4. **Mangga Busuk**

Untuk mangga busuk kulitnya akan berubah menjadi hitam kekuningan



Gambar 2.4 : buah mangga busuk

2.2.3 **Pengolahan Citra**

Pengolahan citra adalah piksel-piksel yang di proses di dalam citra digital yang bertujuan untuk mengambil citra dan memperbaiki kualitas pada gambar yang memiliki gangguan (cacat, kabur, kurang tajam, terlalu kontras, dan lain sebagainya) agar dapat dipahami oleh manusia dan komputer [4].

2.2.4 Jenis-Jenis Citra Digital

Pengolahan citra digital di bagi menjadi 3 bagian, yaitu *RGB image*, *Grayscale image* dan *binary image*.

a. RGB

Citra warna pada setiap piksel memiliki tiga kombinasi warna dari warna dasar (RGB = Red Green Blue). Dalam Setiap warna dasar yang di gunakan menyimpan masing-masing 8 bit = 1 byte, sehingga tingkat warna dasar menjadi 255 warna [5].

b. Grayscale

Proses dari *Grayscale* mengembalikan citra berwarna RGB yang terdiri dari 3 layer menjadi 1 layer [6]. Rumus Untuk mendapatkan nilai *grayscale* yaitu dengan cara mengambil nilai rata-rata dari nilai R, G, dan B [7].

$$s = \frac{r + g + b}{3}$$

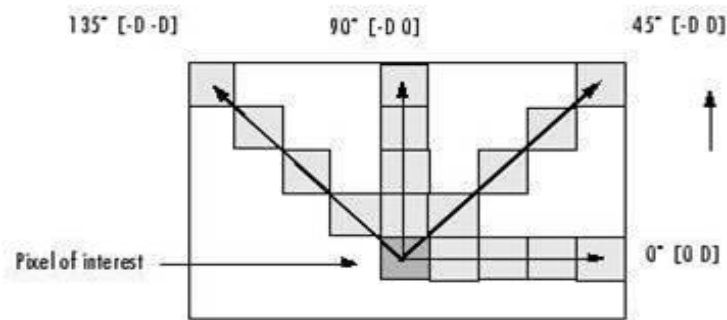
(1)

c. Binary

Warna hitam ataupun putih merupakan warna untuk setiap piksel, maka di perlukan 1 bit per piksel (0 dan 1) atau apabila dalam 8 bit (0 dan 255), sehingga dalam hal penyimpanan itu sangat efisien. Biner yang di wakili oleh gambar akan cocok untuk sebuah teks, sidik jari maupun gambar arsitektur[8].

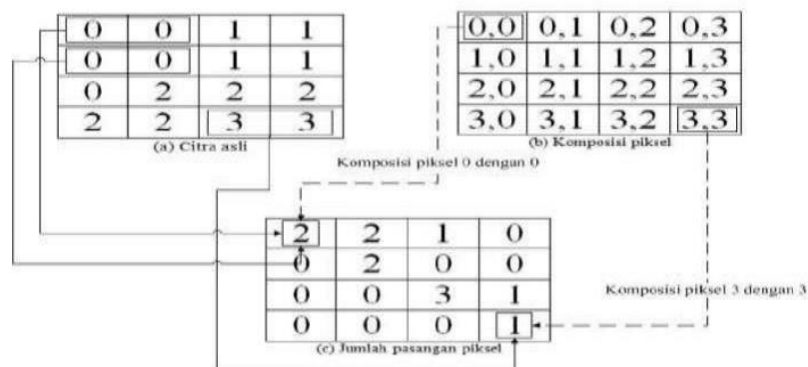
2.2.5 GLCM (*Gray Level Co-occurrence Matrix*)

GLCM adalah salah satu metode yang bertujuan untuk menganalisa tekstur/ekstraksi ciri [9]. Jarak d dan orientasi sudut θ merupakan koordinat pada kedua pasang piksel. GLCM mewakili hubungan antara dua piksel yang bertetangga, dimana kedua piksel tersebut memiliki tingkat keabuan dan juga keduanya memiliki jarak dan arah tertentu. Jarak dinyatakan piksel dan arah dinyatakan sudut. 1, 2, 3 dan seterusnya merupakan nilai dari jarak, sedangkan 0° , 45° , 90° , 135° dan seterusnya merupakan nilai dari arah [10].



Gambar 2.5 : Orientasi sudut dan jarak

Sebagai ilustrasi, ketetanggaan piksel yang dapat dipilih ke arah timur (kanan). Salah satu untuk merepresentasikan hubungan ini yaitu berupa (1,0), yang menyatakan hubungan dua piksel bernilai 1 diikuti dengan piksel bernilai 0, sehingga jumlah kelompok piksel memenuhi hubungan tersebut dihitung [10].



Gambar 2.6 : contoh penentuan awal GLCM

Matriks ini di namakan matriks framwork yang dimana matriks ini akan mengolah menjadi simetris. Caranya seperti pada Gambar 2.6 dengan menambah hasil transpos.

$$\begin{bmatrix} 2 & 2 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

Tranpos
GLCM sebelum dinormalisasi

Gambar 2.7 : Matriks framework menjadi matriks simetris.

Pada ukuran citra yang bergantung bisa di hilangkan dengan cara menormalisasikan nilai pada elemen GLCM agar jumlahnya bernilai 1. Hasilnya bisa di lihat pada Gambar 2.7.

$$\begin{bmatrix} \frac{4}{24} & \frac{2}{24} & \frac{1}{24} & \frac{0}{24} \\ \frac{2}{24} & \frac{4}{24} & \frac{0}{24} & \frac{0}{24} \\ \frac{1}{24} & \frac{0}{24} & \frac{6}{24} & \frac{1}{24} \\ \frac{0}{24} & \frac{0}{24} & \frac{1}{24} & \frac{2}{24} \end{bmatrix}$$

Gambar 2.8 : Normalisasi Matriks GLCM

Tahapan yang dilakukan pada perhitungan GLCM adalah sebagai berikut:

1. Pembentukan matriks awal GLCM dari pasangan dua piksel yang berjajar sesuai dengan arah 0° , 45° , 90° atau 135° .
2. Membentuk matriks yang simetris dengan menjumlahkan matriks awal GLCM dengan nilai transposnya.
3. Menormalisasi matriks GLCM dengan membagi setiap elemen matriks dengan jumlah pasangan piksel.
4. Ekstraksi ciri, yaitu:
 - a. Contrast
 kontras adalah elemen-elemen matriks citra yang di tunjukkan untuk ukuran penyebaran (momen inersia). Nilai kekontrasannya besar apabila letaknya jauh dari diagonal utama.

$$Contrast = \sum_{i_1} \sum_{i_2} (i_1 - i_2)^2 p(i_1, i_2) \quad (2)$$

- b. Homogeneity
 Homogenitas adalah ukuran nilai yang di wakili sebuah persamaan variasi dari intensitas citra. Nilai maksimum pada

homogenitas apabila nilai piksel memiliki nilai yang seragam. Rumus ditunjukkan pada persamaan 4 [11].

$$Homogeneity = \sum_{i_1} \sum_{i_2} \frac{p(i_1, i_2)}{1 + |i_1 - i_2|} \quad (3)$$

c. Energy

Energy adalah keberagaman intensitas keabuan dalam citra yang di perhitungkan. Rumus ditunjukkan pada persamaan 4

$$Energy = \sum_{i_1} \sum_{i_2} p^2(i_1, i_2) \quad (4)$$

d. Entropy

Menunjukkan ukuran yang bentuknya tidak teratur. Harga ENT akan bernilai kecil apabila struktur dari citra tidak teratur, dan apabila citra dari perpindahan derajat keabuan yang merata nilai ENT di nyatakan besar [12].

$$Entropy = - \sum_{i_1} \sum_{i_2} p(i_1, i_2) \log p(i_1, i_2) \quad (5)$$

2.2.6 KNN (K-Nearest Neighbor)

Algoritma KNN adalah metode untuk mengklasifikasi pada objek yang di dasari oleh data pembelajaran dan metode ini menentukan jarak yang paling dekat dengan objek tersebut. Ruang berdimensi banyak di perhitungkan dari data pembelajaran, dimana masing masing dimensi merepresentasikan fitur dari data. Data pembelajaran ini dibagi menjadi beberapa bagian ruang yang di dasarkan

oleh klasifikasi. Pada ruang ini ditandai sebuah titik kelas c jika klasifikasi yang sering ditemukan pada k buah tetangga terdekat titik tersebut adalah kelas c [13]. Untuk mengukur jarak kedekatan antara data baru dan data lama (data *training*) itu ada beberapa caranya, contohnya *euclidean distance* dan *manhattan distance* (*city block distance*) dan yang sering digunakan adalah *euclidean distance* [14].

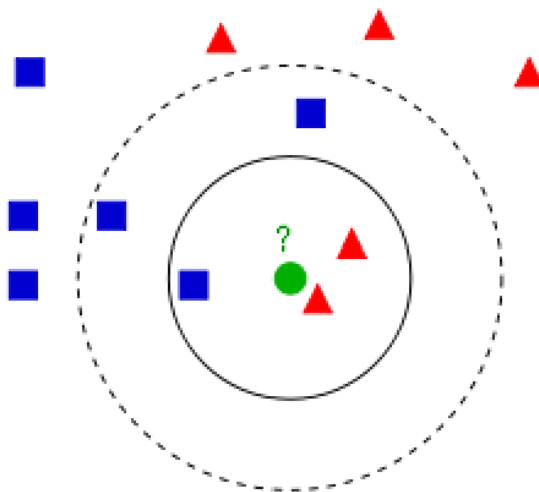
$$Distance = \sqrt{\sum_{i=1}^n (X_{training}^i - X_{testing})^2}$$

(6)

Yaitu:

- $X_{training}^i$: Data training ke- i
- $X_{testing}$: Data testing
- i : Record (baris) ke- i dari table
- n : Jumlah data training

Adapun contoh penjelasan KNN berdasarkan pilihan terbanyak dari tetangga yang akan diklasifikasikan melalui sebuah objek.

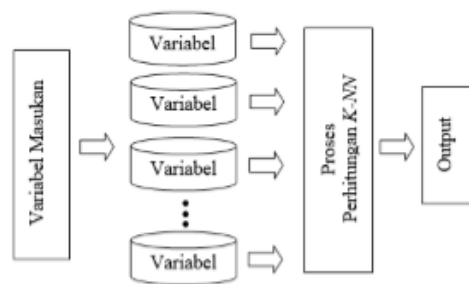


Gambar 2.9 : Contoh Klasifikasi pada *k-nearest neighbor*

Salah satu jenis dari pembelajaran berdasarkan perwujudan adalah metode ini. nilai pendekatan secara lokal dan semua perhitungan ditahan hingga proses klasifikasi.

Metode yang paling sederhana dar) [15]. Pada data pembelajaran di perhitungkan untuk ke ruang berdimensi banyak, dan masing-masing dimensi di wakikan oleh fitur dari data. Klasifikasi data pembelajaran pada ruang ini di bagi menjadi beberapa bagian. Kelas C di tandai dengan sebuah titik pada ruang ini. Pada titik tersebut kelas C akan di klasifikasi jika kelas C sering di temui pada k buah tetangga terdekat. Semua metode pembelajaran mesin yaitu metode *k-nearest neighbor* (KNN). Sebagaimana persamaan jarak Euclidean bisa di hitung melalui dekat ataupun jauhnya tetangga. Lingkaran berwarna hijau dan bujur sangkar biru akan di klasifikasikan pada klasifikasi pertama dan segitiga merah pada klasifikasi kedua. Bisa di lihat pada contoh Gambar 2.8.

Selain itu ada contoh lainnya dalam penerapan KNN seperti :



Gambar 2.10 : Penerapan KNN

1. Tentukan parameter k
2. Hitung jarak antara data yang akan dievaluasi dengan semua pelatihan
3. Urutkan jarak yang terbentuk (urut naik)
4. Tentukan jarak terdekat sampai urutan k
5. Pasangkan kelas yang bersesuaian
6. Cari jumlah kelas dari tetangga yang terdekat dan tetapkan kelas tersebut sebagai kelas data yang akan dievaluasi.

2.2.7 Implementasi GLCM dan KNN

Contoh studi kasus :

Berdasarkan studi kasus dalam penelitian sebelumnya pernah melakukan penelitian dengan judul “IDENTIFIKASI VARIETAS DURIO ZIBETHINUS BERDASARKAN SEBARAN TRIKOMA DAUN MENGGUNAKAN GLCM DAN KNN”.

2.2.7.1 Data Citra

Dalam penelitian ini data yang akan di gunakan 10 varietas dan masing-masing varietas berjumlah 10 citra trikoma daun durian. Jadi total dari keseluruhannya berjumlah 100 sampel yang akan di identifikasi.

2.2.7.2 Pra Proses

Selanjutnya pada tahap ini 100 data citra di ubah dari R.G.B menjadi *Grayscale*. Setiap citra R.G.B komponen warnanya di pisah agar bisa membandingkan hasil dari citra *red*, *green* dan *blue* dan *Grayscale*.

2.2.7.3 Ekstrasi Ciri

GLCM menggunakan ekstrasi ciri dan 0° , 45° , 90° dan 135° sebagai parameter sudut, serta $d = 1$ di jadikan jarak. 13 fitur GLCM di gunakan dalam ekstrasi ciri. Setiap fitur di ambil nilainya untuk di bentuk dalam boxplot pada minitab agar bisa menganalisa nilai GLCM dalam pola sebaran. 74% menjadi nilai tertinggi dalam menggunakan 13 fitur ini. Di penelitian kali ini fitur *sum of squares variance*, *sum average* dan *sum variance* tidak di gunakan. Dan sisanya yaitu *contrast*, *Correlation*, *Energy*, *Entropy*, *Homogeneity* / *Inverse Difference Moment (IDM)*, *Sum Entropy*, *Difference Variance*, *Difference Entropy*, *Information Measures Of Correlation2*, *Inverse Difference Moment (Normalized)* yang di gunakan dalam penelitian karena 11 fitur tersebut yang di perlukan.

2.2.7.4 Pembagian Data

Pada proses pembagian data, data latih dan data uji mendapat presentasi yaitu 80% - 20%. Hasil ini di peroleh dari menggunakan metode *K-Fold Cross Validation* dengan $k = 5$.

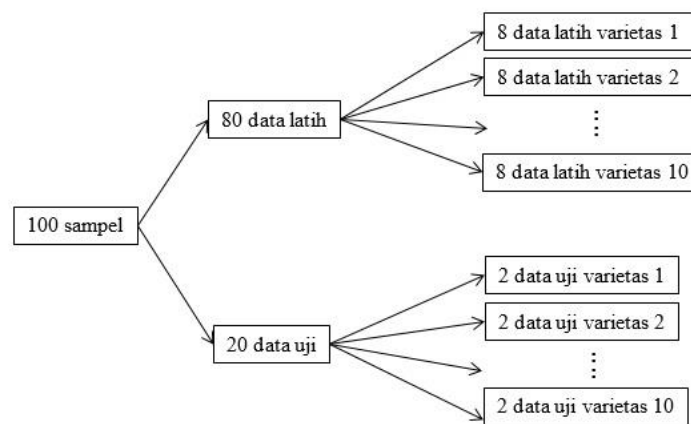
2.2.7.5 Pelatihan dan Pengujian

Kemudian pada tahap ini akan di lakukan proses pengambilan komponen warna model R.G.B agar bisa membandingkan hasil dari *Grayscale* dan R.G.B. Setelah dilakukan percobaan mendapatkan hasil yang bisa di lihat di Tabel 2.2.

	0°	45°	90°	135°	Rata-rata
Grayscale	67%	67%	78%	72%	71.00%
Red	69%	67%	71%	70%	69.25%
Green	67%	70%	71%	69%	69.25%
Blue	66%	66%	67%	69%	67.00%
Rata-rata	67.25%	67.50%	71.75%	70.00%	

Tabel 2.2 : Perbandingan citra *grayscale* dan RGB

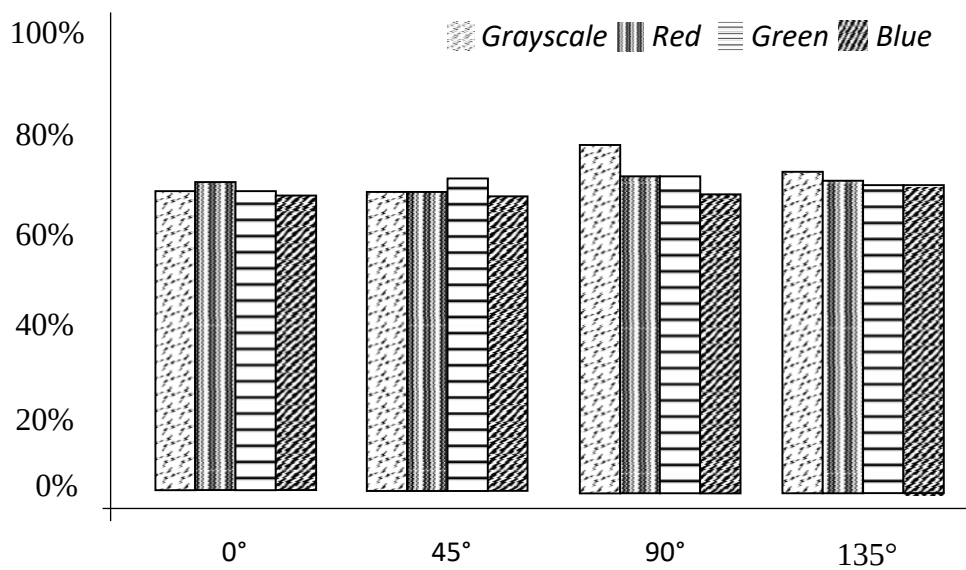
Sudut 90° dan 135° menjadi sudut yang mempunyai hasil dengan akurasi tertinggi. Sudut 90° dan 135° lebih dominan pada jarak 1 dengan intensitas tertentu yang frekuensinya dapat di lihat dari kedua pasang piksel. 71% menjadi akurasi tertinggi dengan di dasari pada citra *Grayscale*.



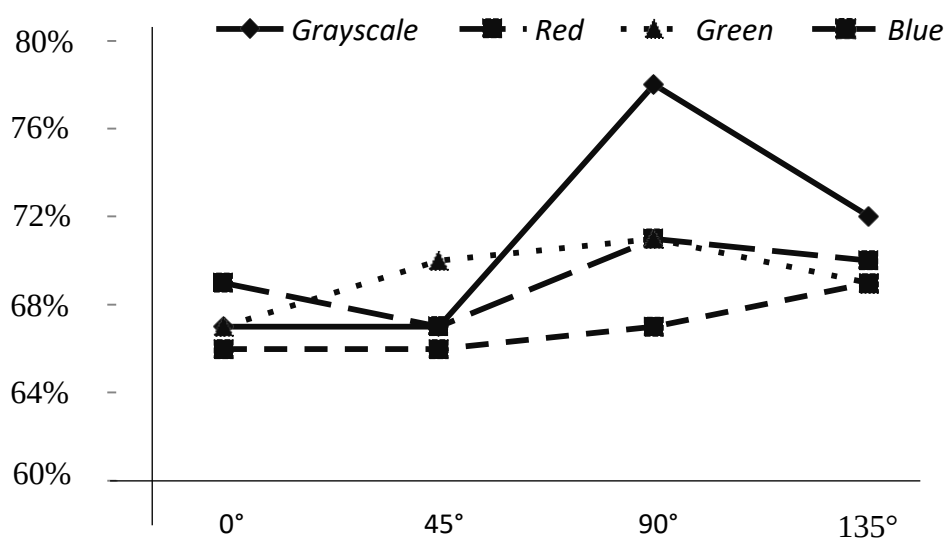
Gambar 2.11 : Pembagian data latih dan data uji

Perbandingan antara R.G.B dan *Grayscale* mendapatkan hasil bahwa tingkat akurasi tertinggi terdapat pada *Grayscale*. Hasil akurasi dari warna merah dan hijau lebih dominan di bandingkan warna biru. Yang berarti citra pola dari trikoma daun durian lebih banyak warna merah, hijau dan lebih

sedikit warna biru. Pada grafik Gambar 2.11 terdapat hasil dari percobaan untuk membandingkan antara komponen warna R.G.B dan *Grayscale*. Untuk sudut 90° mendapatkan hasil yang mempengaruhi besar akurasi. Hasil tersebut memungkinkan bentuk pola pada tekstur trikoma keteraturannya pada piksel ketetangaan tegak lurus menjadi dominan. Pada Gambar 2.12 terdapat hasil perbandingan antara warna *Grayscale* dan komponen warna R.G.B.

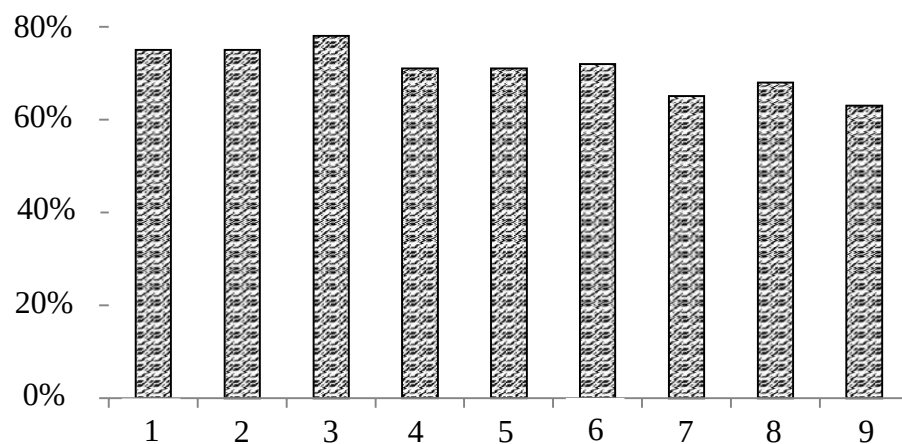


Gambar 2.12 : Perbandingan akurasi dari tiap sudut

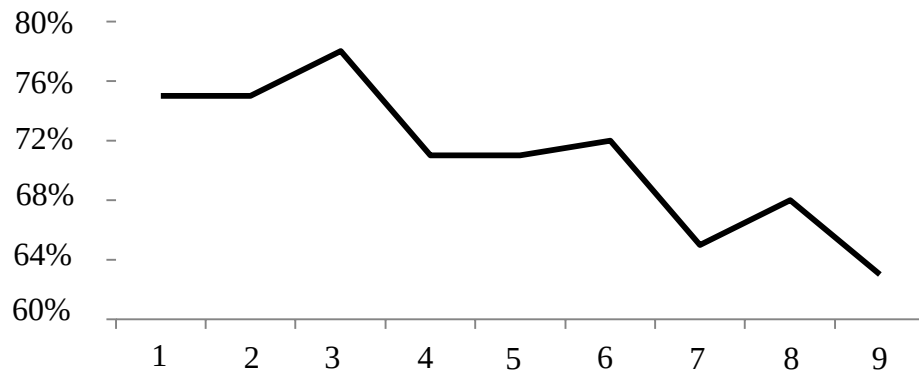


Gambar 2.13 : Perbandingan Akurasi Citra RGB dan Grayscale

Dari Gambar 2.12 bisa di lihat peningkatan akurasi tinggi dari 0° ke 90° , dan terjadi penurunan pada citra *Grayscale* pada bagian 90° ke 135° . Warna biru menjadi citra yang akurasinya paling rendah. Sudut 90° mendapatkan hasil akurasi yang tertinggi, dan menghasilkan nilai 78% dari jarak $d = 1$ dan $k = 3$. Pada $k = 9$ di dapatkan hasil terendah dengan nilai 63%. KNN melakukan perbandingan nilai k agar bisa mendapatkan hasil percobaan dengan akurasi maksimal. Di Gambar 2.13 terdapat hasil perbandingan dari setiap nilai k . Tingkat akurasinya akan semakin kecil jika parameter k saat di identifikasi citra dengan KNN semakin besar. Gambar 2.14 terdapat grafik penurunan tingkat kesalahan klasifikasi. Hasil dari grafik garis dapat dikelompokkan menjadi 3 bagian. Akurasi tertinggi terdapat pada kelompok 1 $k = 1$ sampai $k = 3$. Penurunan mulai terjadi pada $k = 4$ sampai dengan $k = 6$ yaitu kelompok 2. Dan tidak terjadi peningkatan akurasi pada kelompok 3 yaitu $k = 7$ sampai dengan $k = 9$. Berarti pada kelompok 1 menjadi penggunaan KNN yang paling tepat.



Gambar 2.14 : Hasil akurasi pada sudut 90° grayscale



Gambar 2.15 : Penurunan klasifikasi pada KNN

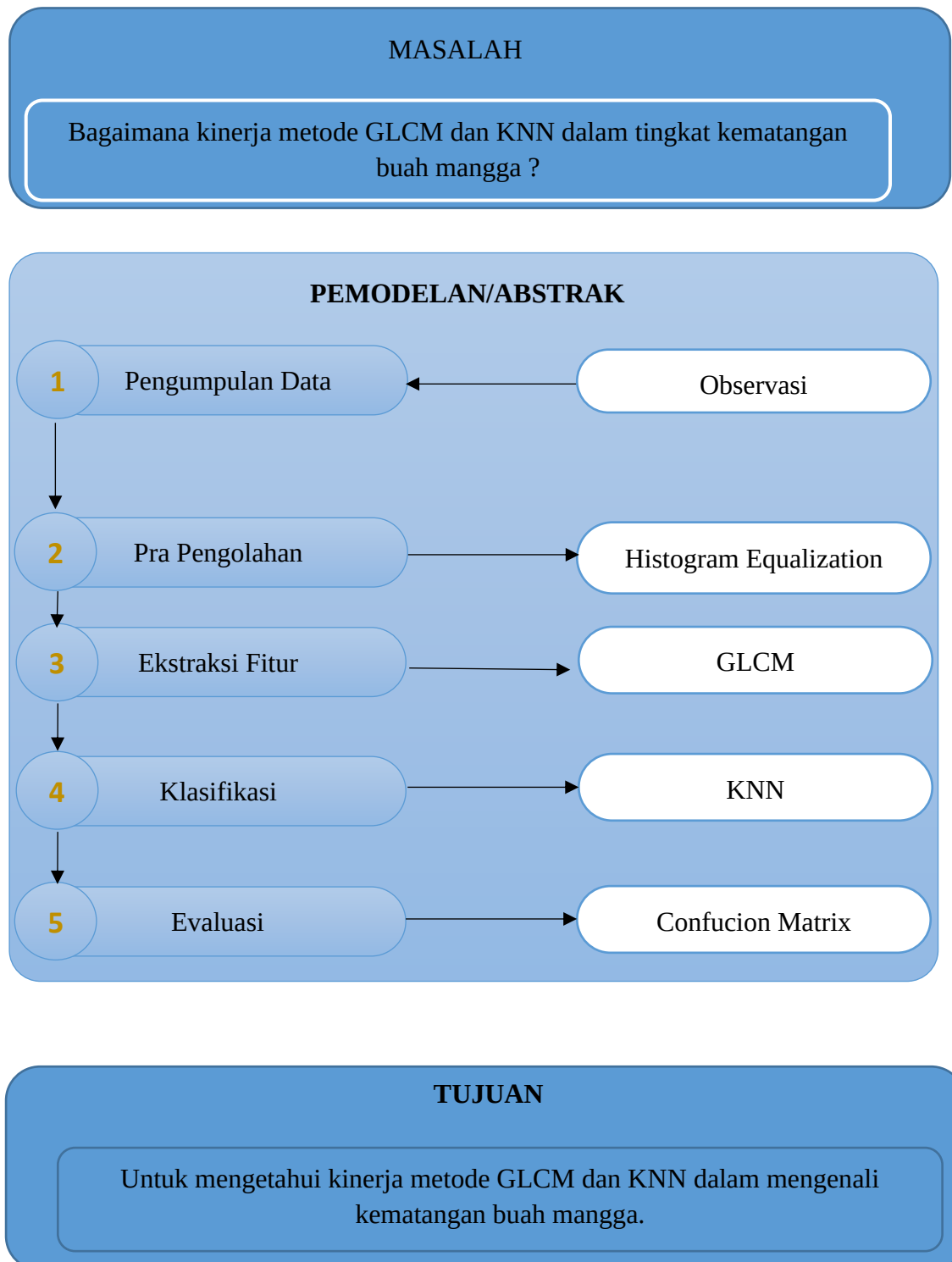
2.2.7.6 Evaluasi

Dari paparan di atas 78% menjadi hasil akurasi dari menggunakan metode *K- Nearest Neighbor* (KNN), sudut 90° pada citra *Grayscale* dan $d = 1$ sebagai jarak. *Confusion matrix* berfungsi agar mengevaluasi klasifikasi atau identifikasi yang sudah dilakukan. Pada tabel 2.3 terdapat klasifikasi yang aktual dan hasil prediksi.

Kelas aktual	Kelas Prediksi									
	1	2	3	4	5	6	7	8	9	10
1	10	0	0	0	0	0	0	0	0	0
2	0	6	0	2	1	0	0	1	0	0
3	0	1	9	0	0	0	0	0	0	0
4	0	2	1	6	1	0	0	0	0	0
5	0	0	0	1	7	0	1	1	0	0
6	0	0	0	0	0	8	0	2	0	0
7	0	0	0	0	0	1	7	1	0	1
8	0	0	0	0	2	0	0	8	0	0
9	2	0	0	0	0	0	0	0	8	0
10	0	0	0	0	0	0	1	0	0	9

Tabel 2.3 : *Confusion Matrix*

2.2.7 Kerangka Pikir



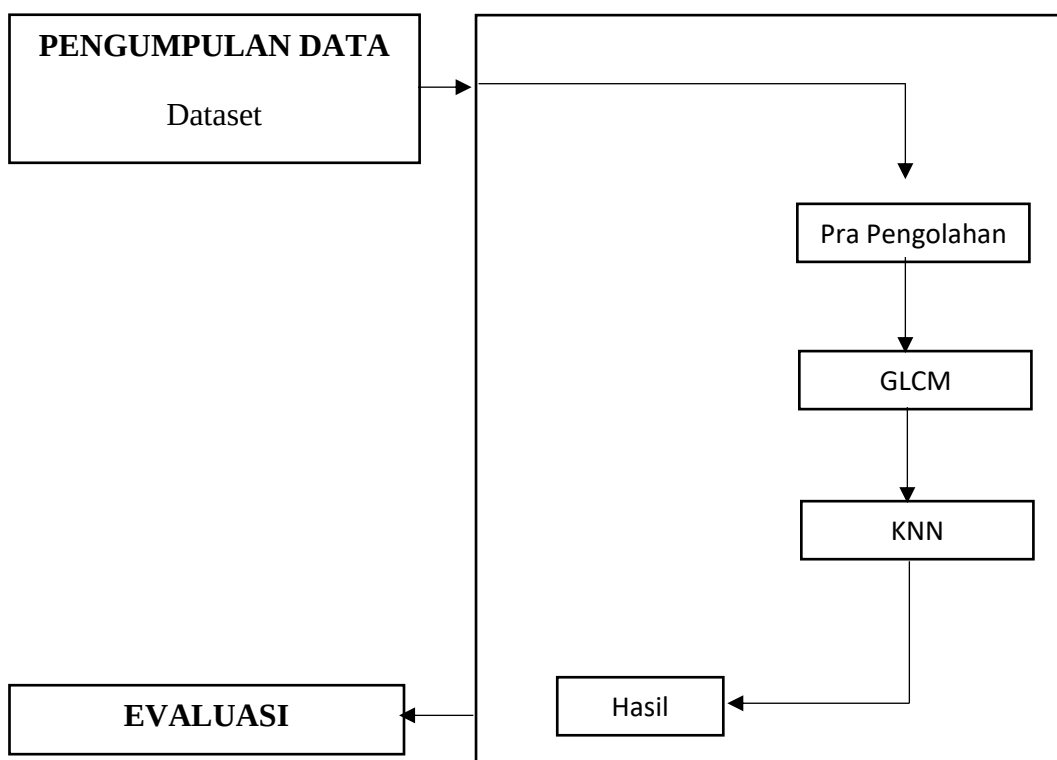
BAB III

METODE PENELITIAN

3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian

Dilihat dari tingkat penerapannya maka penelitian ini merupakan penelitian terapan. Dipandang dari jenis data yang diperoleh maka penelitian merupakan penelitian kuantitatif. Dipandang dari perlakuan terhadap data maka penelitian konfirmatori. Penelitian ini menggunakan metode penelitian *eksperimen* dengan demikian jenis penelitian ini adalah *eksperimental*.

Subjek penelitian ini adalah Klasifikasi pada objek citra pada buah mangga. Penelitian ini dimulai dari Januari 2020 sampai Desember 2020.



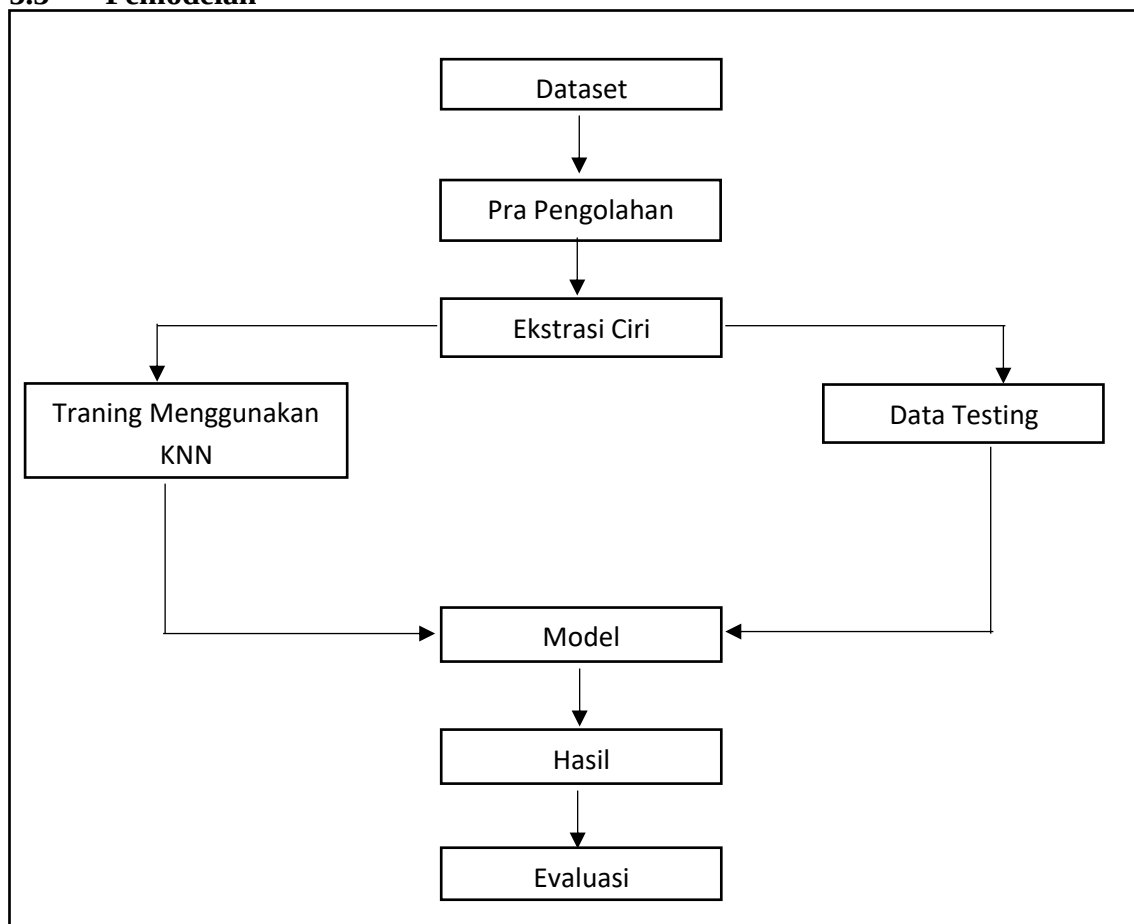
Gambar 3.1 : Desain Penelitian

3.2 Pengumpulan Data

Data yang dikumpulkan dari penelitian adalah data hasil observasi yang sudah peneliti lakukan dan juga jurnal-jurnal sebagai pendukung data tersebut. Tingkat kematangan buah mangga di bagi menjadi 4 bagian yaitu mentah, mengkal, matang dan busuk.

Citra yang dikumpulkan dibagi menjadi dua dataset yaitu: data latih dan data uji. Data latih yang digunakan sebagai pembandingan terhadap data uji pada proses pendeteksian tingkat kematangan pada buah mangga. Sedangkan data uji adalah data digunakan untuk mengetahui hasil deteksi jenis kematangan pada buah mangga yang didapatkan melalui proses deteksi.

3.3 Pemodelan



Gambar 3.2 : Pemodelan

3.3.1 Pra Pengolahan

Sebelum data diolah, terlebih dahulu dilakukan proses pra pengolahan yang menggunakan *Histogram Equalization*, hal ini dilakukan karena *Histogram Equalization* berfungsi untuk mengubah distribusi nilai derajat keabuan pada sebuah citra, sehingga menjadi seragam. *Histogram Equalization* bertujuan untuk mengetahui suatu citra sudah baik atau belum baik.

3.3.2 Ekstraksi Ciri

Pada proses ekstraksi ciri data input akan dibaca dan kemudian diolah dalam dua proses yang berlainan. Proses ekstraksi fitur tekstur menggunakan metode GLCM (*Gray Level Co-Occurrence Matrix*). Masukan dari proses ini adalah citra kematangan buah mangga yang sudah dipotong dengan ukuran tertentu. Semua citra diambil pada kondisi pencahayaan yang terang. Output dari proses ini adalah matriks $N \times 1$ dimana N adalah jumlah vektor ciri.

3.3.3 Data Training

Data Training berupa data yang telah terekstrak cirinya yang selanjutnya akan dilatih menggunakan Algoritma K-Nearest Neighbor, jumlah data training yang digunakan berupa 90% dari sampel tingkat kematangan buah mangga. Algoritma ini akan menentukan atau mencari bobot yang terbaik. GLCM merupakan hasil data training untuk tingkat kematangan buah mangga.

3.3.4 Training Menggunakan KNN

Training menggunakan K-NN menjadikan data training yang berupa angka atau parameter hasil dari citra tingkat kematangan buah mangga yang telah diolah melalui fitur ekstraksi GLCM sebagai data inputan untuk algoritma K-NN. Algoritma K-NN bekerja berdasarkan jarak terdekat dari training dan test dalam penentuannya, kemudian mengambil mayoritas hasil ketentuan kematangan buah mangga yang telah didapatkan untuk dijadikan prediksi dari test.

3.3.5 Data Testing

Data *testing* berupa data yang telah terekstraksi cirinya, untuk menguji data yang telah dilatih, jumlah data training yang digunakan berupa 10% dari sampel tingkat kematangan buah mangga yang berbeda. Data testing digunakan untuk

mengetahui keberhasilan *Classifier* berhasil melakukan klasifikasi, data testing merupakan ciri dari hasil ekstraksi GLCM terhadap kematangan buah mangga.

3.3.6 Model

Model merupakan hasil dari proses training algoritma K-NN menggunakan data testing.

3.3.7 Hasil Klasifikasi

Hasil klasifikasi merupakan output pada data testing yang didapatkan dari proses klasifikasi yang menggunakan algoritma *K-Nearest Neighbor* berdasarkan model yang diperoleh dari data *training*.

3.3.8 Evaluasi

Evaluasi bertujuan untuk mengetahui kinerja dari metode ekstraksi GLCM terhadap tingkat kematangan buah mangga yang digunakan. Proses evaluasi dilakukan pada seluruh data *testing* kemudian hasil *output* yang akan dipetakan kedalam *Confusion Matrix* untuk dihitung nilai akurasi.

BAB IV

HASIL PENELITIAN

4. 1. Hasil Pengumpulan Data

Seperti yang telah diuraikan pada tahapan metode penelitian bahwa dataset gambar yang digunakan pada penelitian ini ada berupa data real, kami telah melakukan proses pengumpulan data tersebut dan berhasil di dapat, Data terdiri 24 citra dan 4 objek dengan format file gambar adalah jpg. Berikut sampel gambar yang berhasil dilakukan :

Tabel 4.1 Dataset Mangga

MANGGA	PENGUMPULAN DATASET		
MANGGA MANTAH			
			
MANGGA MENGKAL			
			

MANGGA	PENGUMPULAN DATA SET		
MANGGA MATANG			
MANGGA BUSUK			

Untuk melakukan proses analisis pada penelitian ini adalah dengan melakukan pemilihan data gambar yang dijadikan sebagai data training dan data testing, sehingganya dari pekerjaan yang kami lakukan pada penelitian ini kami menentukan dari data yang telah berhasil kami dapatkan ditentukan jumlah data training adalah sebanyak 80 data gambar dan data testing adalah sebanyak 30 data gambar.

4.2. Hasil Pemodelan

4.2.1 Pra Pengolahan

Pra-pengolahan citra (image pre-processing), yaitu proses paling awal dalam pengolahan citra sebelum proses utama dilakukan. Pada tahap ini citra mangga yang sudah ada dikonversi agar diperoleh data citra mangga yang sesuai kebutuhan. Tahap ini berfungsi untuk menormalisasi citra buah mangga dari permasalahan luminasi yang terlalu gelap atau terlalu terang sehingga dapat meningkatkan performansi dari sistem pengenalan mangga. Pra-pengolahan dalam penelitian ini dilakukan dengan 4 tahap, yaitu:

1. Pengubahan citra warna ke *grayscale*

Pra-pengolahan pertama yang akan dilakukan adalah merubah citra *training* atau citra *testing* yang awalnya berbentuk citra dari RGB (red, green, blue) menjadi citra bentuk grayscale, perubahan ini dilakukan karena citra grayscale memiliki persamaan yang sederhana dan mampu mengurangi kebutuhan memory dimana nilai warna putih diwakili dengan angka 255 dan nilai warna hitam diwakili dengan angka 0. Berikut gambar proses perubahan citra warna ke grayscale

$$s = \frac{r + g + b}{3} \quad (1)$$

Tabel 4.2 Proses Perubahan Citra Asli Menjadi *Grayscale*

Perbedaan	<i>Images Original</i>	<i>Images Grayscale</i>
Mangga Mentah		
Mangga Mengkal		
Mangga Matang		
Mangga Busuk		

2. Normalisasi citra

Setelah citra asli di konversi ke citra abu-abu maka pra-pengolahan selanjutnya adalah normalisasi citra dengan histogram ekualisasi. Histogram ekualisasi adalah sebuah proses yang mengubah distribusi nilai derajat keabuan pada sebuah citra sehingga menjadi seragam. Tujuan dari histogram equalization adalah untuk memperoleh penyebaran histogram yang merata sehingga setiap derajat keabuan memiliki jumlah piksel yang relatif sama. Berikut gambar normalisasi citra abu-abu dengan histogram ekualisasi.

Tabel 4.3 Proses Normalisasi Citra

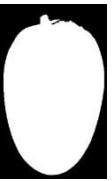
Perbedaan	Citra <i>Grayscale</i>	Citra <i>Histogram</i>
Mangga Mentah		
Mangga Mengkal		
Mangga Matang		

Perbedaan	Citra Grayscale	Citra Histogram
Mangga Busuk		

3. Segmentasi Citra

Segmentasi citra merupakan bagian dari proses pengolahan citra. Segmentasi citra (*image segmentation*) mempunyai arti membagi suatu citra menjadi wilayah-wilayah yang homogen berdasarkan kriteria keserupaan tertentu antara tingkat keabuan suatu piksel dengan tingkat keabuan piksel-piksel tetangganya, kemudian hasil dari proses segmentasi ini akan digunakan untuk proses lebih lanjut. Metode Otsu merupakan salah satu metode untuk segmentasi citra digital dengan menggunakan nilai ambang secara otomatis, yakni mengubah citra digital warna abu-abu menjadi hitam putih berdasarkan perbandingan nilai ambang dengan nilai warna piksel citra digital.

Tabel 4.4 Hasil Segmentasi Otsu

Citra Asli	Nilai R	Nilai G	Nilai B	Hasil Otsu	Hasil Segmentasi
					

4.4.2. Ekstrasi Fitur

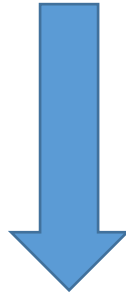
Fitur ekstrasi Grey Level Co-occurrence Matrix (GLCM) merupakan matrix yang menggambarkan frekuensi munculnya pasangan dua pixel dengan intensitas tertentu dalam jarak d dan orientasi arah sudut 0° , 90° , 135° , atau 45° tertentu dalam citra.

Contoh: Jarak 1 dan orientasi arah sudut 0°

Nilai dari Gambar 4.10 berupa matrix 10×10 Yang sebenarnya matrix 104×104 dan di ambil dari 1 contoh gambar 1 dataset mangga busuk yang disederhanakan unruk memudahkan perhitungan :

Tabel 4.5 Nilai Dari Suatu Citra

172	170	171	173	174	175	174	171	172	173
170	170	173	174	173	173	173	171	169	172
172	172	174	175	173	171	171	170	166	169
173	172	173	173	172	171	172	170	170	170
171	169	170	171	170	171	173	174	178	177
171	171	172	173	170	170	173	176	179	179
173	173	175	175	171	169	171	173	172	173
173	172	174	175	172	170	169	169	167	167
171	173	173	172	171	171	170	169	168	169
171	173	173	173	173	173	172	169	171	170



Tabel 4.6 pencarian nilai kesamaan berdasarkan jarak 1 dan arah 0°

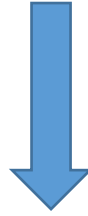
172	170	171	173	174	175	174	171	172	173
170	170	173	174	173	173	173	171	169	172
172	172	174	175	173	171	171	170	166	169
173	172	173	173	172	171	172	170	170	170
171	169	170	171	170	171	173	174	178	177
171	171	172	173	170	170	173	176	179	179
173	173	175	175	171	169	171	173	172	173
173	172	174	175	172	170	169	169	167	167
171	173	173	172	171	171	170	169	168	169
171	173	173	173	173	173	172	169	171	170

Tabel 4.7 Matrix GLCM

i/j	166	167	168	169	170	171	172	173	174	175	176	177	178	179
166	0	0	0	1	0	0	0	0	0	0	0	0	0	0
167	0	1	0	0	0	0	0	0	0	0	0	0	0	0
168	0	0	0	1	0	0	0	0	0	0	0	0	0	0
169	1	1	1	1	3	3	1	0	0	0	0	0	0	0
170	1	0	0	3	4	6	1	3	0	0	0	0	0	0
171	0	0	0	3	6	3	3	7	1	1	0	0	0	0
172	0	0	0	1	1	3	1	5	1	1	0	0	0	0
173	0	0	0	0	3	7	5	9	3	2	1	0	0	0
174	0	0	0	0	0	1	1	3	0	2	0	0	1	0
175	0	0	0	0	0	1	1	2	2	1	0	0	0	0
176	0	0	0	0	0	0	0	1	0	0	0	0	0	1
177	0	0	0	0	0	0	0	0	0	0	0	0	1	0
178	0	0	0	0	0	0	0	0	1	0	0	1	0	0
179	0	0	0	0	0	0	0	0	0	0	1	0	0	1

jumlah total nilai GLCM = 121.

Selanjutnya di table 4.8 nilai dalam pixel di bagi dengan total nilai glcm agar mendapatkan nilai matrix ternormalisasi

**Tabel 4.8** perhitungan matrix normalisasi

i/j	166	167	168	169	170	171	172	173	174	175	176	177	178	179
166	0	0	0	1:121	0	0	0	0	0	0	0	0	0	0
167	0	1:121	0	0	0	0	0	0	0	0	0	0	0	0
168	0	0	0	1	0	0	0	0	0	0	0	0	0	0
169	1:121	1:121	1:121	1:121	3:121	3:121	1:121	0	0	0	0	0	0	0
170	1:121	0	0	3:121	4:121	6:121	1:121	3:121	0	0	0	0	0	0
171	0	0	0	3:121	6:121	3:121	3:121	7:121	1:121	1:121	0	0	0	0
172	0	0	0	1:121	1:121	3:121	1:121	5:121	1:121	1:121	0	0	0	0
173	0	0	0	0	3:121	7:121	5:121	9:121	3:121	2:121	1:121	0	0	0
174	0	0	0	0	0	1:121	1:121	3:121	0	2:121	0	0	1:121	0
175	0	0	0	0	0	1:121	1:121	2:121	2:121	1:121	0	0	0	0
176	0	0	0	0	0	0	0	1:121	0	0	0	0	0	1:121
177	0	0	0	0	0	0	0	0	0	0	0	0	1:121	0
178	0	0	0	0	0	0	0	0	1:121	0	0	1:121	0	0
179	0	0	0	0	0	0	0	0	0	0	1:121	0	0	1:121

Tabel 4.9 Matrix Ternormalisasi

i/j	166	167	168	169	170	171	172	173	174	175	176	177	178	179
166	0	0	0	0,008	0	0	0	0	0	0	0	0	0	0
167	0	0,008	0	0	0	0	0	0	0	0	0	0	0	0
168	0	0	0	0,008	0	0	0	0	0	0	0	0	0	0
169	0,008	0,008	0,008	0,008	0,024	0,024	0,008	0	0	0	0	0	0	0
170	0,008	0	0	0,024	0,033	0,049	0,008	0,024	0	0	0	0	0	0
171	0	0	0	0,024	0,049	0,024	0,024	0,057	0,008	0,008	0	0	0	0
172	0	0	0	0,008	0,008	0,024	0,008	0,041	0,008	0,008	0	0	0	0
173	0	0	0	0	0,024	0,057	0,041	0,074	0,024	0,016	0,008	0	0	0
174	0	0	0	0	0	0,008	0,008	0,024	0	0,016	0	0	0,008	0
175	0	0	0	0	0	0,008	0,008	0,016	0,016	0,008	0	0	0	0
176	0	0	0	0	0	0	0	0,008	0	0	0	0	0	0,008
177	0	0	0	0	0	0	0	0	0	0	0	0	0,008	0
178	0	0	0	0	0	0	0	0	0,008	0	0	0,008	0	0
179	0	0	0	0	0	0	0	0	0	0	0,008	0	0	0,008

Setelah itu nilai kontras, homogeneity, entropy dan energy di hitung secara manual seperti di bawah ini:

a. Menghitung nilai *Contras*

$$\text{Contras} = ((166-169)^2(0,008) + (167-167)^2(0,008) + (168-169)^2(0,008) + (169-166)^2(0,008) + (169-167)^2(0,008) + \dots + (177-178)^2(0,008) + (178-174)^2(0,008) + (178-177)^2(0,008) + (179-176)^2(0,008) + (179-179)^2(0,008))$$

$$\begin{aligned}
&= 9*0.008 + 0*0.008 + 1*0.008 + 9*0.008 + 4*0.008 + 1*0.008 + 0*0.008 + \\
&\dots\dots\dots + 1*0.016 + 0*0.008 + 9*0.008 + 9*0.008 + 1*0.008 + 16*0.008 + \\
&1*0.008 + 9*0.008 + 0*0.008 \\
&= 0.072 + 0 + 0.008 + 0.072 + 0.032 + 0.008 + 0 + 0.024 + 0.096 + 0.072 + 0.128 \\
&+ \dots\dots\dots + 0.072 + 0.064 + 0.016 + 0 + 0.072 + 0.072 + 0.008 + 0.128 + 0.008 + \\
&0.072 + 0 \\
&= \mathbf{3.26}
\end{aligned}$$

b. Menghitung nilai *Energy*

$$\begin{aligned}
\text{Energy} &= ((0.008)^2) + ((0.008)^2) + ((0.008)^2) + ((0.008)^2) + ((0.008)^2) + ((0.008)^2) \\
&+ \dots\dots\dots + ((0.008)^2) + ((0.008)^2) + ((0.008)^2) + ((0.008)^2) + ((0.008)^2) \\
&= 0.000064 + 0.000064 + 0.000064 + 0.000064 + 0.000064 + 0.000064 + \\
&\dots\dots\dots + 0.000064 + 0.000064 + 0.000064 + 0.000064 + 0.000064 + 0.000064 \\
&= \mathbf{0.0256426}
\end{aligned}$$

c. Menghitung nilai *Homogenity*

$$\begin{aligned}
\text{Homogenity} &= ((0.008) / 1 + (166-169)) + ((0.008) / 1 + (167-167)) + ((0.008) / 1 \\
&+ (168-169)) + ((0.008) / 1 + (169-166)) + \dots\dots\dots + ((0.008) / 1 + (178-177)) + \\
&((0.008) / 1 + (179-176)) + ((0.008) / 1 + (179-179)) \\
&= ((0.008 + (-2)) + ((0.008 + 1))) + ((0.008 + 0)) + ((0.008 + 4)) + ((0.008 + 3)) \\
&+ \dots\dots\dots + ((0.008 + 5)) + ((0.008 + 2)) + ((0.008 + 4)) + ((0.008 + 1)) \\
&= (-0.004) + 1.008 + 0.008 + 4.008 + 3.008 + 2.008 + 1.008 + 0.024 + (-0.976) + \\
&\dots\dots\dots + (-1.992) + 0.008 + 5.008 + 2.008 + 4.008 + 1.008 \\
&= \mathbf{60.884}
\end{aligned}$$

d. Menghitung nilai Entropy

$$\text{Entropy} = (0.008.\log 0.008) + (0.008.\log 0.008) + (0.008.\log 0.008) + (0.008.\log 0.008) + \dots + (0.008.\log 0.008) + (0.008.\log 0.008) + (0.008.\log 0.008) + (0.008.\log 0.008)$$

$$= 0.008*(-2.0969) + 0.008*(-2.0969) + 0.008*(-2.0969) + 0.008*(-2.0969) + 0.008*(-2.0969) + \dots + 0.008*(-2.0969) + 0.008*(-2.0969) + 0.008*(-2.0969) + 0.008*(-2.0969)$$

$$= (-0.0167752) + (-0.0167752) + (-0.0167752) + (-0.0167752) + (-0.0167752) + \dots + (-0.0167752) + (-0.0167752) + (-0.0167752) + (-0.0167752) + (-0.0167752)$$

$$= -1.5790594$$

Tabel 4.10 Nilai Contrasts, Homogeneity, Entropy, Energy Data Training

NO	Con(X1)	Hom(X2)	Ent(X3)	Ene(X4)
0	3.26	60.884	-1.5790594	0.0256426
1	1765.884724676237	0.12651927057211085	0.17046116970896247	8.237399227472102
2	2075.1449285115655	0.17448921189027572	0.250670522988019	7.608970071480762
3	649.3340687299864	0.2096136177596333	0.28461956082325957	7.224287105284713
4	2132.376934767196	0.20772083940544062	0.2565871358756697	7.668312291337107
5	1926.421221892239	0.2498911546629404	0.3159214319591113	7.111139552103666
6	1760.896256004863	0.2738304602216356	0.3277769181034011	6.892764725373314
7	1751.0052490278808	0.3722937752134245	0.41691236180083224	6.016840215031475
8	1585.7500141856945	0.2017507101131902	0.26510535198991614	7.449144535727844

NO	Con(X1)	Hom(X2)	Ent(X3)	Ene(X4)
9	1978.3316294902477	0.37792004330103607	0.42182466372564376	5.873513281129708
10	1375.4803591141706	0.13779886223181745	0.20527551773819103	8.01508750717794
...				
70	2522.5216971910204	0.3525330837821684	0.4130538024908105	5.886120582401557
71	2538.231983994359	0.42261703965433645	0.4642929254374026	5.464576913533197
72	2171.49421378055	0.33818200974312607	0.40403210498609765	6.105597296738784
73	1873.9196340644648	0.16256131856490524	0.24410100905388124	7.437040367096292
74	1540.9514899504911	0.44560602062324467	0.49277510456903784	5.266901748399399
75	2034.4662272486717	0.34701861719650645	0.394752258486601	6.254684769488786
76	2177.938078132964	0.3764301755778235	0.43319739162215987	5.7048050122360126
77	2263.859652256625	0.2526345802627678	0.3320780328582289	6.615182725535626
78	1268.6184579527198	0.48520012813595725	0.5401342735844644	4.88312117957183
79	2576.5968876200072	0.2771451555879059	0.343708941164157	6.26527109301731

4.2.3 Klasifikasi

Klasifikasi pada data yang di gunakan yaitu berupa 80 dataset data Training dan 30 data Testing, yaitu terdiri dari 20 gambar mangga mentah, 20 mangga mengkal, 20 mangga matang dan 20 mangga busuk, dari 80 dataset data training mangga yang mana dari masing-masing gambar tersebut mempunyai nilai kontras, homogeneity, entropy, dan energy. Seperti pada tabel 4.11 dibawah ini:

Tabel 4.11 Data Training

No.	Nama File Data Training	Con(X1)	Hom(X2)	Ent(X3)	Ene(X4)	Y
0	IMG_130919	1765.8847	0.1265	0.1704	8.2373	Mangga Busuk
1	IMG_131122	2075.1449	0.1744	0.2506	7.6089	Mangga Busuk
2	IMG_131247	649.3340	0.20961	0.2846	7.2242	Mangga Busuk
3	IMG_101031	2132.3769	0.2077	0.2565	7.6683	Mangga Busuk
20	IMG_153119	890.7183	0.2012	0.3282	6.7187	Mangga Mengkal
21	IMG_153153	1122.6081	0.1730	0.2696	7.0747	Mangga Mengkal
22	IMG_153909	928.3352	0.1687	0.3050	6.6509	Mangga Mengkal
23	IMG_153947	1532.7422	0.2013	0.3101	6.7441	Mangga Mengkal
40	IMG_151734	1339.4818	0.2889	0.3862	6.3285	Mangga Matang
41	IMG_151805	1293.6978	0.2105	0.3158	6.9588	Mangga Matang
42	IMG_151826	1361.5801	0.1938	0.3097	7.0455	Mangga Matang
43	IMG_151835	1589.9966	0.2329	0.3413	6.6860	Mangga Matang
77	IMG_100714	1268.61845	0.4852	0.5401	4.8831	Mangga Mentah
78	IMG_100809	2576.5968	0.2771	0.3437	6.2652	Mangga Mentah
79	IMG_100999	1210.1814	0.2812	0.3520	6.5085	Mangga Mentah

Berikut ini adalah tahapan untuk menghitung nilai Euclidean distance dan menentukan nilai k. tahapan-tahapan tersebut seperti di bawah ini :

1. Tentukan parameter k
2. Hitung jarak antara data yang akan dievaluasi dengan semua pelatihan
3. Urutkan jarak yang terbentuk (urut naik)
4. Tentukan jarak terdekat sampai urutan k
5. Pasangkan kelas yang bersesuaian
6. Cari jumlah kelas dari tetangga yang terdekat dan tetapkan kelas tersebut sebagai kelas data yang akan dievaluasi.

Euclidean Distance

$$\begin{aligned}
 d &= \sqrt{(1765.8847 - 1989.7732)^2 + (0.1265 - 0.1936)^2 + (0.1704 - 0.2460)^2 + (8.2373 - 7.4774)^2} \\
 &= \sqrt{50126.0604 + 0.0045 + 0.0057 + 0.5774} \\
 &= \sqrt{50126.6481}
 \end{aligned}$$

$$= 223.8898$$

Tabel 4.12 Perbandingan Nilai

DATA TRAINIG	Perhitungan jarak <i>Euclidean Distance</i>	Nilai	Y
0	$d = \sqrt{50126.0604 + 0.0045 + 0.0057 + 0.5774}$	223.8898	BUSUK
1	$d = \sqrt{7288.3271 + 0.0003 + 0.0000 + 0.0172}$	85.3718	BUSUK

DATA TRAINIG	Perhitungan jarak <i>Euclidean Distance</i>	Nilai	Y
2	$d = \sqrt{1796777.249 + 0.0002 + 0.0014 + 0.0641}$	13404.4392	BUSUK
3	$d = \sqrt{20335.8152 + 0.0001 + 0.0001 + 0.0364}$	142.6038	BUSUK
20	$d = \sqrt{1207921.673 + 0.0000 + 0.0067 + 0.5756}$	1099.0551	MANGKAL
21	$d = \sqrt{751975.3107 + 0.0004 + 0.0005 + 0.1621}$	867.1651	MANGKAL
22	$d = \sqrt{1126650.628 + 0.0006 + 0.0034 + 0.6831}$	1601.4383	MANGKAL
23	$d = \sqrt{208877.335 + 0.0000 + 0.0041 + 0.5377}$	457.0315	MANGKAL
40	$d = \sqrt{422878.9049 + 0.0090 + 0.0196 + 1.3199}$	650.2924	MATANG

DATA TRAINIG	Perhitungan jarak <i>Euclidean Distance</i>	Nilai	Y
41	$d = \sqrt{\frac{484520.9625 + 0.0002}{0.0048 + 0.2689}}$	696.0755	MATANG
42	$d = \sqrt{\frac{394626.5709 + 0.0000}{0.0040 + 0.1865}}$	628.1932	MATANG
43	$d = \sqrt{\frac{159821.3299 + 0.0015}{0.0090 + 0.6263}}$	399.7773	MATANG
77	$d = \sqrt{\frac{520064.1734 + 0.0850}{0.0864 + 6.7303}}$	721.1595	MENTAH
78	$d = \sqrt{\frac{344361.9375 + 0.0069}{0.0095 + 1.4694}}$	586.8248	MENTAH
79	$d = \sqrt{\frac{607763.3746 + 0.0076}{0.0112 + 0.9387}}$	779.5924	MENTAH

Tabel 4.13 Hasil Pengurutan dari Nilai Terkecil sampai Terbesar

Data	Euclidean	Y
1	85.3718	BUSUK
3	142.6038	BUSUK
0	223.8898	BUSUK
43	399.7773	MATANG
23	457.0315	MENGKAL
78	586.8248	MENTAH
42	628.1932	MATANG
40	650.2924	MATANG
41	696.0755	MATANG
77	721.1595	MENTAH
79	779.5924	MENTAH
21	867.1651	MENGKAL
22	1601.4383	MENGKAL
20	1099.0551	MENGKAL
2	13404.4392	BUSUK

Tabel 4.14 Hasil Jumlah K yang Sesuai Data

DATA	Euclidean	Y
1	85.3718	BUSUK
3	142.6038	BUSUK
0	223.8898	BUSUK
43	399.7773	MATANG

DATA	Euclidean	Y
23	457.0315	MENGKAL
78	586.8248	MENTAH
42	628.1932	MATANG

Jumlah K = 7 Sesuai Yang Ditentukan Sebelumnya

Y	Jumlah Data
BUSUK	3
MATANG	2
MENGKAL	1
MENTAH	1

Tetangga Paling Banyak Muncul = BUSUK

Pada tabel 4.14 ini yaitu hasil nilai dari 30 dataset tersebut di dapatkan dengan jarak 5 dan arah 135^0 .

Tabel 4.15 Hasil Perhitungan Data Testing

No	Nama File Data testing	Contras	Homogeneity	Entropy	Energy	Y Actual	Y Klasifikasi
0	IMG_2020110440	1989.7732	0.1936	0.2460	7.4774	Busuk	Busuk
1	IMG_2020112531	2172.4197	0.2618	0.3147	7.0549	Busuk	Busuk
2	IMG_2020112507	1676.1641	0.2083	0.2670	7.5730	Busuk	Busuk
3	IMG_2020112505	1497.7772	0.2809	0.3916	6.7414	Busuk	Busuk
4	IMG_2020112855	4449.6895	0.2923	0.3332	6.8463	Busuk	Mentah
5	IMG_2020112805	1755.6047	0.1514	0.2133	7.9623	Busuk	Busuk

No	Nama File Data testing	Contras	Homogeneity	Entropy	Energy	Y Actual	Y Klasifikasi
6	IMG_2020112827	1513.6759	0.1438	0.2142	7.8710	Busuk	Busuk
7	IMG_2020112916	1656.8690	0.4228	0.4950	5.3725	Mengkal	Mengkal
8	IMG_2020111732	938.1978	0.1403	0.2729	7.1960	Mengkal	Mengkal
9	IMG_2020111723	1573.0135	0.1614	0.2677	7.2935	Mengkal	Mengkal
10	IMG_2020111716	1596.3842	0.1518	0.2590	7.3729	Mengkal	Mengkal
11	IMG_2020112946	989.3998	0.1603	0.2866	7.0825	Mengkal	Mengkal
12	IMG_2020111707	1027.5284	0.1471	0.2717	7.1441	Mengkal	Mengkal
13	IMG_2020111722	931.5743	0.1833	0.3193	6.8206	Mengkal	Mengkal
14	IMG_2020111741	909.8576	0.1386	0.2879	6.9787	Mengkal	Mengkal
15	IMG_2020111546	828.6980	0.2067	0.3377	6.8860	Mengkal	Mengkal
16	IMG_2020112900	1242.1360	0.2160	0.3277	6.8971	Matang	Matang
17	IMG_2020112929	1151.7820	0.4607	0.5236	5.1420	Matang	Matang
18	IMG_2020112306	1339.2092	0.1559	0.2562	7.0888	Matang	Matang
19	IMG_2020112916	1196.7179	0.4495	0.5139	5.1721	Matang	Matang
20	IMG_2020111753	1192.1547	0.2532	0.3580	6.4918	Matang	Matang
21	IMG_2020112556	1321.5151	0.2592	0.3876	6.5022	Matang	Matang
22	IMG_2020112910	1140.3071	0.3925	0.4566	5.7114	Matang	Matang
23	IMG_2020112941	1290.8530	0.1393	0.2366	7.4529	Matang	Matang
24	IMG_2020112505	1654.8840	0.4791	0.5133	5.1427	Mentah	Mengkal
25	IMG_2020112555	2734.0092	0.3075	0.3851	6.0758	Mentah	Mentah
26	IMG_2020112554	2234.8136	0.1659	0.2453	7.4686	Mentah	Mentah

No	Nama File Data testing	Contras	Homogeneity	Entropy	Energy	Y Actual	Y Klasifikasi
27	IMG_2020110612	553.7740	0.5880	0.6311	3.9765	Mentah	Mengkal
28	IMG_2020110649	1181.5357	0.1302	0.2311	7.6029	Mentah	Busuk
29	IMG_2020111549	1028.0069	0.1684	0.3297	6.9329	Mentah	Mengkal

4.3 Evaluasi

4.3.1 Confusion Matrix

Pada penelitian ini menggunakan confusion Matrix sebagai metode. Dalam perhitungan akurasi pada penerapan pengenalan tingkat kematangan buah mangga menggunakan tekstur warna. Evaluasi kinerja tingkat kematangan buah mangga di dasarkan dari jumlah pengujian objek.

Tabel 4.14 Hasil Data Testing

		Klasifikasi	
		TP	FP
Aktual	FN	1	1
	TN	25	3

Jumlah data yang diklasifikasi dengan benar = 25

Jumlah data yang diklasifikasi salah = 5

Akurasi = (Data Uji Dikenali)/(Data uji) x 100 % Akurasi = 25/30 x 100 % =
83.33 %

BAB V

PEMBAHASAN

5.1 Pembahasan Model

Tabel 5.1 hasil eksperimental

Nilai K	Arah GLCM	Jarak	Data training	Data testing	Nilai akurasi
1	0 °	1	80	30	36,66%
2	135 °	2	80	30	50%
3	45 °	3	80	30	60%
7	135 °	5	80	30	83,33%

Pada tahap pengujian metode terdapat kendala pada tahap pengenalan kematangan mangga, karena penggunaan sampel mangga harus diperhatikan dari kontras dan posisi dari data sampel, karena ada beberapa sampel data yang tidak bisa dikenali atau tidak terdeteksi. Arah dan jarak GLCM sangat berpengaruh untuk pengenalan kematangan buah mangga, hasil yang di dapatkan untuk metode GLCM dengan K-NN dengan arah 135° dan jarak 7 lebih tinggi tingkat akurasi. Hasil yang di dapatkan dari pengujian metode akurasi yang diperoleh sudah cukup baik, hasil dari perhitungan akurasi di atas bahwa semakin banyak nilai K semakin tinggi nilai akurasi yang diperoleh, dan sebaliknya. jumlah K 7 lebih tinggi tingkat akurasi tingkat kematangan buah mangga menggunakan tekstur warna, ketimbang jumlah K 2. Dari hasil perhitungan akurasi untuk pengujian Klasifikasi mendapatkan nilai sebesar 83,33%. Pada penelitian ini penggunaan metode K-NN kurang baik untuk tahap klasifikasi.

5.2 Alat Penelitian

Alat yang digunakan pada penelitian ini (Kamera) Menggunakan Handphone Redmi Note 8 Pro

5.3 Pembahasan Sistem

1. Tampilan Menu Utama



Gambar 5.1 Tampilan Menu Utama

2. Tampilan Browse Data Training

Halaman ini adalah tampilan jika pengguna menekan tombol browse untuk data training.

3. Tampilan Browse Data Testing

Halaman ini adalah tampilan jika pengguna menekan tombol browse untuk data testing. Tampilan ini akan muncul saat pengguna memilih/mengklik *browser* pada *Image Testing*, dan pengguna akan memilih folder yang tersimpan gambar.

4. Tampilan Setelah Pemilihan Folder *Training* dan data *Testing*

Tampilan ini akan memunculkan nama folder yang telah dipilih pengguna.

5. Tampilan Pemasukan Nilai GLCM

Tampilan ini akan muncul saat pengguna telah memasukan nilai dan memilih perhitungan pada GLCM.

6. Tampilan Pemasukan Nilai K

Tampilan ini akan muncul saat pengguna telah memasukan nilai K.

7. Tampilan Hasil Tingkat Kematangan Buah Mangga

Tampilan ini akan muncul saat pengguna mengklik/memilih tombol atau button proses setelah menginput data traning, data testing, arah dan jarak GLCM serta nilai k.

8. Tampilan Hasil Klasifikasi Dan Nilai Akurasi

Tampilan ini akan muncul bersamaan dengan tampilan Tingkat Kematangan Buah Mangga.



Gambar 5.2 Hasil Klasifikasi

BAB VI

KESIMPULAN

6.1 Kesimpulan

Berdasarkan rumusan masalah, maka tujuan penelitian ini yaitu Mengetahui hasil penerapan GLCM dan K-NN untuk proses Tingkat Kematangan Buah Mangga. Menggunakan tekstur warna dalam tingkat akurasi kematangannya maka dapat ditarik suatu kesimpulan bahwa Pada percobaan model K-NN, dengan jumlah $K = 1$, dan Arah GLCM = 0° , dengan jarak = 1, maka mendapatkan nilai akurasi tertinggi Sebesar = 36,66%, dengan data training = 80, dan Data Testing = 30. Sedangkan pada percobaan model K-NN, dengan jumlah $K = 2$, dan Arah GLCM = 135° , dengan jarak = 2, maka mendapatkan nilai akurasi tertinggi sebesar = 50%, dengan data training = 80, dan Data Testing = 30. Dan pada percobaan model K-NN, dengan jumlah $K = 3$, dan Arah GLCM = 45° , dengan jarak = 3, maka mendapatkan nilai akurasi tertinggi Sebesar = 60%, dengan data training = 80, dan Data Testing = 30, dan percobaan terakhir dengan jumlah $K = 7$, dan Arah GLCM = 135° , dengan jarak = 5, maka mendapatkan nilai akurasi tertinggi sebesar = 83,33%, dengan data training = 80 dan data testing = 30.

6.2 Saran

Setelah melakukan Penelitian Tingkat Kematangan Buah Mangga, ada beberapa saran yang perlu di perhatikan untuk mencapai tujuan yang di harapkan, yaitu sebagai berikut:

1. Penulis berharap untuk penelitian selanjutnya agar mendapatkan hasil akurasi yang lebih baik.
2. Penulis berharap agar hasil segmentasi berikutnya menjadi lebih baik lagi.
3. Penulis berharap agar pengambilan gambar bisa di optimalkan lagi agar hasil yang lebih baik
4. Penulis berharap untuk penelitian selanjutnya menggunakan kamera selain xiami, agar bisa mendapatkan hasil lebih baik lagi.

DAFTAR PUSTAKA

- [1] C. Bagus and M. Imron, "Klasifikasi Buah Mangga Berdasarkan Tingkat Kematangan Menggunakan Least-Squares Support Vector Machine," *Explor. IT J. Keilmuan dan Apl. Tek. Inform.*, vol. 10, no. 2, pp. 1–8, 2018, doi: 10.35891/explorit.v10i2.1255.
- [2] A. Patriot, S. Pamungkas, N. Nafi, and N. Q. Nawafilah, "K-NN Klasifikasi Kematangan Buah Mangga Manalagi Menggunakan $L * A * B$ dan Fitur Statistik," vol. 4, pp. 1–8, 2019.
- [3] M. Widyaningsih, "Identifikasi Kematangan Buah Apel Dengan Gray Level Co-Occurrence Matrix (GLCM)," *J. SAINTEKOM*, vol. 6, no. 1, p. 71, 2017, doi: 10.33020/saintekom.v6i1.7.
- [4] I. Machroz *et al.*, "Klasifikasi Kematangan Mangga Menggunakan Metode Jaringan Syaraf Tiruan Levenberg Marquardt," *Semin. Nas. Teknol. Inf.*, vol. 4, pp. 55–59, 2017.
- [5] P. Dasar and P. Pengomposan, "UNIVERSITAS SUMATERA UTARA Medan," 2014.
- [6] A. Suryowinoto and A. Hamid, "Penggunaan Pengolahan Citra Digital dengan Algoritma Edge Detection dalam Mengidentifikasi Kerusakan Kontur Jalan," *Semin. Nas. Sains dan Teknol. Terap. V*, pp. 149–154, 2017.
- [7] C. N. Santi, "Mengubah Citra Berwarna Menjadi Gray-Scale dan Citra biner," *Teknol. Inf. Din.*, vol. 16, no. 1, pp. 14–19, 2011.
- [8] O. N. Shpakov and G. V. Bogomolov, "Technogenic activity of man and local sources of environmental pollution," *Stud. Environ. Sci.*, vol. 17, no. C, pp. 329–332, 1981, doi: 10.1016/S0166-1116(08)71924-1.
- [9] R. Widodo, A. W. Widodo, and A. Supriyanto, "Pemanfaatan Ciri Gray Level Co-Occurrence Matrix (GLCM) Citra Buah Jeruk Keprok (Citrus

- reticulata Blanco) untuk Klasifikasi Mutu,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 5769–5776, 2018.
- [10] N. W. Roberto, “Penerapan Metode K-Nearest Neighbor dalam Identifikasi Kesegaran Ikan,” 2019.
- [11] G. T. Situmorang, A. W. Widodo, and M. A. Rahman, “Penerapan Metode Gray Level Cooccurrence Matrix (GLCM) untuk Ekstraksi Ciri pada Telapak Tangan,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 5, pp. 4710–4716, 2019.
- [12] T. Ismail, “Analisis Keseluruhan Sudut Co-Occurrence Matrix pada GLCM dalam Mengklasifikasi Citra,” pp. 1–55, 2018.
- [13] W. Yustanti, “Algoritma K-Nearest Neighbour untuk Memprediksi Harga Jual Tanah,” *J. Mat. Stat. dan komputasi*, vol. 9, no. 1, pp. 57–68, 2012.
- [14] H. Leidiyana, “Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor,” *J. Penelit. Ilmu Komputer, Syst. Embed. Log.*, vol. 1, no. 1, pp. 65–76, 2013.
- [15] subairi subairi, R. Rahmadwati, and E. Yudaningtyas, “Implementasi Metode k-Nearest Neighbor pada Pengenalan Pola Tekstur Citra Saliva untuk Deteksi Ovulasi,” *J. EECCIS*, vol. 12, no. 1, pp. 9–14, 2018.

CODING PROGRAM

1. Pembuatan Form

```

import tkinter as tk
import tkinter.filedialog as fdialog
import tkinter.ttk as ttk
from PIL import Image, ImageTk
import coba
import cv2

class Aplication():
    def __init__(self, parent):
        self.parent = parent
        self.frame ()
        self.container()
        self.widget()

    def frame(self):
        self.mframe = tk.LabelFrame(self.parent, background='white', width
= 1366,
                                height=900, borderwidth=1, relief='sunken')
        self.mframe.grid(row=0, column=0, sticky='NSEW', padx=5,
pady=5)
        self.mframe.grid_propagate(0)
    def container(self):
        self.topcontainer = tk.Frame(self.mframe, width=1366, height=100,
                                borderwidth=1, background='lightseagreen',
relief='raised')
        self.topcontainer.grid(row=0, column=0, columnspan=4,
sticky='NSEW')
        self.topcontainer.grid_propagate(0)

```

```

#input gambar
self.leftcontainer = tk.Frame(self.mframe, height=160,width=500,
                               borderwidth=2, background='whitesmoke',
relief='raised')
self.leftcontainer.grid(row=1, column=0, sticky='NSEW')
self.leftcontainer.grid_propagate(0)

#GLCM
self.leftcontainer1 = tk.Frame(self.mframe, height=160,width=450,
                               borderwidth=1, background='ghost white',
relief='raised')
self.leftcontainer1.grid(row=1, column=1, sticky='NSEW')
self.leftcontainer1.grid_propagate(0)

#KNN
self.leftcontainer2 = tk.Frame(self.mframe, height=150, width=450,
                               borderwidth=1, background='ghost white',
relief='raised')
self.leftcontainer2.grid(row=1, column=2, sticky='NSEW')
self.leftcontainer2.grid_propagate(0)

#PROSES
self.topcontainer1 = tk.Frame(self.mframe, height=50, width=20,
                               borderwidth=2, background='ghost white',
relief='raised')
self.topcontainer1.grid(row=2, column=0,columnspan=3,
sticky='NSEW')
self.topcontainer1.grid_propagate(0)

#Hasil gambar

```

```

self.leftcontainer4 = tk.Frame(self.mframe,height=250,width=200,
                                borderwidth=1, background='darkgreen',
relief='raised')
    self.leftcontainer4.grid(row=3, column=0, columnspan=4,
sticky='NSEW')
    self.leftcontainer4.grid_propagate(0)

#hasil akhir
self.leftcontainer5 = tk.Frame(self.mframe,height=150,
                                borderwidth=4, background='lightseagreen',
relief='raised')
    self.leftcontainer5.grid(row=4, column=0, columnspan=4,
sticky='NSEW')
    self.leftcontainer5.grid_propagate(0)

#image testing
self.leftcontainer6 = tk.Frame(self.mframe, height=250,width=200,
                                borderwidth=1, background='darkgreen',
relief='raised')
    self.leftcontainer6.grid(row=3, column=1, sticky='NSEW')
    self.leftcontainer6.grid_propagate(0)

#image grayscale
self.leftcontainer7 = tk.Frame(self.mframe, height=250,width=200,
                                borderwidth=1, background='darkgreen',
relief='raised')
    self.leftcontainer7.grid(row=3, column=2, sticky='NSEW')
    self.leftcontainer7.grid_propagate(0)

def widget(self):
    ltrain = tk.Label(self.leftcontainer, text="Input Gambar",

```

```

        relief='ridge', width=15, background='white')
ltrain.grid(row=0, column=0, padx=1, pady=2, sticky="W")

ltrain = tk.Label(self.leftcontainer1, text="Gray Level Co-Occurrence
Matrix",
        relief='ridge', width=25, background='white')
ltrain.grid(row=0, column=0, padx=1, pady=2, sticky="W")

ltrain = tk.Label(self.leftcontainer2, text="K - Nearest Neighbor",
        relief='ridge', width=20, background='white')
ltrain.grid(row=0, column=0, padx=1, pady=2, sticky="W")

#frame
#logo
self.PathTestingImg = 'E:/skripsi/python/design/logo.jpg'
self.imT = Image.open(self.PathTestingImg)
self.imT = ImageTk.PhotoImage(self.imT)
self.label1 = tk.Label(self.topcontainer, width=1366, height=100,
bg="white", image=self.imT)
self.label1.pack(side="top")

#frame1
#label data training
self.label1 = tk.Label(self.leftcontainer, text="Image Training",
font="Times", relief="ridge", width=1, background="seagreen")
self.label1.grid(row=1, column=0, sticky="NSEW", padx=10,
pady=2)

#textbox data training
self.valtrain = tk.StringVar(self.leftcontainer, value="")

```

```

self.txttrain= tk.Entry(self.leftcontainer, textvariable =self.valtrain,
relief='sunken', width=35)

self.txttrain.grid(row=1, column=1, padx=5, pady=15)


#button browse training
self.path2 = tk.Button(self.leftcontainer, text="Browse",
font="Times", width=6, background="seagreen",
command=lambda:self.valtrain.set(fdialog.askdirectory()))
self.path2.grid(row=1, column=5, padx=5)


#label data testing
self.label2 = tk.Label(self.leftcontainer, text="Image Testing",
font="Times", relief="ridge", width=1, background="seagreen")
self.label2.grid(row=2, column=0, sticky="NSEW", padx=10,
pady=2)


#textbox data testing
self.valtrainn = tk.StringVar(self.leftcontainer, value="")
self.txttest= tk.Entry(self.leftcontainer, textvariable = self.valtrainn,
relief='sunken', width=35)
self.txttest.grid(row=2, column=1, padx=5, pady=15)


#button browse testing
self.path2 = tk.Button(self.leftcontainer, text="Browse",
font="Times", width=6, background="seagreen",
command=lambda:self.valtrainn.set(fdialog.askopenfilename()))
self.path2.grid(row=2, column=5, padx=5)


#frame2
#COMBOBOX GLCM
arah = tk.IntVar(self.leftcontainer1)

```

```

self.label = tk.Label(self.leftcontainer1, text="GLCM", width=12,
font="Times", relief="ridge", background="seagreen")
self.label.grid(row=1, column=0, padx=10 , pady=2)

#arah GLCM
arah = tk.StringVar(self.leftcontainer1, value="")
self.cbarahglcm = ttk.Combobox(self.leftcontainer1, textvariable =
arah, width=10)
self.cbarahglcm['value'] = (0, 45, 90, 135)
self.cbarahglcm.current(0)
self.cbarahglcm.grid(row=1, column=1, sticky="NSEW", pady=10)

#jarak GLCM
self.label1 = tk.Label(self.leftcontainer1, text="Jarak", width=12,
font="Times", relief="ridge", background="seagreen")
self.label1.grid(row=2, column=0, padx=10 , pady=2)

jarak = tk.StringVar(self.leftcontainer1, value="")
self.tbjarak = tk.Entry(self.leftcontainer1, textvariable = jarak,
width=29)
self.tbjarak.grid(row=2, column=1, sticky="NSEW", padx=1,
pady=5 )

#frame 3
#klasifikasi KNN
#label KNN
self.label = tk.Label(self.leftcontainer2, text="Nilai K",
font="Times", relief="ridge", width=12, background="seagreen")
self.label.grid(row=2, column=0, sticky="NSEW", padx=8, pady=5)

#textbox knn

```



```

        nilai2 = tk.StringVar(self.leftcontainer2, value="")
        self.textboxKnn = tk.Entry(self.leftcontainer2, textvariable = nilai2,
width=30)
        self.textboxKnn.grid(row=2, column=1, sticky="NSEW", pady=5)

        #frame 4
        #proses
        self.button_proses = tk.Button(self.topcontainer1, text="PROSES",
width=20, height=1,
                                command=self.proses)
        self.button_proses.grid(row=2, column=0 ,padx=650,pady=5,
sticky="NW")

#    #frame 5
#    #output
        #label hasil Image Asli
        self.label = tk.Label(self.leftcontainer4, text="Image Testing",
font="Times", background="lightgreen", fg="black")
        self.label.grid(row=1, column=1, padx=200, pady=17)
        #images Hasil Testing
        self.noimage1 ='E:/skripsi/python/design/gmbar1.png'
        self.noimageTest1 = Image.open(self.noimage1)
        self.noimageTest1 = ImageTk.PhotoImage(self.noimageTest1)
        self.labelTest1 = tk.Label(self.leftcontainer4, bg = "gray",
image=self.noimageTest1)
        self.labelTest1.grid(row=2, column=1, padx=195, pady=1,
sticky="W")

        #label Grayscale
        self.label = tk.Label(self.leftcontainer6, text="Grayscale",
font="Times", background="lightgreen", fg="black")

```

```

self.label.grid(row=1, column=3, padx=160, pady=17)
#images Hasil Grayscale
self.noimage2 ='E:/skripsi/python/design/gmbar1.png'
self.noimageTest2 = Image.open(self.noimage2)
self.noimageTest2 = ImageTk.PhotoImage(self.noimageTest2)
self.labelTest2 = tk.Label(self.leftcontainer6, bg = "gray",
image=self.noimageTest2)
self.labelTest2.grid(row=2, column=3, padx=160, pady=1,
sticky="N")

#label Segmentasi
self.label = tk.Label(self.leftcontainer7, text="Segmentasi",
font="Times", background="lightgreen", fg="black")
self.label.grid(row=1, column=2, padx=150, pady=17)
#images Hasil Segmentasi
self.noimage3 ='E:/skripsi/python/design/gmbar1.png'
self.noimageTest3 = Image.open(self.noimage3)
self.noimageTest3 = ImageTk.PhotoImage(self.noimageTest3)
self.labelTest3 = tk.Label(self.leftcontainer7, bg = "gray",
image=self.noimageTest3)
self.labelTest3.grid(row=2, column=2, padx=150, pady=1,
sticky="S")
#
# #frame 6
#label keterangan hasil output
self.label12 = tk.Label(self.leftcontainer5, text="OUTPUT",
width=25, height=5, font="Times", background="lightgreen")
self.label12.grid(row=0, column=0, padx=600, pady=5)

# output = tk.StringVar(self.leftcontainer5, value="")

```

```

# self.output = tk.Entry(self.leftcontainer5, textvariable = output,
width=50)
# self.output.grid(row=0, column=1, sticky="NSEW", padx=8,
pady=5 )

def proses(self):
# pass
# get image training
folder_train = self.txttrain.get()
# get image testing
folder_test = self.txttest.get()
# glcm
neigh= self.cbarahglcm.get()
angle = self.tbjarak.get()

# textbox iterasi
K = self.textboxKnn.get()

# hasil output akhir
a = coba.analisis()
hasil, img_segmen, gray = a.model_algoritma(folder_train,
folder_test, neigh, angle, K)
self.label12.configure(text = hasil)

# hasil gambar
# hasil image 1
imgTest = Image.open(self.txttest.get())
imgTest = imgTest.resize((165, 165), Image.ANTIALIAS)
imgTest = ImageTk.PhotoImage(imgTest)
self.labelTest1.configure(image=imgTest)

```

```

self.labelTest1.image_names = imgTest

#hasil image 2
imgGray = Image.fromarray(gray)
imgGray = imgGray.resize((165, 165), Image.ANTIALIAS)
imgGray = ImageTk.PhotoImage(imgGray)
self.labelTest2.configure(image=imgGray)
self.labelTest2.image_names = imgGray

#hasil image 3
imgsegmen = cv2.cvtColor(img_segmen, cv2.COLOR_BGR2RGB)
imgsegmen = Image.fromarray(imgsegmen)
imgsegmen = imgsegmen.resize((165,165), Image.ANTIALIAS)
imgsegmen = ImageTk.PhotoImage(imgsegmen)
self.labelTest3.configure(image=imgsegmen)
self.labelTest3.image_names = imgsegmen

def main():
    root = tk.Tk()
    root.geometry("1024x520")
    root.configure(background='green')
    app = Aplication(root)
    app.parent.title("Aplikasi")
    root.mainloop()

if __name__ == '__main__':
    main()

```

2. Image Processing

```
import cv2
from imutils import paths
import numpy as np
import os
from scipy.stats import entropy
from skimage import feature
from sklearn.neighbors import KNeighborsClassifier
```

class analisis:

```
    def segmentasi(self, img):
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        blur = cv2.GaussianBlur(gray,(5,5),0)
        ret, alpha = cv2.threshold(np.uint8(blur), 0, 255,
cv2.THRESH_BINARY_INV+cv2.THRESH_OTSU)
        alpha[alpha==255] = 1
        rect = cv2.boundingRect(alpha)
        r = img[:, :, 0] * alpha
        g = img[:, :, 1] * alpha
        b = img[:, :, 2] * alpha
        hasil = cv2.merge((r,g,b))
        x,y,w,h = rect
        result = hasil[y:y+h, x:x+w]
        return result, gray

    def glcmfitur(self, img, neigh, angle):
        img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        matrix_glcm = feature.greycomatrix(img, [neigh], [angle],
symmetric=False, normed=True)
        contrast = feature.greycomprops(matrix_glcm, 'contrast')[0,0]
        homogeneity = feature.greycomprops(matrix_glcm, 'homogeneity')[0,0]
```

```

        ent = entropy(matrix_glcmm.flatten())
        energy = feature.greycomprops(matrix_glcmm, 'energy')[0,0]
        hasil_fitur = [contrast, energy, homogeneity, ent]
        return hasil_fitur
# def glcmfitur(self, img, neigh, angle):
#     glcm = feature.greycomatrix(img, [neigh], [angle],
# symmetric=False,
#                                     normed= True)
###     asm = feature.greycomprops(glcm, 'ASM')[0,0]
#     con = feature.greycomprops(glcm, 'contrast')[0,0]
#     ene = feature.greycomprops(glcm, 'energy')[0,0]
#     hmg = feature.greycomprops(glcm, 'homogeneity')[0,0]
###     cor = feature.greycomprops(glcm, 'correlation')[0,0]
#     ent = entropy(glcm.flatten())
#     return [con, ene, hmg, ent]

def get_Img_Path(self, folderPath):
    imPath = []
    for path in paths.list_images(folderPath):
        imPath.append(path)
    return imPath

def get_label(self, folderPath):
    label = []
    imPath = self.get_Img_Path(folderPath)
    for item in imPath:
        label.append(os.path.split(os.path.dirname(item))[-1])
    return label

def training_data(self, folderPath, neigh, angle):
    pathList = self.get_Img_Path(folderPath)

```

```

images = []
for imPath in pathList:
    img = cv2.imread(imPath)
    images.append(img)
data_train = []
for img in images:
    hasil_segmen, gray = self.segmentasi(img)
    ekstraksi = self.glcmfitur(hasil_segmen, neigh, angle)
    data_train.append(ekstraksi)
label_train = self.get_label(folderPath)
return data_train, label_train

def testing_data(self, img, neigh, angle):
    data_test = []
    img = cv2.imread(img)
    hasil_segmen, gray = self.segmentasi(img)
    hasil_ekstrak_glcm = self.glcmfitur(hasil_segmen, neigh, angle)
    data_test.append(hasil_ekstrak_glcm)
    return data_test, hasil_segmen, gray

def model_algoritma(self, folder_train, folder_test, neigh, angle, K):
    x_train, y_train = self.training_data(folder_train, neigh, angle)
    x_test, hasil_segmen, gray = self.testing_data(folder_test, neigh,
angle)
    clf = KNeighborsClassifier(n_neighbors=int(K), weights='uniform',
                             algorithm='auto', metric='euclidean')
    clf.fit(x_train, y_train)
    final_hasil = clf.predict(x_test)
    return final_hasil, hasil_segmen, gray

```

DAFTAR RIWAYAT HIDUP



Nama : Mohammad Rezky B. Pakaya
Tempat, Tanggal Lahir : Gorontalo, 15 Agustus 1998
Pekerjaan : Mahasiswa
Email : nodarpakaya@gmail.com

Daftar Riwayat Hidup :

1. Tahun 2010, menyelesaikan Pendidikan di Sekolah Dasar Negeri 83 Kota Tengah, Kecamatan Kota Tengah, Kota Gorontalo.
2. Tahun 2013, menyelesaikan Pendidikan di Sekolah Menengah Pertama Negeri 11 Gorontalo, Kecamatan Kota Timur, Kota Gorontalo.
3. Tahun 2016, menyelesaikan Pendidikan di Sekolah Menengah Kejuruan Negeri 1 Gorontalo, Kecamatan Sipatana, Kota Gorontalo.
4. Tahun 2016, telah diterima menjadi Mahasiswa di Perguruan Tinggi Swasta Universitas Ichsan Gorontalo.

GAMBAR DATA TRAINING

No.	Data Training	Nama File	No.	Data Training	Nama File
0		IMG_130919	40		IMG_151734
1		IMG_131122	41		IMG_151805
2		IMG_131247	42		IMG_151826
3		IMG_101031	43		IMG_151835
20		IMG_153119	77		IMG_100714
21		IMG_153153	78		IMG_100809
22		IMG_153909	79		IMG_100999
23		IMG_153947			

GAMBAR DATA TESTING

No	Data testing	Nama File	No	Data testing	Nama File
0		IMG_2020110440	5		IMG_2020112805
1		IMG_2020112531	6		IMG_2020112827
2		IMG_2020112507	7		IMG_2020112916
3		IMG_2020112505	8		IMG_2020111732
4		IMG_2020112855	9		IMG_2020111723

No	Data testing	Nama File	No	Data testing	Nama File
10		IMG_2020111716	15		IMG_2020111546
11		IMG_2020112946	16		IMG_2020112900
12		IMG_2020111707	17		IMG_2020112929
13		IMG_2020111722	18		IMG_2020112306
14		IMG_2020111741	19		IMG_2020112916

No	Data testing	Nama File	No	Data testing	Nama File
20		IMG_2020111753	25		IMG_2020112555
21		IMG_2020112556	26		IMG_2020112554
22		IMG_2020112910	27		IMG_2020110612
23		IMG_2020112941	28		IMG_2020110649
24		IMG_2020112505	29		IMG_2020111549



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO**

UPT. PERPUSTAKAAN PUSAT

SURAT KEPUTUSAN MENDIKS RI NO. 84/D/0/2001

Jln. Achmad Nadjamuddin No.17 Telp. (0435) 829975 Fax. (0435) 829976 Gorontalo

SURAT KETERANGAN BEBAS PUSTAKA

No.013/perpus_fikom/XII/2020

Perpustakaan Senin, 07 Desember 2020 Fakultas Ilmu komputer (FIKOM) Universitas Ichsan Gorontalo dengan ini menerangkan bahwa :

Nama : Mohammad Rezky B. Pakaya
Nim : T3116101
No anggota : M202013

Terhitung sejak tanggal 07 Desember 2020, dinyatakan telah bebas dari pinjaman buku dan koleksi lainnya diperpustakaan Fakultas Ilmu komputer.

Demikian keterangan ini di buat untuk di pergunakan sebagaimana mestinya.

Gorontalo, 07 Desember 2020
Kepala Perpustakaan
Fakultas Ilmu Komputer

Apriyanto Alhamad, M.Kom
NIDN 09240486



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN
(UNISAN) GORONTALO**

SURAT KEPUTUSAN MENDIKNAS RI NOMOR 84/D/O/2001
Jl. Achmad Nadjamuddin No. 17 Telp (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT REKOMENDASI BEBAS PLAGIASI

No. 0698/UNISAN-G/S-BP/XII/2020

Yang bertanda tangan di bawah ini:

Nama : Sunarto Taliki, M.Kom
NIDN : 0906058301
Unit Kerja : Pustikom, Universitas Ichsan Gorontalo

Dengan ini Menyatakan bahwa :

Nama Mahasisw : MOHAMMAD REZKY BAGUSYCAHPUTRA PAKAY
NIM : T3116101
Program Studi : Teknik Informatika (S1)
Fakultas : Fakultas Ilmu Komputer
Judul Skripsi : Pengenalan Tingkat Kematangan Buah Mangga Menggunakan Metode GLCM dan KNN

Sesuai dengan hasil pengecekan tingkat kemiripan skripsi melalui aplikasi Turnitin untuk judul skripsi di atas diperoleh hasil Similarity sebesar 21%, berdasarkan SK Rektor No. 237/UNISAN-G/SK/IX/2019 tentang Panduan Pencegahan dan Penanggulangan Plagiarisme, bahwa batas kemiripan skripsi maksimal 35% dan sesuai dengan Surat Pernyataan dari kedua Pembimbing yang bersangkutan menyatakan bahwa isi softcopy skripsi yang diolah di Turnitin SAMA ISINYA dengan Skripsi Aslinya serta format penulisannya sudah sesuai dengan Buku Panduan Penulisan Skripsi, untuk itu skripsi tersebut di atas dinyatakan BEBAS PLAGIASI dan layak untuk diujikan.

Demikian surat rekomendasi ini dibuat untuk digunakan sebagaimana mestinya.

Gorontalo, 05 Desember 2020
Tim Verifikasi,



Sunarto Taliki, M.Kom
NIDN. 0906058301

Tembusan :

1. Dekan
2. Ketua Program Studi
3. Pembimbing I dan Pembimbing II
4. Yang bersangkutan
5. Arsip



SKRIPSI_1_T3116101_MOHAMMAD REZKY B PAKAYA.docx

Dec 4, 2020

6808 words / 44653 characters

T3116101 MOHAMMAD REZKY B PAKAYA

PENGENALAN TINGKAT KEMATANGAN BUAH MANGGA MEN...

Sources Overview

21%

OVERALL SIMILARITY

1	id.123dok.com	3%
2	www.scribd.com	2%
3	jurnal.yudharta.ac.id	1%
4	j-ptlik.ub.ac.id	1%
5	pt.scribd.com	<1%
6	strmikplk.ac.id	<1%
7	repository.ipb.ac.id	<1%
8	media.neliti.com	<1%
9	ojs.uniska-bjm.ac.id	<1%
10	docplayer.info	<1%
11	milyunima.wordpress.com	<1%
12	jurnal.fikom.uml.ac.id	<1%
13	repositorium.ac.id	<1%
14	123dok.com	<1%
15	repository.usu.ac.id	<1%
16	www.idosi.org	<1%

17	semantika.polgan.ac.id INTERNET	<1%
18	Rozzi Kesuma Dinata, Hafizal Akbar, Novia Hasdyna. "Algoritma K-Nearest Neighbor dengan Euclidean Distance dan Manhattan Dista... CROSSREF	<1%
19	ejournal.uin-suska.ac.id INTERNET	<1%
20	garuda.ristekdikti.go.id INTERNET	<1%
21	journal.unhas.ac.id INTERNET	<1%
22	seminar.ilkom.unsri.ac.id INTERNET	<1%
23	digilib.uinsgd.ac.id INTERNET	<1%
24	mafiadoc.com INTERNET	<1%
25	journal.unusida.ac.id INTERNET	<1%
26	www.coursehero.com INTERNET	<1%
27	Sukemi, Edi Sukrisno. "Identification using the K-Means Clustering and Gray Level Co-occurrence Matrix (GLCM) At Maturity Fruit Oil H... CROSSREF	<1%
28	ejournal.unmus.ac.id INTERNET	<1%
29	repositori.usu.ac.id INTERNET	<1%
30	ejournal.nusamandiri.ac.id INTERNET	<1%
31	repository.uin-suska.ac.id INTERNET	<1%
32	diansulistyo.wordpress.com INTERNET	<1%
33	id.scribd.com INTERNET	<1%
34	eprints.sinus.ac.id INTERNET	<1%
35	jurnal.untan.ac.id INTERNET	<1%
36	www.fikom-unisan.ac.id INTERNET	<1%
37	mustakimtelematika.wordpress.com INTERNET	<1%
38	text-id.123dok.com INTERNET	<1%
39	johannesalmatupang.wordpress.com INTERNET	<1%
40	repo.unand.ac.id INTERNET	<1%

41	es.scribd.com	INTERNET	<1%
42	zombiedoc.com	INTERNET	<1%
43	digilib.unila.ac.id	INTERNET	<1%
44	eprints.dinus.ac.id	INTERNET	<1%
45	putuadisusanta.wordpress.com	INTERNET	<1%
46	vanessavaya.wordpress.com	INTERNET	<1%

Excluded search repositories:

- Submitted Works

Excluded from Similarity Report:

- None

Excluded sources:

- None



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER**

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

Berita Acara Perbaikan/Revisi Ujian SKRIPSI

Pada hari ini, Rabu 09-Desember-2020, Pukul 13.00-15.00 Wita. Telah dilaksanakan Ujian SKRIPSI mahasiswa Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Nama : Mohammad Rezky B. Pakaya
 Nim : T3116101
 Pembimbing I : Rezqiwati Ishak, M.Kom
 Pembimbing II : Sudirman S. Panna, M.Kom
 Judul SKRIPSI : Pengenalan Tingkat Kematangan Buah Mangga Menggunakan Metode KNN

Oleh Komite Seminar sebagai berikut :

No	Komite Seminar	Status	Tanda Tangan
1	Zohrahayaty, M.Kom	Ketua	
2	Budy Santoso, S.Kom, M.Eng	Anggota	
3	Kartika Chandra Pelangi, M.Kom	Anggota	
4	Rezqiwati Ishak, M.Kom	Anggota	
5	Sudirman S. Panna, M.Kom	Anggota	