# PEnerapan Metode *Gray Level Co-Occurrence Matrix* Dan *K-Nearest Neighbor* untuk peNGENALAN KEMATANGAN BUAH PISANG BERDASARKAN tekstur WARNA

**Oleh**

**AMIN DWI SATRIO**

**T3115144**

**SKRIPSI**

**Untuk memenuhi salah satu syarat ujian**

**guna memperoleh gelar Sarjana**

****

**PROGRAM SARJANA**

**TEKNIK INFORMATIKA**

**UNIVERSITAS ICHSAN GORONTALO**

**GORONTALO**

**2019**

# pERSETUJUAN SKRIPSI

# PEnerapan Metode *Gray Level Co-Occurrence Matrix* Dan *K-Nearest Neighbor* untuk peNGENALAN KEMATANGAN BUAH PISANG BERDASARKAN tekstur WARNA

**Oleh**

**AMIN DWI SATRIO**

**T3115144**

**SKRIPSI**

**Untuk memenuhi salah satu syarat ujian**

**guna memperoleh gelar Sarjana**

**dan telah disetujui Oleh Tim Pembimbing**

**Gorontalo, 30 November 2019**

|  |  |
| --- | --- |
| **Pembimbing I**  **Irvan Abraham Salihi, M.Kom**  **NIDN. 0928028101** | **Pembimbing II**  **Apriyanto Alhamad, M.Kom**  **NIDN. 0924048601** |

# PENGESAHAN SKRIPSI

# PEnerapan Metode *Gray Level Co-Occurrence Matrix* Dan *K-Nearest Neighbor* untuk peNGENALAN KEMATANGAN BUAH PISANG BERDASARKAN tekstur WARNA

Oleh

AMIN DWI SATRIO

T3115144

Diperiksa oleh Panitia Ujian Sastra Satu (S1)  
Universitas Ichsan Gorontalo

|  |  |  |
| --- | --- | --- |
| 1. | Penguji I  Asmaul Husna N, M.Kom | ………………………….. |
| 2. | Penguji II  Sudirman S. Panna, M.Kom | ………………………….. |
| 3. | Penguji III  Serwin, M.Kom | ………………………….. |
| 4. | Pembimbing I  Irvan A. Salihi, M.Kom | ………………………….. |
| 5. | Pembimbing II  Apriyanto Alhamad, 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 arahan dari Tim Pembimbing.
3. Dalam karya tulis ( Skripsi ) saya ini tidak terdapat karya atau pendapat yang telah dipublikasikan orang lain, kecuali secara tertulis dicantumkan sebagai acuan/sitasi dalam naskah dan dicantumkan pula dalam daftar pustaka
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaraan 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, 30 November 2019

Yang Membuat Pernyataan,

Amin Dwi Satrio

# ABSTRAct

*The introduction or detection of banana ripeness in this study was divided into two groups namely Immature and Mature. This study aims to classify the level of maturity based on the image of a banana by using the feature extraction method that is the gray level co-occurrence matrix. This research method consists of: conversion of RGB data to grayscale, image normalization, detection of maturity, feature extraction and classification. In this study, training data of 100 banana images are used, which are public data taken from a TIER-BASED DATASET dataset provider : MUSA - ACUMINATA BANANA FRUIT SPECIES. The data consists of the level of maturity above and each maturity of bananas consists of Matang 50, and Immature 50. The image data is processed into grayscale images which are then detected at the level of maturity of bananas. After the maturity of the bananas is obtained, then detection is carried out on the part of maturity found. The next step is to calculate the characteristics using the gray level co-occurrence matrix. The algorithm used for the classification process is the k-nearest neighbor algorithm. The final test results show that the proposed method has been able to detect maturity with the accuracy of the results calculated using a confusion matrix of 76%. Thus the application of the gray level co-occurrence matrix and k-nearest neighbor methods on the problem of detecting the level of maturity needs to be developed accurately.*

*Keywords: Introduction of Banana Maturity, Gray Level Co-Occurrence Matrix, K-Nearest Neighbor*

# ABSTRAK

Pengenalan atau Deteksi kematangan buah pisang dalam penelitian ini dibagi menjadi dua kelompok yaitu Belum Matang dan Matang. Penelitian ini bertujuan untuk mengklasifikasi tingkat kematangan berdasarkan citra buah pisang dengan menggunakan metode fitur ekstraksi yaitu *gray level co-occurrence matrix*. Metode penelitian ini terdiri dari : konversi data *rgb* ke *grayscale*, normalisasi citra, deteksi kematangan, ekstraksi fitur dan klasifikasi. Dalam penelitian ini digunakan data latih sebanyak 100 citra pisang yang merupakan data *public* yang diambil dari sebuah penyedia dataset dengan nama Tier-Based Dataset : Musa – Acuminata Banana Fruit Species. Data tersebut terdiri dari tingkat kematangan di atas dan masing-masing kematangan buah pisang terdiri dari Matang 50, dan Belum Matang 50. Data citra tersebut diolah menjadi citra *grayscale* yang kemudian dilakukan deteksi tingkat kematangan buah pisang. Setelah didapat kematangan buah pisang kemudian dilakukan *pedeteksian* pada bagian kematangan yang ditemukan. Yang selanjutnya 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 mendeteksi kematangan dengan hasil akurasi yang digitung menggunakan *confusion matrix* sebesar 76 %. Dengan demikian penerapan metode *gray level co-occurrence matrix* dan *k-nearest neighbor* pada masalah deteksi tingkat kematangan perlu dikembangkan lagi hasil akurat.

Kata kunci : Pengenalan kematangan buah pisang*, Gray Level Co-Occurrence Matrix, K-Nearest Neighbor*

# KATA PENGANTAR



Alhamdulillah, penulis dapat menyelesaikan usulan penelitian ini dengan judul: **“PENERAPAN Metode *Gray Level Co-Occurrence Matrix* (GLCM) Dan *K-Nearest Neighbor* (K-NN) UNTUK PENGENALAN KEMATANGAN BUAH PISANG BERDASARKAN TEKSTUR WARNA KULIT”**, untuk memenuhi salah satu syarat penyusunan Skripsi Program Studi Teknik Informatika Fakutas Ilmu Komputer Universitas Ichsan Gorontalo.

Penulis menyadari sepenuhnya bahwa usulan pengujian 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 Muh. Ichsan Gaffar, SE., M.Ak, selaku Ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo;
2. Bapak Dr. Abdul Gaffar La Tjokke, 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 Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
5. Ibu Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
6. Bapak Sudirman Malangi, M.Kom, selaku Wakil Dekan III Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
7. Bapak Irvan Abraham Salihi, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo Sekaligus Pembimbing Utama;
8. Bapak Apriyanto Alhamad, M.Kom, selaku Pembimbing Pendamping;
9. Bapak dan Ibu Dosen Universitas Ichsan Gorontalo yang telah mendidik dan mengajarkan berbagai disiplin ilmu kepada kami;
10. Kedua Orang Tua Saya yang tercinta, atas segala kasih sayang, jerih payah dan doa restunya dalam membesarkan dan mendidik penulis;
11. Rekan-rekan seperjuangan yang telah banyak memberikan bantuan dan dukungan moril yang sangat besar kepada penulis;
12. Kepada semua pihak yang ikut membantu dalam penyelesaian skripsi ini yang tak sempat penulis sebutkan satu-persatu.

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

|  |  |
| --- | --- |
|  | Gorontalo, 30 November 2019  Penulis |

# DAFTAR ISI

[**PERSETUJUAN SKRIPSI** ii](#_Toc26143034)

[**PENGESAHAN SKRIPSI** iii](#_Toc26143036)

[**PERNYATAAN SKRIPSI** iv](#_Toc26143038)

[**ABSTRACT** v](#_Toc26143039)

[**ABSTRAK** vi](#_Toc26143040)

[**KATA PENGANTAR** vii](#_Toc26143041)

[**DAFTAR ISI** ix](#_Toc26143042)

[**DAFTAR GAMBAR** xi](#_Toc26143043)

[**DAFTAR TABEL** xii](#_Toc26143044)

[**DAFTAR LAMPIRAN** xiii](#_Toc26143045)

[**BAB I PENDAHULUAN** xiii](#_Toc26143046)

[1. 1 Latar Belakang 1](#_Toc26143047)

[1. 2 Identifikasi Masalah 4](#_Toc26143048)

[1. 3 Rumusan Masalah 4](#_Toc26143049)

[1. 4 Tujuan Penelitian 4](#_Toc26143050)

[1. 5 Manfaat Penelitian 5](#_Toc26143051)

[1. Manfaat Teoritis 5](#_Toc26143052)

[2. Manfaat Praktis 5](#_Toc26143053)

[**BAB II LANDASAN TEORI** 6](#_Toc26143054)

[2. 1 Tinjauan Studi 6](#_Toc26143055)

[2. 2 Tinjauan Pustaka 8](#_Toc26143057)

[2. 2. 1 Pisang *(Musa Paradisiaca)* 8](#_Toc26143058)

[2. 2. 2 Computer Vision 8](#_Toc26143059)

[2. 2. 3 Pengolahan Citra (Image Processing) 9](#_Toc26143060)

[2. 2. 5 Pra pengolahan Data citra 17](#_Toc26143064)

[2. 2. 6 Gray Scalling 18](#_Toc26143065)

[2. 2. 7 GLCM ( Gray Level Co-Occurrence Matrix ) 19](#_Toc26143066)

[2. 2. 8 K–NN ( K–Nearest Neighbor) 22](#_Toc26143071)

[2. 2. 9 Evaluasi Model 26](#_Toc26143077)

[2. 2. 10. UML (*Unified Modeling Language*) 27](#_Toc26143079)

[2. 2. 11. Python 29](#_Toc26143081)

[2. 3 Kerangka Pikir 30](#_Toc26143082)

[**BAB III METODE PENELITIAN** 31](#_Toc26143083)

[3. 1 Jenis, Metode, Subjek, Objek, Waktu dan Lokasi Penelitian 31](#_Toc26143084)

[3. 2 Pengumpulan Data 31](#_Toc26143085)

[3. 3 Pemodelan 32](#_Toc26143086)

[3. 3. 1 Pra Pengolahan 32](#_Toc26143087)

[3. 3. 2 Ekstraksi Ciri 33](#_Toc26143088)

[3. 3. 3 Data Traning 33](#_Toc26143089)

[3. 3. 4 Traning Menggunakan K-NN 33](#_Toc26143090)

[3. 3. 5 Model 33](#_Toc26143091)

[3. 3. 6 Data Testing 33](#_Toc26143092)

[3. 3. 7 Hasil Klasifikasi 34](#_Toc26143093)

[3. 3. 8 Evaluasi 34](#_Toc26143094)

[**BAB IV HASIL PENELITIAN** 35](#_Toc26143095)

[4. 1. Hasil Pengumpulan Data 35](#_Toc26143096)

[4. 2. Hasil Pemodelan 36](#_Toc26143098)

[4. 2. 1. Pra Pengolahan 36](#_Toc26143099)

[4. 2. 2. Ekstraksi Fitur 38](#_Toc26143102)

[4. 2. 3. Cross Validation 41](#_Toc26143106)

[4. 2. 7. Klasifikasi 42](#_Toc26143107)

[4. 4. Evaluasi 44](#_Toc26143120)

[4. 3. Confusion Matrix 44](#_Toc26143121)

[**BAB V PEMBAHASAN** 46](#_Toc26143124)

[5. 1. Pembahasan Model 46](#_Toc26143125)

[5. 2. Pembahasan Sistem 49](#_Toc26143128)

[**BAB VI PENUTUP** 52](#_Toc26143134)

[6. 1. Kesimpulan 52](#_Toc26143135)

[6. 2. Saran 52](#_Toc26143136)

[DAFTAR PUSTAKA 53](#_Toc26143137)

# DAFTAR GAMBAR

[**Gambar 2. 1**: Ilustrasi Citra 12](#_Toc29801785)

[**Gambar 2. 2**: Citra Grayscale 13](#_Toc29801786)

[**Gambar 2. 3**:Hubungan antara ; (a) citra asli; (b) matrix glcm 19](#_Toc29801791)

[**Gambar 2. 4**: Ilustrasi penentuan arah 20](#_Toc29801792)

[**Gambar 2. 5**:Menentukan pixel tetangga 20](#_Toc29801793)

[**Gambar 2. 6**:Proses membentuk matriks glcm 21](#_Toc29801794)

[**Gambar 2. 7**:k-nn dengan k tetangga, (a) 1-nn, (b) 3-nn, (c) 5-nn, (d) 7-nn 24](#_Toc29801796)

[**Gambar 4. 1**: Nilai dari suatu citra 38](#_Toc29801828)

[**Gambar 5. 1**: Tampilan Awal Aplikasi 49](#_Toc29801846)

[**Gambar 5. 2**: Tampilan Proses Load Data Training 50](#_Toc29801847)

[**Gambar 5. 3**: Tampilan Proses Load Data Testing 50](#_Toc29801848)

[**Gambar 5. 4**: Tampilan Nama Folder Yang Telah Dipilih 51](#_Toc29801849)

[**Gambar 5. 5**: Tampilan Hasil Pengenalan 51](#_Toc29801850)

# DAFTAR TABEL

[**Tabel 2. 1:** Penelitian Terkait 6](#_Toc26144244)

[**Tabel 2. 2:** Klasifikasi Data Traning 24](#_Toc26144261)

[**Tabel 2. 3:** Perhitungan Kuadrat Jarak Data *training* dengan Data *testing* 25](#_Toc26144262)

[**Tabel 2. 4:** Penentuan 3 Tetangga Terdekat dari Data testing 25](#_Toc26144263)

[**Tabel 2. 5:** Klasifikasi Kelas Tetangga Terdekat 25](#_Toc26144264)

[**Tabel 2. 6:** Model *confusion matrix* 26](#_Toc26144266)

[**Tabel 2. 7:** Tipe Diagram UML (*Unified Modeling Language* ) 28](#_Toc26144268)

[**Tabel 4. 1:** Data Gambar Kematangan Pisang 35](#_Toc26144285)

[**Tabel 4. 2:** Proses perubahan citra asli menjadi *grayscale* 36](#_Toc26144288)

[**Tabel 4. 3:** Proses normalisasi citra 37](#_Toc26144289)

[**Tabel 4. 4:** Proses normalisasi citra 37](#_Toc26144289)

[**Tabel 4. 5:** Matrix GLCM 39](#_Toc26144292)

[**Tabel 4. 6:** Matrix Ternormalisasi 39](#_Toc26144293)

[**Tabel 4. 7:** Hasil Perhitungan GLCM 42](#_Toc26144296)

[**Tabel 4. 8:** Nilai Perbandingan 42](#_Toc26144296)

[**Tabel 4. 9:** Perhitungan Euclidean 43](#_Toc26144297)

[**Tabel 4. 10:** Tetangga Terdekat 43](#_Toc26144298)

[**Tabel 4. 11:** Hasil Data Testing 44](#_Toc26144310)

[**Tabel 4. 12:** Hasil Data Uji 45](#_Toc26144311)

[**Tabel 5. 1:** Hasil Uji Coba Nilai K = 2 47](#_Toc26144314)

[**Tabel 5. 2:** Hasil Uji Coba Nilai K = 3 48](#_Toc26144315)

# DAFTAR LAMPIRAN

[*Lampiran 1 Coding* 56](#_Toc29798668)

[*Lampiran 2 Riwayat Hidup Mahasiswa* 58](#_Toc29798669)

[*Lampiran 3 Surat Rekomendasi Penelitian* 59](#_Toc29798670)

[*Lampiran 4 Surat Rekomendasi Bebas Plagiasi* 60](#_Toc29798671)

[*Lampiran 5 Hasil Cek Plagiasi Turnitin* 60](#_Toc29798672)

# BAB I PENDAHULUAN

## Latar Belakang

Pada perkembangan teknologi ini membuat manusia ingin meningkatkan efektifitas dan efisiensi dalam berbagai bidang, salah satunya dalam bidang pertanian. Dalam dunia pertanian, kemajuan teknologi sangat dibutuhkan untuk menunjang kegiatan–kegiatan yang ada pada bidang pertanian, salah satunya dalam pengolahan hasil pertanian dan perkebunan. Industri pengolahan hasil pertanian dan perkebunan kini semakin berkembang pesat seiring perkembangan teknologi yang ada khususnya untuk produksi buah pisang. Buah pisang sendiri merupakan salah satu komoditas hortikultura dari kelompok buah - buahan yang saat ini cukup diperhitungkan adalah tanaman pisang[1]. Hal ini dapat dilihat dari produksi pisang di Indonesia terus meningkat. Menurut Badan Pusat Statistik dan Direktorat Jenderal Hortikultura, produksi pisang 3 tahun terakhir ini terus mengalami perkembangan. Pada tahun 2013 produksi pisang sebanyak 6.279.279 ton. Dan pada tahun 2014 dan 2015 produksi pisang di Indonesia mengalami peningkatan yang sangat pesat sekali yakni berturut-turut 6.862.558 ton dan 7.299.266 ton[2].

Banyaknya produksi pisang di Indonesia, tapi tidak seimbang dengan pengolahan terhadap buah pisang tersebut. Penelitian ini akan mencoba membantu pihak pengolahan pisang yang masih sedikit, namun powerfull dalam pengolahan pisang. Pada proses pengolahan buah pisang tersebut, ada beberapa tahapan sebelum produk yang dihasilkan di distribusikan ke konsumen. Salah satu tahapannya adalah pemilihan pisang hasil pertanian dan perkebunan sesuai dengan kebutuhan untuk diolah. Pemilihan jenis buah pisang sangat perlu dilakukan, sebab berbeda jenis pisang berbeda pula cara pengolahan dan hasil olahanya. Sedangkan untuk kondisi buah pisang ditentukan oleh beberapa parameter, diantaranya adalah parameter tingkat kematangan yang dilihat dari sisi warna dari pisang. Mutu pisang yang baik sangat ditentukan oleh tingkat ketuaan buah dan penampakannya. Tingkat ketuaan buah diukur berdasarkan umurnya, sedang penampakan yang baik diperoleh dari penanganan pasca panen yang baik [3]. Umumnya klasifikasi kematangan buah pisang dilakukan dengan cara manual yaitu menggunakan indera manusia.

Kemajuan teknologi komputer termasuk interaksi antara manusia dengan computer pada saat ini sudah berkembang sangat pesat. Bahkan dibidang pertanian penggunaan teknologi computer sudah banyak digunakan. Penerapannya digunakan untuk deteksi penyakit pada tanaman, klasifikasi mutu, penentuan berat dan identifikasi jenis buah-buahan maupun sayuran[4]. Teknologi pengolahan citra memungkinkan untuk memilah dan mengamati produk pertanian dan perkebunan tersebut secara otomatis dengan bantuan aplikasi pengolah citra.

Sebagai upaya untuk mengatasi keterbatasan kemampuan petani dalam melakukan identifikasi kematangan buah pisang maka perlu dibuat suatu pengolahan citra. Pengolahan citra diharapkan dapat membantu membedakan jenis buah pisang dari gambar yang dimasukkan ke dalam pengolahan citra. Dalam penelitian kali ini metode pengolahan citra yang digunakan adalah Metode preprocessing GLCM (Gray Level Co-occurrent Matrix) termasuk dalam metode statistik yang menggunakan distribusi derajat keabuan dengan mengukur tingkat kekontrasan, granularitas(kedetailan), dan kekasaran suatu daerah dari hubungan ketetanggaan antar piksel di dalam citra. Dengan melakukan perhitungan tertentu pada matrik co-occurrence maka akan dapat diketahui nilai derajat keabuan dari citra yang bersebelahan dengan citra yang diambil. Dan metode Algoritma K-NN (K-Nearest Neighbour) sebagai metode data mining yang digunakan untuk mendapatkan informasi dan mengambil nila dari yang terdekat, dengan tujuan penelitian dapat memudahkan dalam membedakan jenis buah pisang[5].

Metode *K-Nearest Neighbor* (K-NN) memang sudah sangat sering digunakan sebagai metode pengujian yang berhubungan dengan buah, hal ini seperti percobaan yang dilakukan sebelumnya, diantaranya : “*Identifikasi tumbuhan obat herbal berdasarkan citra daun menggunakan metode Gray Level Co-Occurrence Matrix (GLCM) dan untuk klasifikasinya menggunakan metode K-Nearest Neighbor (K-NN)”* yang dilakukan oleh F.S sutojo dan D.R.I mendapatkan nilai akurasi 83.33%[6], sedangkan pada “*Identifikasi jenis buah apel menggunakan Algoritma K-NN dengan ekstrasi fitur histogram*” memberikan tingkat akurasi 90%[7]. Begitu pula pada *“Aplikasi pemilihan buah menggunakan metode LVQ yang dikombinasi dengan metode K-NN”* menghasilkan akurasi sebesar 95%[8].*“Metode Klasifikasi Mutu Jambu Biji Menggunakan KNN Berdasarkan Fitur Warna Dan Tekstur”* yang dilakukan oleh Taftyani Yusuf dan Agus Harjoko menghasilkan akurasi 91.25%[9].

Dari penelitian yang dilakukan sebelumnya dapat dikatakan bahwa metode *K-Nearest Neighbor* (K-NN) baik digunakan untuk klasifikasi pada buah pisang. Dari sebuah pisang ini akan dilakukan pengklasifikasian terhadap jenis pisang berdasarkan bentuk, ukuran, dan warna dari sebuah pisang ini.

Pada penelitian ini menggunakan ekstraksi ciri berupa metode *Gray Level Co-Occurrence Matrix* (GLCM) berdasarkan beberapa penelitian yang sudah pernah dilakukan, diantaranya : *“Klasifikasi jenis kualitas keju dengan menggunakan metode Gray Level Co-Occurrence Matrix (GLCM) dan klasifikasinya menggunakan metode Support Vector Machine (SVM)”* yang dilakukan oleh Dr. Ir. Bambang Hidayat, DEA. Dengan akurasi sebesar 97.92% dengan waktu komputasi 0.0286s dengan menggunakan metode GLCM[10]. Selanjutnya pada penelitian beliau lainnya *“Deteksi batik bojonegoro menggunakan Gray Level Co-Occurrence Matrix (GLCM) dan untuk klasifikasinya menggunakan metode Naive Bayes”* dengan akurasi terbaik sebesar 85% dengan waktu komputasi 206.6715 detik[11]. Kemudian penelitian *“Simulasi dan analisis pengenalan citra daging sapi maupun babi menggunakan metode ekstraksi ciri tekstur Grey Level Coocurent Matrix (GLCM) dan metode klasifikasi dengan Fuzzy Logic.”* Pada penelitian inididapat akurasi yang baik dengan jarak pengambilan citra dengan kamera 8cm, *epoch* 100, derajatkeabuan GLCM. dengan akurasi 98.5%[12].

Pada penelitian ini untuk meningkatkan akurasi pengenalan metode *Gray Level Co-Occurrence Matrix* (GLCM) dapat dikombinasikan dengan metode *K-Nearest Neighbor* dalam proses klasifikasi. K-NN merupakan salah satu teknik klasifikasi yang paling dasar dan sederhana. Metode ini memiliki kelebihan yakni pelatihan yang sangat cepat, sederhana dan mudah di pelajari, tahan terhadap data yang memiliki derau dan efektif jika data pelatihan besar[13]. Hasil ini menunjukan bahwa penggunaan metode K-NN baik dalam meningkatkan akurasi klasifikasi jika dikombinasikan dengan metode lain.

Dengan demikian metode K-NN dapat dikombinasikan dengan metode GLCM pada proses klasifikasi sehingga dapat meningkatkan akurasi pengenalan (Identifikasi) kematangan buah pisang. Sistem di bangun menggunakan *Spyder Phyton* sebagai editor untuk mengimplementasikan script, dan open CV untuk pemrosesan citranya.

Berdasarkan berbagai pemaparan di atas, telah membawa peneliti untuk melakukan penelitian dengan judul: “**Penerapan Metode *Gray Level Co-occurrence Matrix* (GLCM) Dan *K-Nearest Neighbor* (K-NN) Untuk Pengenalan Kematangan Buah Pisang Berdasarkan Tekstur Warna Kulit**”. Diharapkan dari penelitian ini dapat memberi kontribusi, berupa aplikasi yang dapat membantu masyarakat dalam mengenali kematangan buah pisang.

## Identifikasi Masalah

1. Belum adanya sistem yang mampu mengenali keseluruhan kematangan buah pisang dari warna kulitnya dengan hanya melihatnya saja tanpa bantuan metode klasifikasi sebagai alat bantu.
2. Masih minimnya penerapan teknologi computer vision terhadap buah pisang dan perlunya penambahan algoritma klasifikasi untuk meningkatkan akurasi algoritma GLCM pada pengenalan kematangan buah pisang.

## Rumusan Masalah

Bagaimana peningkatan uji coba akurasi metode GLCM apabila diterapkan metode K-NN dalam proses identifikasi pengenalan kematangan buah pisang ?

## Tujuan Penelitian

Untuk mengetahui hasil akurasi metode GLCM dan KNN dalam proses klasifikasi pengenalan kematangan buah pisang.

## Manfaat Penelitian

### Manfaat Teoritis

Memberikan masukan bagi perkembangan ilmu pengetahuan dan teknologi, khususnya pada bidang ilmu komputer, yaitu berupa penerapan metode K-NN untuk mengoptimalkan kinerja metode GLCM pada sistem pengenalan buah pisang.

### Manfaat Praktis

Sebagai solusi untuk mempermudah masyarakat dalam mengenali kematangan buah pisang.

# BAB II LANDASAN TEORI

## Tinjauan Studi

Penelitian tentang pengenalan tingkat kematangan buah pisang berdasarkan kulitnya dan metode yang digunakan telah dilakukan oleh beberapa peneliti sebelumnya. Berikut beberapa penelitian terkait:

**Tabel 2. 1:** Penelitian Terkait

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | PENELITI | JUDUL/TAHUN | METODE | HASIL |
| 1. | Taftyani Yusuf dan Agus Harjoko | Metode Klasifikasi Mutu Jambu Biji Menggunakan KNN Berdasarkan Fitur Warna Dan Tekstur/2017 | *K-Nearest Neighbor* (KNN) | Hasil dari penelitian mereka menghasilkan akurasi 91.25% |
| 2. | Ion Ataka Halela, Bowo Nurhadiyono,S.Si, M.Kom, dan Farah Zakiyah Rahmanti. | Identifikasi Jenis Buah Apel Menggunakan Algoritma *K-Nearest Neighbor* (KNN) dengan Ekstraksi Fitur *Histogram/*2016 | *K-Nearest Neighbor, Histogram* | Hasil dari penelitian disimpulkan identifikasi pada citra apel menggunakan algoritma K-*Nearest Neighbor* dengan menggunakan 100 data citra apel dengan terdiri dari 50 jenis apel envy dan 50 jenis apel pasific rose. Data uji sebanyak 10 citra yang terdiri dari 5 jenis citra apel envy dan 5 jenis citra apel pasific rose memperoleh akurasi sebesar 90%. |
| NO | PENELITI | JUDUL/TAHUN | METODE | HASIL |
| 3. | Fittria Shofrotun Ni’mah, T. Sutojo, dan De Rosal Ignatius Moses Setiadi | Identifikasi Tumbuhan Obat Herbal Berdasarkan Citra Daun Menggunakan  Algoritma *Gray Level Co-occurence Matrix* dan *K-Nearest Neighbor/*2018 | *Gray Level Co-Occurrence Matrix* (GLCM) dan *K-Nearest Neighbor* | Penelitian ini telah berhasil mengimplementasikan  algoritma GLCM dan KNN  Algoritma tersebut dapat digunakan untuk  mengidentifikasi tumbuhan obat herbal berdasarkan  citra daun dengan akurasi rata-rata sebesar 83,33%. |
| 4. | Feri Wibowo, Dimara Kusuma Hakim, dan Sigit Sugiyanto. | Pendugaan Kelas Mutu Buah Pepaya  Berdasarkan Ciri Tekstur GLCM Menggunakan Algoritma  *K-Nearest Neighbor/*2018 | *Gray Level Co-Occurrence Matrix* (GLCM) dan *K-Nearest Neighbor* (K-NN) | Berdasarkan hasil penelitian pendugaan kelas mutu buah pepaya (*carica* *papay L*) dapat disimpulan proses pendugaan kelas mutu buah pepaya *Calina Ipb-9* dilakukan menggunakan program yang telah berhasil dirancang dan dibuat menggunakan teknologi pengolahan citra digital. Diperoleh tujuh fitur tekstur GLCM yaitu *energi, entropy,* *contras, homogenity, idm, variance* dan *dissimilarity.* Hasil pengujian pada Tabel 2 menunjukan  nilai akurasi terbaik adalah jumlah *k*  tentangga 9 dengan nilai akurasi sebesar  88,88% |

## Tinjauan Pustaka

### Pisang *(Musa Paradisiaca)*

Pisang *(Musa Paradisiaca)* merupakan salah satu komoditas pertanian yang sangat digemari masyarakat, dan menjadi salah satu komoditas tanaman buah yang mulai dikebunkan selain mangga, durian, rambutan, manggis, jeruk, nenas dan pepaya. Tanaman pisang (*Musa* *spp*) telah diproklamirkan sejak sebelum masehi (SM). Nama *Musa* diambil dari nama seorang dokter bernama Antonius Musa pada zaman Kaisar Romawi Octavianus Augustus (63 SM – 14 M), beliau selalu menganjurkan pada kaisarnya untuk makan pisang setiap harinya agar tetap kuat, sehat, dan segar (Mudjajanto, ES & Lilik K.2008).

Tanaman pisang dapat dikatakan sebagai tanaman serbaguna, mulai dari akar, batang (bonggol), batang semu (pelepah), daun, bunga, buah sampai kulitnyapun dapat dimanfaatkan untuk berbagai keperluan. Buah pisang kaya akan sumber vitamin dan karbohidrat serta sangat digemari orang karena enak dimakan baik sebagai buah meja atau melalui pengolahan terlebih dahulu. Di Indonesia, pisang masih biasa ditanam oleh masyarakat sebagai tanaman pekarangan ataupun perkebunan dalam skala kecil, pemeliharaan serta pemanfaatannyapun kurang maksimal. Untuk itu perlu ada suatu pendekatan khusus, agar tanaman pisang dikenal manfaatnya secara luas oleh masyarakat (Prihatini dkk,1999: 94)[14].

### Computer Vision

*Computer Vision* adalah sebuah kemampuan sebuah komputer yang ke desain agar mampu melihat sebuah object sehingga mampu menampilkan objek digital dan bisa mengoleksi data secara visual komputer bisa melakukan beberapa pekerjaan yang tidak bisa dilakukan oleh manusia:

1. Komputer mampu melihat data dalam bentuk *pixel* bahkan dalam warna yang berbeda.
2. Komputer mampu membandingkan dua object gambar yang sama persis.
3. Komputer mampu melihat sebuah object data selama berjam-jam bahkan berhari – hari. Vision itu sendiri adalah suatu proses evaluasi sebuah data yang bersumber dari image umumnya camera, dengan teknik ektrasi menggunakan algoritma tertentu.

Dalam istilah sederhana, *computer vision* adalah bagaimana komputer/mesin dapat melihat, teknik *computer vision* mampu memvisualisasikan data menganalisa berupa gambar image atau dalam bentuk video. Tujuan utama dari *computer vision* adalah agar komputer atau mesin dapat meniru kemampuan perseptual mata manusia dan otak, atau bahkan dapat mengunggulinya untuk tujuan tertentu[15].

### Pengolahan Citra (*Image Processing*)

*Image processing* adalah suatu bentuk pengolahan atau pemrosesan sinyal dengan input berupa gambar (*image*) dan ditransformasikan menjadi gambar lain sebagai keluarannya dengan teknik tertentu. *Image processing* dilakukan untuk memperbaiki kesalahan data sinyal gambar yang terjadi akibat transmisi dan selama akuisisi sinyal, serta untuk meningkatkan kualitas penampakan gambar agar lebih mudah diinterpretasi oleh sistem penglihatan manusia baik dengan melakukan manipulasi dan juga penganalisisan terhadap gambar.(Fitriandy, 2013)[16].

Pengolahan citra merupakan proses pengolahan sinyalyang inputnya adalah citra. Outputnya dapat berupacitra atau sekumpulan karakteristik atau parameter yang berhubungan dengan citra.Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh bit-bit tertentu.(Nazaruddin, 2013)[17].

Menurut (Nazaruddin, 2013) terdapat operasi pengolahan citra sebagai berikut diantaranya

1. Pembesaran atau pengecilan ukuran citra
2. Rotasi citra
3. Penajaman citra
4. Penonjolan fitur tertentu dari citra
5. Kompresi citra
6. Koreksi citra yang kabur atau tidak fokus
7. Segmentasi citra
8. Pengurangan *noise*
9. Pengenalan objek

Selain itu menurut (Nazaruddin, 2013) juga pengolahan citra memiliki beberapa fungsi, diantaranyaadalah:

1. Digunakan sebagai proses memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau komputer
2. Digunakan untuk Teknik pengolahan citra dengan mentrasformasikan citra menjadi citra lain contoh : pemampatan citra (*image compression*)
3. Sebagai proses awal (*preprocessing*) dari komputer visi.

Menurut (Hermawan, 2013) terdapat pengertian – pengertian dasar *Image processing*[18].

1. *Pixel* ( *picture elemen* )

Adalah titik terkecil (elemen) yang membentuk suatu gambar pada layer monitor, disebut juga dengan dot.

1. Resolusi

Adalah Bamyaknya titik ( *pixel* ) yang menyusun suatu gambar atau layer. Ukurannya adalah jumlah *pixel* horizontal dikalikan jumlah *pixel* vertical. Contoh : ukuran 640 x 480 , artinya 640 *pixel* mendatar, dan 480 vertikal.

1. Dots per inchi (dpi / *pixel* per inchi )

Banyaknya titik atau *pixel* tiap satuan inchi. Contoh : 600 dpi , berarti terdapat 600×600 *pixel* tiap inchi persegi.

1. Warna

Beberapa model warna :

* 1. RGB (*red, green, blue*)

Merupakan kombinasi dari elemen warna *red* (merah), *green*(hijau), dan *blue* (biru). Tiap titik atau *pixel* merupakan kombinasidari ketiga elemen warna tsb. Setiap elemen memiliki 0 – 255tingkat warna . Contoh : warna hitam merupakan kombinasi dari R= 0, G=0, B=0; warna putih merupakan kombinasi dari R = 255, G= 255, B= 255; warna kuning dihasilkan dari kombinasi R= 255, G=255, B= 0. Sehingga kombinasi warna yang dapat dibentuk dari mode rgb adalah 255 x 256 x 256= 16777216 (16 juta warna).

b. CMYK (*cyan , magenta, yellow , black*)

Merupakan sistem presentasi warna untuk proses cetak 4 warna( *cyan , magenta, yellow , dan black* ) . Dengan sistem ini gambarakan di simpan dalam 4 channel

1. *Color Depth*

*Color Depth* adalah besarnya informasi data dari 1 satuan sample(dot/*pixel*). Setiap bitmap/ dot/ *pixel*, bias berupa hitam , putih, abu – abu atau warna. *Color Depth* dinyatakan dalam angka yang digunakan untuk menyatakan beberapa variasi warna yang mampu ditampilkan oleh suatu bitmap /dot/*pixel*, yang dinyatakan secara kuantitatif. Rumusnya adalah 2 n (dua pangkat n , dimana n adalah besarnya *bit depth*).

1. Menghitung ukuran file raster image

Ukuran dalam byte = *width* x *height* x *colordepth*

Keterangan :

a. *width* : lebar dari images, diukur dengan *pixel*

b. *Height* : tinggi dari images, diukur dengan *pixel*

c. *Color depth* : jumlah bit warna yang diukur dalam bit / *pixel*

1 byte = 8 bit

1 KB (kilobyte) = 1024 byte

1 MB (megabyte) = 1024 KB

1. Bitmap file format
   1. *Microsoft bitmap* (. Bmp)

Digunakan di Microsoft windows

1. TIFF – *Tagged Image File Format* (.tif)

Digunakan untuk faxing images (biasanya)

1. JPEG – *Joint Photographic Expert Group* (.jpg)

Berguna untuk menyimpan *photographic images*

1. GIF – *Graphics Interchange Format* (.gif)

Banyak digunakan di web sites

1. PNG – *Portable Network Graphic* (.png)

Format baru untuk web *graphics*

1. PCD – Kodak photo CD

Format baru untuk menyimpan image dalm bentuk terkompresi dalam CD



**Gambar 2. 1:** Ilustrasi Citra

(Sumber: https://pixabay.com/id/photos/pisang-buah-kuning)

Sebuah citra digital dapat di anggap sebagai sebuah larik besar yang berisi sample *point* dari sebuah gambar kontinyu yang masing – masing *point* mempunyai kualitas kecerahan. Pengertian tersebut lebih di kenal dengan istilah *pixsel* yang sering digunakan pada citra digital.

*Pixsel* mempuyai dua parameter, yaitu koordinat, dan intentitas atau warna, nilai yang terdapat pada koordinat *(x,y)* adalah *f(x,y),* yaitu besar intensitas atau warna dari titik piksel itu. Oleh sebab itu, sebuah citra digital dapat ditulis dalam bentuk matriks berikut.

Berdasarkan gambaran tersebut, secara matematis citra digital dapat dituliskan sebagai fungsi *f(x,y),* dimana harga x (baris) dan y (kolom) merupakan koordinat posisi dan *f(x,y)* adalah nilai fungsi pada setiap titik *(x,y)* yang menyatakan besar intensitas citra atau tingkat keabuan atau warna dari *pixel* di titik tersebut (sutoyo, et al, 2009).

Suatu *pixsel* memiliki nilai dalam rentang tertentu, dari nilai minimum hingga maksimum. Jangkauan yang digunakan berbeda - beda tergantung dari jenis warnanya. Namun secara umum jangkauannya adalah 0 (nol) sampai dengan 255. Citra dengan penggambaran seperti ini di golongkan kedalam citra integer. Citra digital secara umum dikelompokan menjadi 3 macam, yaitu :

* Citra Biner

Dimana masing – masing *pixel* hanya berwarna hitam dan putih, maka hanya di perlukan satu bit per – *pixel*. Citra yang di representasikan sebagai citra biner sangat cocok digunakan untuk citra biner sangat cocok digunakan untuk citra yang menggambarkan *text, finger print*, atau rencana *arsitektural.*

* Citra abu – abu *( Grayscale ).*

Dimana masing – masing pixel berisikan warna abu – abu dengan nilai normal antara 0 (hitam) sampai 255 (putih). Range tersebut berarti masing – masing pixel dapat direpresentasikan oleh nilai 8 bit atau satu byte. Citra *Grayscale* dengan range nilai yang lain juga digunakan. Akan tetapi pada umumnya citra tersebut memiliki *range* pangkat dua dari 8 bit.



**Gambar 2. 2:** Citra Grayscale

* Citra RGB atau *True Colour*.

Untuk citra RGB masing – masing piksel mempunyai sebuah warna khusus. Warna dideskripsikan oleh kombinasi warna merah *(Red),* hijau *(Green)*, dan biru *(Blue)*. Jika masing - masing komponen mempunyai range antara 0-255. Maka total range yang digunakan untuk citra RGB adalah 155 pangkat 3 atau 16.777.216 kemungkinan warna. Range tersebut merupakan range yang mencukupi untuk membuat sebuah citra yang penuh dengan warna. Oleh Karena jumlah bit yang di butuhkan oleh masing – masing *pixsel* adalah 24 maka citra RGB biasa di sebut dengan 24-bit color image. Sebagai sebuah citra yang terdiri dari tumpukan *(stuck)* 3 matriks yang mewakili nilai merah, hijau, dan biru. Dengan kata lain untuk setiap piksel memiliki 3 buah nilai.

Sedangkan pada citra digital terdapat beberapa elemen dasar pembentuk citra digital, diantaranya :

1. Kecerahan *(brightness)*

Kecerahan intesitas cahaya rata – rata dari suatu area yang melingkupinya.

1. Kontras *(contrast)*

Sebaran terang *(lightness)* dan sebaran gelap *(darkness)* di dalam suatu citra. Citra dengan kontras rendah komposisi citranya sebagian besar terang atau sebagian besar gelap. Citra dengan kontras yang baik , komposisi gelap dan terangnya tersebar merata.

1. Kontur *(Contour)*

Merupakan keadaan yang ditimbulkan oleh perubahan intensitas pada piksel – piksel tetangga, sehingga kita dapat mendeteksi tepi objek di dalam citra.

1. Warna *(Color)*

Adalah persepsi yang dirasakan sistem visual manusia terhadap panjang gelombang cahaya yang di pantulkan objek. Warna – warna yang dapat ditangkap oleh mata manusia merupakan kombinasi cahaya dengan panjang berbeda. Kombinasi yang memberikan rentang warna paling lebar adalah *Red* (R), *Green* (G), *Blue* (B).

1. Bentuk *(shape)*

Adalah properti intrinsic dari objek 3 (tiga) dimensi, dengan pengertian bahwa bentuk merupakan property intrinsik utama untuk visual manusia. Umumnya citra yang di bentuk oleh manusia merupakan 2D, sedangkan objek yang dilihat adalah 3D.

1. Tekstur *(texture)*

Walaupun ada definisi secara formal tentang tekstur, secara intuitif tekstur menyatakan ciri dari permukaan objek yang menggambarkan pola visual. Ciri ini berisi informasi tentang komposisi struktur permukaan, seperi misalnya awan, daun, batu bata, dan kain. Selain itu juga menjelaskan hubungan antara permukaan untuk lingkungan sekitarnya (Al-Teyeche, 2003). Selain itu tekstur adalah sifat – sifat atau karakteristik yang memiliki oleh suatu daerah yang cukup besar sehingga secara alami sifat tersebut dapat berulang dalam daerah tersebut. Sehingga tekstur menjadi salah satu fitur yang penting. Ciri tekstur antara lain meliputi kehalusan *(smoothless)*, kekerasan *(coarseness),* dan keteraturan *(regularity)*.

Penggunaan fitur tekstur telah banyak digunakan secara luas oleh peneliti dalam menyelesaikan masalah pengenalan pola *(patern recognition)* dan *computer vision.* Secara umum, representasi tekstur dapat diklasifikasikan menjadi dua, yaitu : structural dan statistic (Gonzales, et al, 2002).

Untuk representasi tekstur secara statistic, dilakukan dengan menganalisa distribusi statistik dari intensitas citra termasuk diantaranya adalah teknik *multiresolution filtering seperti Glem Filter*.

Meskipun sebuah citra kaya akan informasi, namun seringkali citra yang kita miliki mengalami penurunan mutu (*degradasi*), misalnya mengandung cacat atau noise, warnanya terlalu kontras, kurang tajam, kabur *(blur)*, dan sebagainya tentu saja citra semacam ini menjadi lebih sulit diinterpretasi Karena informasi yang disampaikan oleh citra tersebut menjadi berkurang.

Agar citra yang mengalami gangguan mudah di interpretasi, maka citra tersebut perlu di manipulasi menjadi citra lain yang memiliki kualitas lebih baik. Bidang studi yang menyangkut hal ini adalah pengolahan citra *(image prosessing)* (Munir, 2004).

Konsep dasar pemrosesan suatu objek pada gambar menggunakan pengolahan citra diambil dari kemampuan indra pengelihatan manusia yang selanjutnya di hubungkan dengan kemampuan otak manusia. Dalam sejarahnya pengolahan citra telah di aplikasikan dalam berbagai bentuk, dengan tinggkat kesuksesan yang cukup besar. Seperti berbagai cabang ilmu lainnya, pengolahan citra menyangkut pula berbagai cabang ilmu lainnya, pengolahan citra menyangkut pula berbagai gabungan cabang – cabang ilmu. Diantaranya adalah optic, *electronic*, matematika, *fotografi,* dan teknologi computer (Gonzalez, 2002).

*Image prosessing* (pengolahan citra) mencakup dua aspek proses pengubah sebuah citra seperti sebagai berikut :

1. Meningkatkan kualitas informasi dari sebuah citra (gambar) yang digunakan untuk kepentingan interpretasi manusia.
2. Mengubah citra dari sebuah gambar yang digunakan untuk mempermudah pemrosesan persepsi mesin *autonomous* agar lebih mudah dalam mengambil keputusan.

Pada umumnya, objektifitas dari pengolahan citra adalah mentransformasi atau menganalisa suatu gambar sehingga informasi baru tentang gambar dibuat lebih jelas, Citra dalam pengolahannya dibagi menjadi tiga *level* proses, yaitu:

1. *Low\_level process* : proses-proses yang berhubungan dengan operasi primitif seperti image pre-processing untuk mengurangi *noise, input* dan *output-nya* berupa gambar.
2. *Mid-level process* : proses-proses yang berhubungan dengan tugas-tugas seperti segmentasi gambar (membagi gambar menjadi objek-objek), pengenalan *(recoqnition)* suatu objek individu. Pada *mid-level process, input* pada umumnya berupa gambar tetapi *output-nya* berupa atribut yang dihasilkan dari proses yang dilakukan gambar tersebut seperti garis, garis *contour*, dan objek-objek individu.
3. *High-level process* : proses-proses yang berhubungan dengan hasil dari *mid-level proces*.

Umumnya tahap-tahapan dalam pemrosesan citra untuk mendapatkan informasi, yaitu :

* Menentukan citra. Pertama kali harus menentukan objek citra yang akan diolah. Untuk kepentingan ini dapat menggunakan kamera atau scanner.
* Pemrosesan awal. Pemrosesan awal dijalakan sebelum proses utama dijalankan, bagian penting pada tahap ini adalah untuk menjalankan beberapa operasi dasar untuk menghasilkan gambar akhir sehingga lebih mudah dikerjakan pada tahap selanjutnya, Pada kasus ini contoh operasi yang biasa dilakukan adalah mengkonversi citra ke skala keabuan *(gray scalling)*, meningkatkan kontras, dan lainnya.
* Representasikan dan deskripsi. Pada tahap ini merujuk pada fitur khusus yang memungkinkan pengguna untuk membedakan objek.

Pengenalan, klasifikasi, dan interpretasi, pada tahap ini telah didapatkan kode pos dan merepresentasikannya pada sebuah string.

### Akuisisi Citra

Tujuan akuisisi citra adalah menentukan data yang diperlukan dan memilih metode perekaman *digital*, Tahap ini dimulai dari objek yang akan diambil gambarnya, persiapan alat-alat sampai pada percitraan[19]. Pencitraan adalah kegiatan transformasi dari citra tempak menjadi citra digital. Beberapa alat yang dapat digunakan seperti : kamera *digital, scanner*, foto sinar-x atau sinar infra merah, dan alat penangkapan bayang lainnya. Hasil dari akuisisi citra ini ditentukan oleh kemampuan *sensor* atau alat untuk mendigitalisasikan sinyal yang terkumpul pada sensor atau alat tersebut.

### Pra pengolahan Data citra

Tehnik pra-pemrosesan digunakan untuk mempersiapkan citra agar dapat menghasilkan ciri yang lebih baik pada pada tahap pemisahan cirri (ekstraksi ciri) terhadap proses pengenalan pola. Pra-pemrosesan adalah pengubahan data mentah untuk membantu kemampuan komputasional dan pencari ciri. Pada tahap pra-pemrosesan, kita cukup mengkaji metode atau pendekatan fungsi yang tepat untuk melakukan pemrosesan awal (pra-pemrosesan).

### Gray Scalling

*Gray scalling* adalah tehnik yang digunakan untuk mengubah citra berwarna menjadi bentuk tingkat aras keabuan. Jumlah warna citra *grey* adalah 256, karena citra *grey* jumlah *bit*-nya adalah 8, sehingga jumlah warnanya adalah =256, nilainya berada pada jangkauan 0-255. Pengubahan citra warna ke citra skala keabuan dapat menggunakan beberapa algoritma diantaranya :*averaging, Luminosity, Desaturation,* citra minimal dan *Maximal decamposition, single color channel.*

Metode *Luminosity* ini menggunakan perhitungan nilai berdasarkan tingkat luminositas. Metode ini juga bekerja dengan cara merata-ratakan nilai, tetapi metode ini membentuk rata-rata dengan bobot tertentu berdasarkan persepsi mata manusia. Mata manusia lebih sensitif terhadap warna hijau, oleh karena itu warna hijau diberi bobot yang lebih besar dari pada warna lainnya, Metode *luminosity* yang bekerja menggunakan persepsi mata manusia menghasilkan citra yang paling baik dari beberapa metode yang digunakan untuk konversi citra ke skala keabuan.

Adapun beberapa persamaan dari metode luminosity sebagai berikut :

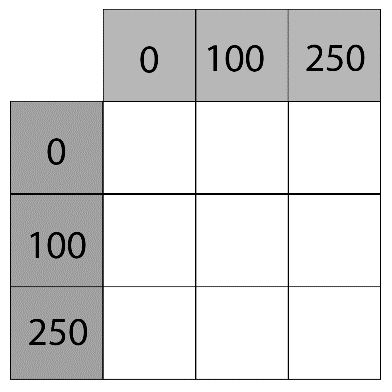
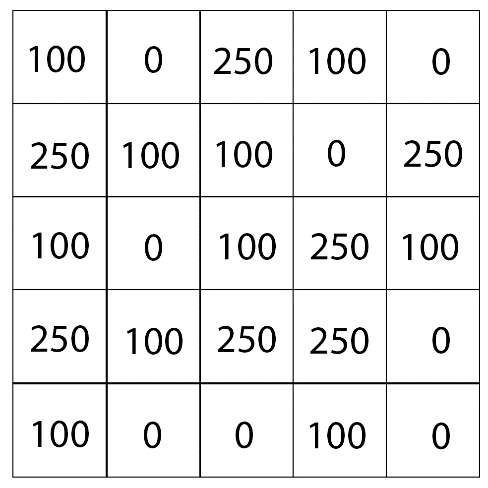
P1= (0,21R)+(0.71G)+(0.07B)

P2=(0.2125R)+(0.7154G)+(0.0721B)

P3=(0.50R)+(0.419G)+(0.081B)

P4=(0.229R)+(0.587G)+(0.114B)

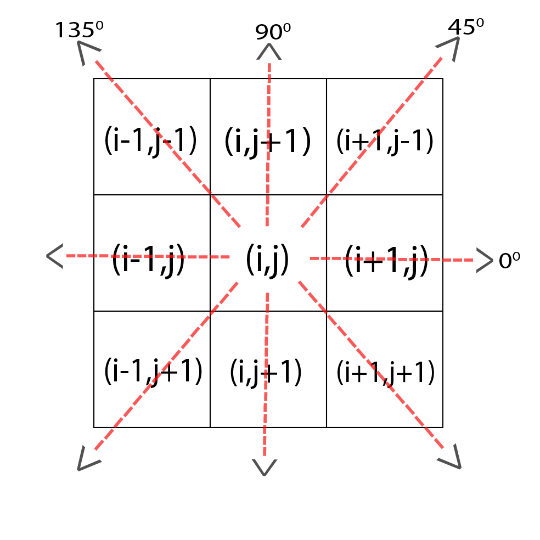
### GLCM ( *Gray Level Co-Occurrence Matrix* )

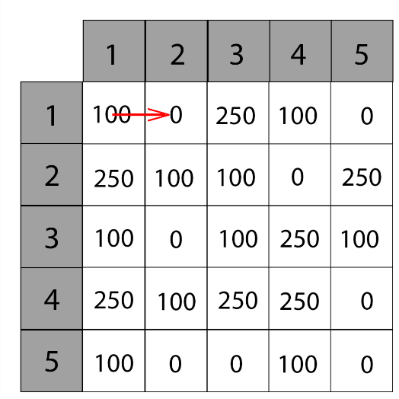
**** Glcm adalah salah satu metode statistika yang dapat digunakan untuk analisis tekstur. Langkah awal untuk mendapatkan informasi tekstur dari citra adalah menentukan *co-occurrence matrix*. *Co-occurrence matrix* menunjukkan hubungan spasial antara gray level dalam citra tekstur. Setiap elemen 𝑃 dengan posisi (𝑖,) pada glcm menunjukkan frekuensi relatif dua piksel gray level 𝑖 dan 𝑗. Co-occurrence matrix dalam penelitian ini dihitung pada arah horizontal 0° dengan jarak 1 piksel[20]. Setiap citra akan menghasilkan sebuah *co-occurrence matrix*. *Co-occurrence matrix* final dinormalisasi menggunakan persamaan untuk mengubah matriks glcm menjadi tabel probabilitas, dapat disimpulkan bahwa setiap cintra akan memiliki ukuran matrix kookurensi yang berbeda tergantung pada banyaknya tinggat keabuan pada citra tersebut.

1. Citra asli b. Matriks glcm

# Gambar 2.3 *hubungan antara ; (a) citra asli; (b) matrix glcm*

Gambar 2.3 adalah citra asli dengan ukuran 5x5. Dari table a dapat dilihat ada 3 tingkat keabuan, yaitu 0, 100, dan 250. Karena pada cintra asli terdapat 3 tingkat keabun, maka matriks kookurensi yang akan dibuat adalah matriks dengan ukuran 3x3 seperti pada gambar 2.1 bagian b. Matriks kookurensi dapat dituliskan p(i,j;*d,θ*). Dalam membangun hubungan antara pixel diperlukan sudut *θ* digunakan dalam menentukan arah tetangga dan *d* menentukan jarak special[11].

******Gambar 2.4** ilustrasi penentuan arah

 Dari ilustrasi pada gambar 2.4, dapat ditentukan kearah mana suatu pixel akan di hubungkan dengan pixel tetangganya. Dengan ilustrasi ini dapat ditentukan matriks kookurensi dari citra asli pada gambar 2.1 dimana matriks glcm yang dibentuk dengan jarak special *d=*1 dan sudut *θ=*0°, atau dapat dituliskan p(i,j;1,0°)[21].

**Gambar 2.5** *menentukan pixel tetangga*

Pada gambar 2.4 menentukan tetangga untuk membentuk hubungan antra pixel berdasarkan aturan p(i,j;1,0°). Karena jarak special dan sudut arah 0°, maka pixel tetangga dari *i* atau *pixel* *j* berada di kanan degan jarak 1, contoh jika i (₁,₁), maka j (₁,₂).

# D:\Projeck\Don't Open\Proposal\Mmbntuk matrix.pngGambar 2.6 *proses membentuk matriks glcm*

Gambar 2.4 menjelaskan bagaimana membentuk matriks kookurensi berada di sebalah kanan dan matriks asli sebelah kanan. Nilai cell didapatkan dari nilai hubungan atara pixel pada citra asli yang memenuhi sarat p(i,j;1,0°) dimana jarak antara satu pixel dengan pixel tetangga sama dengan 1 dengan arah sudut 0°. Dan yang ditunjukan pada lingkaran merah dimana nilai i=0 dan j=0, d=1 dan θ=0°, dan pada nilai i=100 dan j=0, d=0 dan θ=0° ada 6 hubungan pixel yang memenuhi. Cara yang sama untuk mengisi matriks glcm hingga semua nilai matriks glcm ditemukan[22].

Setelah memperoleh matriks glcm, ada fitur yang bisa di peroleh dari matriks tersebut, fitur tersebut antara lain *contrast, correlation, energy, homogeneity,* dan *entropy*[23].

1. Contrast
2. Correlation
3. Energy
4. Homogenity
5. Entropy

*Contrast, correlation, energy, homogeneity*, dan *entropy* dihitung untuk co-occurrence matrix yang diperoleh sehingga setiap fitur mempunyai sebuah nilai. Informasi fitur untuk setiap citra direpresentasikan sebagai sebuah vektor yang memiliki 3 elemen fitur. Informasi fitur ini menjadi masukan untuk proses klasifikasi dengan k-nn[20].

Analisa tekstur (*texture analysis*) dikerjakan menggunakan GLCM (*Gray Level Co-occurrence Matrix*) yaitu dengan manusia mengenal tekstur lembut dan kasar walaupun tangan tidak menyentuh benda tersebut, tapi otak mampu membedakan hal tersebut melalui visualisasi yang ditangkap oleh mata. Tekstur dapat dicirikan sebagai berikut

* 1. Pengulangan pola dari variasi lokal sehingga membentuk kesatuan yang utuh.
  2. Menyediakan informasi susunan spasial dari warna dan intensitas citra.
  3. Dicirikan dengan distribusi spasial dari levelintensitas dari nilai pixel ketetanggaan.
  4. Tidak bisa didefinisikan sebagai suatu satu point / nilai tertentu karena merupakan sebuah pola /kesatuan.

### K–NN ( *K–Nearest Neighbor*)

Algoritma metode k-nn sangatlah sederhana, bekerja berdasarkan jarak terpendek dari query instance ke data traning unutk menentukan knnnya. Data traning diproyeksikan ke ruang berdimensi banyak, dimana masing – masing dimensi mempresentasikan fitur dari data. Ruang ini menjadi bagian – bagian berdasarkan klasifikasi data traning. Salah satu teknik klasifikasi paling dasar dan sederhana adalah K-Nearest Neighbor (k-nn)[24]. Teknik klasifikasi ini memiliki beberapa kelebihan yaitu :

1. Pelatihan sangat cepat
2. Sederhana dan mudah dipelajari
3. Tahan terhadap data pelatihan yang memiliki derau
4. Efektif jika data pelatihan besar

Metode klasifikasi K-Nearest Neighbor melakukan proses pencocokan/pengenalan berdasarkan jumlah tetangga terdekat untuk penentukan kelasnya. Untuk mencari jarak kelas ada beberapa cara yaitu dengan euclideaen distance.Tahapan dalam metode klasifikasi *K-Nearest Neighbor* yaitu :

1. Menentukan niai k
2. Menghitung jarak antara citra testing dengan seluruh citra pada database menggunakan persamaan euclidean distance, persamaan dan menentukan citra terdekat dengan citra testing berdasarkan nilai k.

Rumus dari euclidean distance

Keterangan:

D (a, b) : jarak Euclidian

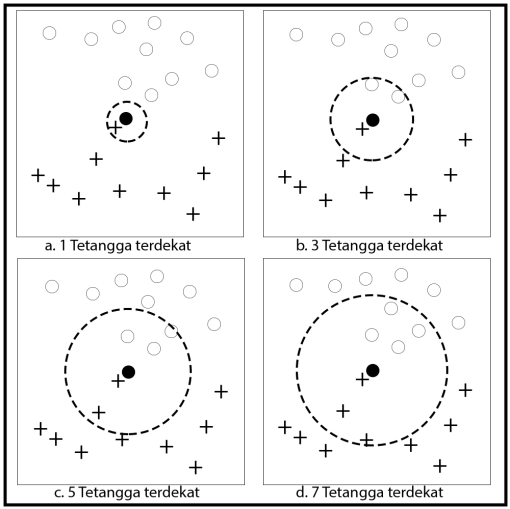
x : data 1

y : data 2

i : fitur

Ke – n : jumlah fitur

1. Menentukan hasil klasifikasi berdasarkan kelas yang memiliki anggota terbanyak.
2. jika terjadi konflik atau keadaan seimbang pada kelas dengan jumlah anggota yang sama maka digunakan pemecahan konflik.



**Gambar 2.7** *k-nn dengan k tetangga, (a) 1-nn, (b) 3-nn, (c) 5-nn, (d) 7-nn*

Gambar 2.7 menunjukan jumlah tetangga yang paling dekat yang dapat dimuat dalam rentang nilai k yang telah ditentukan. Gambar a menunjukan 1 tatangga terdekat, gambar b menunjukan 3 tetangga terdekat, gambar c 5 tetangga terdekat, dan d 7 tetangga terdekat. Untuk menentukan jarak tetangga terdekat, terdapat berbagai macam cara salah satunya adalah dengan pendekatan *euclidean distance*[21].

Contoh Perhitungan K-Nearest Neighbor:

Terdiri dari 2 atribut dengan skala kuantitatif sebagai data training yaitu X1 dan X2 serta Y yaitu kelas baik dan buruk seperti pada 2.2. [25].

# Tabel 2.2 Klasifikasi Data Traning

|  |  |  |
| --- | --- | --- |
| **X1** | **X2** | **Y** |
| 7 | 7 | Buruk |
| 7 | 4 | Buruk |
| 3 | 4 | Baik |
| 1 | 4 | Baik |

Terdapat data testing yaitu X1 = 3 dan X2 = 7, tentukan nilai Y! Langkah penyelesaian:

1. Tentukan parameter K = jumlah tetangga terdekat, misalkan ditetapkan K = 3
2. Hitung jarak antara data baru dengan semua data training seperti pada tabel 2.3.

# Tabel 2.3 Perhitungan Kuadrat Jarak Data training dengan Data testing

|  |  |  |
| --- | --- | --- |
| **X1** | **X2** | **Kuadrat Jarak dengan Data Baru (3,7)** |
| 7 | 7 | (7-3)2 + (7-7)2 = 16 |
| 7 | 4 | (7-3)2 + (4-7)2 = 25 |
| 3 | 4 | (3-3)2 + (4-7)2 = 9 |
| 1 | 4 | (1-3)2 + (4-7)2 = 13 |

1. Urutkan hasil kuadrat jarak tersebut secara ascending dan tetapkan tetangga terdekat berdasarkan nilai K (dimana K = 3) seperti pada tabel 2.4 dan tabel 2.5.

# Tabel 2.4 Penentuan 3 Tetangga Terdekat dari Data testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **X1** | **X2** | **Euclidean** | **Peringkat Jarak** | **Tetangga Terdekat** |
| 7 | 7 |  | 3 | YA |
| 7 | 4 |  | 4 | TIDAK |
| 3 | 4 |  | 1 | YA |
| 1 | 4 |  | 2 | YA |

**Tabel 2.5** Klasifikasi Kelas Tetangga Terdekat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **X1** | **X2** | **EuclideanDistance** | **Peringkat Jarak** | **Tetangga Terdekat** | **Y** |
| 7 | 7 | 4 | 3 | YA | BURUK |
| 7 | 4 | 5 | 4 | TIDAK | - |
| 3 | 4 | 3 | 1 | YA | BAIK |
| 1 | 4 | 3,6 | 2 | YA | BAIK |

1. Berdasarkan hasil dari Tabel 2.5 dimana nilai K = 3, mayoritas hasil Y yang diperoleh yaitu “Baik”. Jadi hasil klasifikasi data testing dengan nilai X1 = 3 dan X2 = 7 adalah Y = kelas Baik[26].

### Evaluasi Model

Evaluasi model dalam penelitian ini dilakukan untuk menguji performansi dari sistem yang dibangun.Beberapacarauntuk melakukan evaluasi adalah dengan menghitung akurasi (*accuracy*), presisi (*precision*), dan *recall* dari hasil pengolahan. Metode yang digunakan untuk perhitungan akurasi adalah *confusion matrix*dan *cross validation* untuk mengevaluasi kinerja model atau algoritma*.*

*Confusion matrix* adalah suatu metode yang digunakan untuk melakukan perhitungan akurasi pada konsep data mining*.* Evaluasi dengan *confusion matrix* menghasilkan nilai akurasi, presisi dan recall akurasi dalam klasifikasi adalah persentase ketepatan record data yang diklasifikasikan secara benar setelah dilakukan pengujian pada hasil klasifikasi. Presisi atau *confidence* adalah proporsi kasus yang diprediksi positif yang juga positif benar pada data yang sebenarnya. *Recall* atau *sensitivity* adalah proporsi kasus positif yang sebenarnya yang diprediksi positif secara benar[27].

*Cross validation* adalah salah satu metode yang digunakan untuk mengetahui rata – rata keberhasilan suatu system dengan cara melakukan perulangan dengan mengacak atribut input sehingga system tersebut teruji untuk beberapa atribut input yang acak dan cross validation adalah teknik umum untuk memperkirakan kinerja pengklasifikasi[28].

|  |  |  |
| --- | --- | --- |
| Klasifikasi sebenarnya | Hasil klasifikasi | |
| + | - |
| + | Benar positif (a) | Salah negatif (b) |
| - | Salah positif (c) | Benar negatif(d) |

# Tabel 2.6 model confusion mateix

Perhitungan akurasi dengan tabel confusion matrix adalah sebagai berikut:

Akurasi = (a+d)/(a+b+c+d)

Presisi didefinisikan sebagai rasio item relevan yang dipilih terhadap semua item yang terpilih. Presisi dapat diartikan sebagai kecocokan antara permintaan informasi dengan jawaban terhadap permintaan tersebut. Rumus presisi adalah:

Presisi = a/(c+a)

Recall didefinisikan sebagai rasio dari item relevan yang dipilih terhadap total jumlah item relevan yang tersedia. Recall dihitung dengan rumus:

Recall = a/(a+d)

Presisi dan recall dapat diberi nilai dalam bentuk angka dengan menggunakan perhitungan persentase (1-100%) atau dengan menggunakan bilangan antara 0-1. Sistem rekomendasi akan dianggap baik jika nilai presisi dan recallnya tinggi[26].

### 2. 2. 10. UML (*Unified Modeling Language*)

*Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membanngun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan system[29]. Atau suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual dan juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek[30].

Tujuan pemanfaatan UML :

1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
2. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
3. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
4. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
5. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
6. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
7. Memiliki integrasi praktik terbaik.

Untuk perancangan sebuah aplikasi dengaan menggunakan UML, UML mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Tipe diagram UML dapat dilihat pada table berikut ini :

# Tabel 2.7 Tipe Diagram UML (*Unified Modeling Language* )

|  |  |
| --- | --- |
| Diagram | Tujuan |
| Activity | Perilaku prosedural dan paralel |
| Class | Class, Fitur dan relasinya |
| Communication | Interaksi diantara obyek. Lebih menekankan ke link |
| Component | Struktur dan koneksi dari komponen |
| Composite structure | Dekomposisi sebuah class saat runtime |
| Deployment | Penyebaran/ instalasi ke klien |
| Interaction Overview | Gabungan antara activity dan sequence diagram |
| Object | Contoh konfigurasi instance |
| Package | Struktur hierarki saat kompilasi |
| Sequence | Interaksi antara obyek. Lebih menekankan pada urutan |
| State Machine | Bagaimana event mengubah sebuah obyek |
| Timing | Interkasi antara obyek. Lebih menekankan pada watu |
| Use Case | Bagaimana User berintaraksi dengan sebuah system |

### 2. 2. 11. Python

Python adalah bahasa pemrograman model skrip (scripting language) yang berorientasi obyek. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi. Python merupakan bahasa pemrograman yang freeware atau perangkat bebas dalam arti sebenarnya, tidak ada batasan dalam penyalinannya atau mendistribusikannya. Lengkap dengan source codenya, debugger dan profiler, antarmuka yang terkandung di dalamnya untuk pelayanan antarmuka, fungsi sistem, gui (antarmuka pengguna grafis), dan basis datanya[31].

Beberapa fitur yang dimiliki python adalah:

1. Memiliki kepustakaan yang luas; dalam distribusi python telah disediakan modul-modul.
2. Memiliki tata bahasa yang jernih dan mudah dipelajari.
3. Memiliki aturan layout kode sumber yang memudahkan pengecekan, pembacaan kembali dan penulisan ulang kode sumber.
4. Berorientasi obyek.
5. Dapat dibangun dengan bahasa python maupun c/c++.

## Kerangka Pikir

**PEMODELAN / ABSTRAKSI**

GLCM

Pra Pengolahan

Ekstraksi Fitur

**MASALAH**

1. Bagaimana peningkatan uji coba akurasi metode GLCM apabila diterapkan metode K-NN dalam proses identifikasi pengenalan kematangan buah pisang ?

Data Public

Pengumpulan Dataset

K-Nearest Neighboard (KNN)

Klasifikasi

Evaluasi

Confucion Matrix

**TUJUAN**

1. Untuk mengetahui hasil akurasi metode GLCM dan KNN dalam proses klasifikasi pengenalan kematangan buah pisang

Histogram equalization dan deteksi citra buah pisang

**BAB III  
METODE PENELITIAN**

1. **Jenis, Metode, Subjek, Objek, Waktu dan Lokasi Penelitian**

Dipandang dari tingkat penerapannya, maka penelitian ini merupakan penelitian terapan karena penelitian ini berfokus penerapannya untuk memberikan solusi atas permasalahan secara praktis. Dipandang dari jenis informasi yang diolah, maka penelitian ini merupakan penelitian kuantitatif. Dipandang dari perlakuan terhadap data, maka penelitian ini merupakan penelitian konfirmatori. Penelitian ini menggunakan metode penelitian eksperimen. Dengan demikian jenis penelitian ini adalah penelitian eksperimental. Subjek penelitian ini adalah pengenalan kematangan buah pisang. Penelitian ini dimulai dari Januari 2019 sampai dengan Juni 2019.

1. **Pengumpulan Data**

Dataset yang digunakan dalam penelitian ini adalah dataset public dengan nama *Tier-Based Dataset : Musa – Acuminata Banana Fruit Species*, dataset yang di unduh dari https://data.mendeley.com/datasets/zk3tkxndjw, dataset ini merupakan data yang pada umumnya digunakan dalam eksperiment untuk klasifikasi pengenalan buah pisang. Dataset ini akan dibagi menjadi dua bagian yaitu data training dan data testing. Berikut penjelasannya mengenai *Tier-Based Dataset : Musa–Acuminata Banana Fruit Species* :

Jumlah gambar : 1.191 Gambar

Objek : 100 (Gambar yang diambil)

Matang = 50 Gambar

Belum Matang = 50 Gambar

Tingkat Kematangan : Belum Matang & Matang

Format : JPG

Quality : Low Quality

URL : https://data.mendeley.com/datasets/zk3tkxndjw

1. **Pemodelan**

**Gambar 3. 1** Proses pemodelan

Klasifikasi

Data Uji

Pemrosesan Citra Buah Pisang

Sistem Pengenalan Kematangan Buah Pisang

Data Latih

Normalisasi

Hasil Pelathian

Eksraksi Fitur Citra

GLCM

Data Citra Buah Pisang

Konversi Warna

Pra Pengolahan

Proses Pengenalan / Klasifikasi

Pembelajaran

Nilai Ekstraksi Buah Pisang

K-NN

Hasil Pengenalan

### Pra Pengolahan

Sebelum data diolah, terlebih dahulu dilakukan intensity adjustment dan segmentasi. Hal ini dilakukan karena intensity adjustment bertujuan untuk meningkatkan cintra yang gelap dan proses segmentasi ialah proses pemisahan antara objek yang dikehendaki dan objek yang tidak dikehendaki.

### Ekstraksi Ciri

Ekstraksi ciri berfungsi sebagai deteksi fitur dari suatu citra. Ciri yang dapat digunakan membedakan citra satu dengan citra yan g lain, di antaranya adalah ciri bentuk, ukuran, geometri, tekstur, dan warna. Pada tahap ini digunakan *Gray Level Co-occurrence Matrix*. Masing-masing citra diekstrak cirinya berdasarkan parameter-parameternya tertentu dan dikelompokan pada kelas tertentu, parameter – parameter yang digunakan berupa *Contras, Correlation, Energy, Homogeneity*, dan *Entropy*.

### Data Traning

Data Traning berupa data yang telah terekstrak cirinya yang selanjutnya akan dilatih menggunakan algoritma *K-Nearest Neighbor*, jumlah data traning yang digunakan berupa 10% dari sampel buah pisang dengan citra buah pisang yang berbeda. Algoritma ini akan menentukan/mencari bobot yang terbaik. Data traning ini berupa hasil dari ekstraksi *Gray Level Co-occurrence Matrix* terhadap kematangan buah pisang.

### Traning Menggunakan K-NN

Traning menggunakan K-NN yaitu menjadikan data traning menjadi data inputan. Algoritma KNN bekerja berdasarkan jarak terpendek dari training dan test dalam penentuannya, kemudian menggambil mayoritas hasil ketentuan yang telah didapatkan untuk dijadikan prediksi dari test.

### Model

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

### Data Testing

Data testing merupakan data yang telah terekstraksi cirinya, untuk menguji data yang telah dilatih, jumlah data traning yang digunakan berupa 90% dari sampel pisang dengan tingkat kematangan yang berbeda. Data testing digunakan untuk mengetahui keberhasilan classifier berhasi melakukan klasifikasi, data testing merupakan ciri dari hasil ekstraksi dari Gray Level Co-oocurrence Matrix terhadap kematangan buah pisang.

### Hasil Klasifikasi

Hasil klasifikasi merupakan output, pada data testing yang didapatkan dari proses klasifikasi yang menggunakan algoritma K-NN berdasarkan model yang diperoleh dari data traning

### Evaluasi

Evaluasi bertujuan untuk mengetahiu hasilkinerja dari metode yang digunakan, evaluasi dilakukan pada data *testing* dan *output* yang dihasilakan akan dimasukan ke dalam *Confusion Matrix* untuk menghitung nilai akurasi.

# BAB IV

**HASIL PENELITIAN**

## Hasil Pengumpulan Data

Data yang digunakan berupa gambar atau *images* berupa dua tingkat kematangan. Data yang diambil dari dataset public dengan nama Tier-Based Dataset : Musa – Acuminata Banana Fruit Species, dataset yang di unduh dari https://data.mendeley.com/datasets/zk3tkxndjw. Berikut ini beberapa data yang telah dikumpulkan :

# Tabel 4.1 Data Gambar Kematangan Pisang

|  |  |
| --- | --- |
| Belum Matang | BelumMatang3.jpgBelumMatang4.jpgBelumMatang5.jpgBelumMatang6.jpgBelumMatang7.jpgBelumMatang8.jpg |
| Matang | M1.jpgM2.jpg**SangatMatang6.jpg**Matang1.jpgMatang2.jpgMatang4.jpg |

## Hasil Pemodelan

## Pra Pengolahan

Pra-pengolahan citra (*image pre-processing*), yaitu proses paling awal dalam pengolahan citra sebelum proses utama dilakukan. Pada tahap ini citra buah pisang yang sudah ada dikonversi agar diperoleh data citra buah pisang yang sesuai kebutuhan. Tahap ini berfungsi untuk menormalisasi citra buah pisang dari permasalahan luminasi yang teralalu gelap atau terlalu terang sehingga dapat meningkatkan performansi dari sistem pengenalan kematangan buah pisang. Pra-pengolahan dalam penelitian ini dilakukan dengan 2 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

# Tabel 4.2 Proses perubahan citra asli menjadi grayscale

|  |  |  |
| --- | --- | --- |
| **Tingkat Kematangan** | **Image Asli** | **Image Grayscale** |
| Belum Matang | BelumMatang1.jpg | Screenshot (43).png |
| Matang | M2.jpg | Screenshot (44).png |

* + 1. Normalisasi citra

Setelah citra asli di konversi ke citra abu-abu maka pra-pengolahan selanjutnya adalah normalilasi 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

|  |  |  |
| --- | --- | --- |
| **Tingkat Kematangan** | **Citra Grayscale** | **Citra Histogram** |
| Belum Matang | Screenshot (43).png | Screenshot (46).png |
| Matang | Screenshot (44).png | Screenshot (45).png |

1. Deteksi Tingkat Kematangan

Setelah citra histrogram di dapatkan maka pra-pengolahan selanjutnya adalah pendeteksian buah pisang terhadap citra dengan thresholding. Thresholding merupakan salah satu metode segmentasi citra yang memisahkan antara objek dan background dalam suatu citra berdasarkan pada perbedaan tingkat kecerahannya atau gelap terangnya. Region citra cenderung gelap akan dibuat semakin gelap (hitam sempurna dinilai intensitas sebesar 0), sedangkan region citra yang cenderung terang akan dibuat semakin terang (putih sempurna dinilai intensitas sebesar 1)

# Tabel 4.4 Proses Deteksi Kematangan

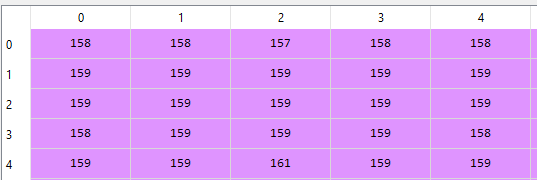
|  |  |  |
| --- | --- | --- |
| **Tingkat Kematangan** | **Citra Historgram** | **Citra Thresholding** |
| Belum Matang | **C:\Users\ASUS\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot (54).png** | **C:\Users\ASUS\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot (53).png** |
| Matang |  | **C:\Users\ASUS\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot (52).png** |

## Ekstraksi Fitur

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

Contoh: Pada jarak 1 dan orientasi arah sudut 0o

Nilai pada Gambar 4.1 berupa matrix 4 x 4, dan matrix sebenarnya 101 x 101 yang disederhanakan unruk memudahkan perhitungan



# Gambar 4.1 Nilai dari suatu citra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i/j | 157 | 158 | 159 | 161 |
| 157 | 0 | 1 | 0 | 0 |
| 158 | 1 | 2 | 1 | 0 |
| 159 | 0 | 1 | 12 | 1 |
| 161 | 0 | 0 | 1 | 0 |

# Tabel 4.5 Matrix GLCM

Jumlah Total nilai GLCM pada Tabel 4.4 Matrix GLCM = 20

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i/j | 157 | 158 | 159 | 161 |
| 157 | 0 | 0.05 | 0 | 0 |
| 158 | 0.05 | 0.1 | 0.05 | 0 |
| 159 | 0 | 0.05 | 0,6 | 0.05 |
| 161 | 0 | 0 | 0.05 | 0 |

# Tabel 4.6 Matrix Ternormalisasi

1. Menghitung nilai *Contras*

Contras = (157-158)2(0,05) + (158-157)2(0,05) + (158-158)2(0,1) + (158-159)2(0,05) + (159-158)2(0,05) + (159-159)2(0,6) + (159-161)2(0,05) + (161-159)2(0,05)

Contras = 0,1 + 0,1 + 0 + 0,1 + 0,1 + 0 + 0,2 + 0,2

Contras = **0.8**

1. Menghitung nilai *Correlation*

Correlation =

= (157 \* (0.05)) + (158 \* (0.05)) + (158 \* (0.1)) + (158 \* (0.05)) + (159 \* (0.05)) + (159 \* (0.6)) + (159 \* (0.05)) + (161 x (0.05))

*µi* = 7.8 + 7.9 + 15.8 + 7.9 + 7.95 + 95.4 + 7.95 + 8.05

*µi =* **158.75**

*µj =* (157 \* (0.05)) + (158 \* (0.05)) + (158 \* (0.1)) + (158 \* (0.05)) + (159 \* (0.05)) + (159 \* (0.6)) + (159 \* (0.05)) + (161 x (0.05))

*µj =* 7.8 + 7.9 + 15.8 + 7.9 + 7.95 + 95.4 + 7.95 + 8.05

*µj =* **158.75**

**0,550**

**0,550**

Correlation = ((157 - 158.75)\*(158 – 158.75)\*( 0,05)/(+ ((158 - 158,75)\*(157 - 158.75)\*( 0,05)/(+ ((158 - 158,75)\*(158 - 158.75)\*( 0,1)/(+ ((158 – 158.75)\*(159– 158.75)\*(0,05)/(+ ((159 - 158,75)\*(158 - 158.75)\*( 0,05)/() + ((159 - 158,75)\*(159 - 158.75)\*( 0,6)/() + ((159 - 158,75)\*(161 - 158.75)\*( 0,05)/() + ((161 - 158,75)\*(159 - 158.75)\*( 0,05)/()

Correlation = **0,867**

1. Menghitung nilai *Energy*

Energy = ((0,05)2)+ ((0,05)2 )+ ((0,1)2)+ ((0,05)2)+ ((0,05)2) + ((0,6)2) + ((0,05)2) + ((0,05)2)

Energy = **0,385**

1. Menghitung nilai *Homogenity*

Homogenity = ((0,05) / 1 + ( 157-158)) + ((0,05) / 1 + (158-157)) + ((0,1) / 1 + ( 158-158)) + ((0,05) / 1 + (158-159)) + ((0,05) / 1 + (159-158)) + ((0,6) / 1 + (159-159)) + ((0,05) / 1 + ( 159-161)) + ((0,05) / 1 + ( 161-159))

Homogenity = **1**

1. Menghitung nilai *Entropy*

Entropy = (0,05.log 0,05)+ (0,05.log 0,05)+ (0,1.log 0,1)+ (0,05.log 0,05)+ (0,05.log 0,05) + (0,6.log 0,6) + (0,05.log 0,05) + (0,05.log 0,05)

Entropy = **0,623**

Lima ciri yang di dapatkan dari matrix *Co-Occurrence* inilah yang akan digunakan untuk tahap klasifikasi.

## Cross Validation

Total dari keseluruhan data yang digunakan sebagai data traning ada 100 data, yang akan di uji akurasi dengan cara di bagi menjadi 2 literasi, yang masing – masing iterasi berjumlah 50 dengan arah GLCM 45o dan jarak 4.

1. Akurasi Iterasi 1 = Akurasi = = 66 %
2. Akurasi Iterasi 2 = Akurasi = = 18 %
3. Total akurasi = 84 %

## Klasifikasi

1. Menghitung nilai citra matrix pada data pembanding

# Tabel 4.7 Nilai Data Citra Matrix dan Data Pembanding GLCM

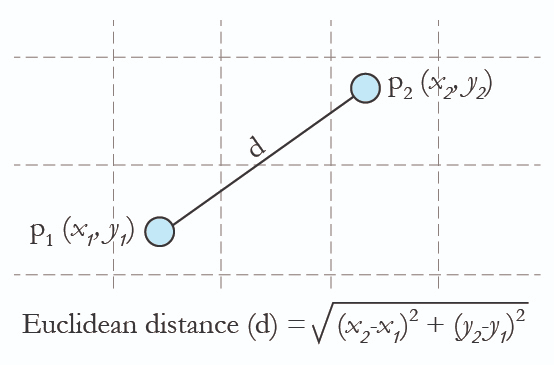
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ASM(X1)** | **Ene(X2)** | **Con(X3)** | **Cor(X4)** | **Hom(X5)** | **Dis(X6)** | **Y** |
| 0.6030155 | 0.7765407 | 1470.6912 | 0.9396701 | 0.9773830 | 5.7674165 | Matang |
| 0.6060717 | 0.7785060 | 516.87005 | 0.9794100 | 0.9920513 | 2.0269413 | Matang |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 0.0 | B. Matang |
| 0.5978262 | 0.773192 | 1256.2653 | 0.9495873 | 0.980680 | 4.9265306 | Matang |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 0.0 | B. Matang |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 0.0 | B. Matang |
| 0.5549735 | 0.7449654 | 1204.1666 | 0.9566167 | 0.9814817 | 4.7222222 | Matang |
| Data Pembanding | | | | | | |
| 0.6156551 | 0.7846369 | 1773.6047 | 0.9237710 | 0.972724 | 6.9553128 | ? |

1. Menghitung nilai fitur

**Tabel 4.8** Nilai Fitur GLCM

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **X1** | **X2** | **X3** | **X4** | **X5** | **NILAI FITUR** |
| 0.60301555 | 0.77654076 | 1470.69121 | 0.939670 | 0.977383 | 302.9158775 |
| 0.60607171 | 0.77850607 | 516.870053 | 0.979410 | 0.992051 | 1256.744378 |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 1773.618461 |
| 0.59782628 | 0.77319226 | 1256.26530 | 0.949587 | 0.980680 | 517.3434396 |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 1773.618461 |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 1773.618461 |
| 0.55497352 | 0.74496545 | 1204.166667 | 0.956616 | 0.981481 | 569.4424842 |
| Data Pembanding | | | | | |
| 0.6156551 | 0.7846369 | 1773.60476 | 0.923771 | 0.972724 | ? |

1. Mencari Nilai Euclidean Distance dari Nilai Fitur GLCM dan Menentukan Peringkat



Rumus :

# Tabel 4.9 Perhitungan Euclidean

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **(X1)** | **(X2)** | **(X3)** | **(X4)** | **(X5)** | **(X6)** | **Euclidean** | **Peringkat** | **Y** |
| 0.6030155 | 0.776540 | 1470.6912 | 0.939670 | 0.97738302 | 5.767416 | √302.915= 17.404478 | 1 | Matang |
| 0.6060717 | 0.778506 | 516.87005 | 0.979410 | 0.99205133 | 2.026941 | √125.674=  35.450590 | 4 | Matang |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 0.0 | √177.361=  42.114349 | 5 | B. Matang |
| 0.597826285 | 0.773192269 | 1256.265306 | 0.949587318 | 0.980680569 | 4.926530612 | √517.344=  22.745184 | 2 | Matang |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 0.0 | √177.361=42.114349 | 5 | B. Matang |
| 1.0 | 1.0 | 0.0 | 1.0 | 1.0 | 0.0 | √177.361=42.114349 | 5 | B. Matang |
| 0.554973524 | 0.744965451 | 1204.166667 | 0.956616764 | 0.981481766 | 4.722222222 | √569.443=  23.862994 | 3 | Matang |

# Tabel 4.10 Tetangga Terdekat

|  |  |  |  |
| --- | --- | --- | --- |
| **Euclidean** | **Peringkat** | **Tetangga Terdekat** | **Y = Kategori KNN** |
| 302.9158775 | 1 | Ya | Matang |
| 1256.744769 | 4 | Tidak | Matang |
| 1773.618743 | 5 | Tidak | Belum Matang |
| 517.3443691 | 2 | Ya | Matang |
| 1773.618743 | 5 | Tidak | Belum Matang |
| 1773.618743 | 5 | Tidak | Belum Matang |
| 569.44333 | 3 | Ya | Matang |

Berdasarkan hasil dari Tabel diatas dimana nilai K = 3, mayoritas hasil Y yang diperoleh yaitu “Matang”. Jadi hasil klasifikasi data testing dengan nilai X1 = 0.615655154, X2 = 0.784636957, X3 = 1773.604767, X4 = 0.923771057, X5 = 0.972724683 dan X6 = 0.972724683

## Evaluasi

## Confusion Matrix

Pada penelitian ini menggunakan confudion Matrix sebagai metode. Dalam perhitungan akurasi pada penerapan deteksi pengenalan kematangan pisang. Evaluasi kinerja deteksi kematangan di dasarkan dari jumlah pengujian objek salah dan benar yang di deteksi dapat dilihat dari table berikut :

# Tabel 4.11 Hasil Data Testing

|  |  |  |
| --- | --- | --- |
| **Image Testing** | **Klasifikasi** | **Hipotesis** |
| M2.jpg | Matang | Matang |
| BelumMatang8.jpg | Belum Matang | Matang |
| BelumMatang3.jpg | Belum Matang | Belum Matang |
| Matang4.jpg | Matang | Matang |
| BelumMatang7.jpg | Belum Matang | Belum Matang |
| BelumMatang6.jpg | Belum Matang | Belum Matang |
| Matang1.jpg | Matang | Matang |
| 2018-01-07-01-12-29.jpg | Matang | Matang |

# Tabel 4.12 Hasil Data Uji

|  |  |  |
| --- | --- | --- |
|  | Matang | Belum Matang |
| Matang | 4 | 0 |
| Belum Matang | 1 | 3 |

Jumlah data yang diklasifikasi dengan benar = 7

Jumlah data yang diklasifikasi salah = 1

Akurasi =

Akurasi== 76 %

# BAB V

**PEMBAHASAN**

## Pembahasan Model

Pada percobaan model K-NN, dengan jumlah K = 2, dan Arah GLCM = 90o dengan jarak = 1, maka mendapatkan nilai akurasi tertinggi Sebesar = 71%, dengan data training = 100, yaitu belum matang = 50, dan matang = 50, Dengan Hasil kematangan belum matang.

Sedangkan pada percobaan model K-NN, dengan jumlah K = 2, dan Arah GLCM = 45o dengan jarak = 2 dan 3, maka mendapatkan nilai akurasi tertinggi sebesar = 74%, data training = 100, yaitu belum matang = 50, dan matang = 50, Dengan Hasil kematangan belum matang.

Pada percobaan model K-NN, dengan jumlah K = 2, dan Arah GLCM = 45o dengan jarak = 3 dan 4, Maka Mendapatkan Nilai Akurasi Sebesar = 0.73847320564 atau 74%, dengan data training = 100, yaitu belum matang = 50, dan matang = 50, Data Testing = 3, Dengan Hasil kematangan belum matang.

Pada percobaan model K-NN, dengan jumlah K = 2, dan Arah GLCM = 45o dengan jarak = 1, Maka Mendapatkan Nilai Akurasi Sebesar = 80%, dengan data training = 100, yaitu belum matang = 50, dan matang = 50, Data Testing = 3, Dengan Hasil kematangan matang.

Pada percobaan dengan jumlah K = 3, dan Arah GLCM = 90o dengan jarak = 3, maka mendapatkan nilai akurasi tertinggi Sebesar = 74%, dengan data training = 100, yaitu belum matang = 50, dan matang = 50, Data Testing = 3, Dengan Hasil kematangan matang.

Sedangkan pada percobaan dengan jumlah K = 3, dan Arah GLCM = 0o dan 135o dengan jarak = 3 , maka mendapatkan nilai akurasi tertinggi sebesar = 75%, dengan data training = 100, yaitu belum matang = 50, dan matang = 50 Data Testing = 3, Dengan Hasil kematangan matang.

Dan dengan jumlah K = 3, dan Arah GLCM = 45o dan 90o dengan jarak = 3, Maka Mendapatkan Nilai Akurasi Sebesar = 71 %, dengan data training = 100, yaitu belum matang = 50, dan matang = 50, Data Testing = 3, Dengan Hasil kematangan belum matang.

Pada percobaan model K-NN, dengan jumlah K = 3, dan Arah GLCM = 90o dengan jarak = 3, Maka Mendapatkan Nilai Akurasi Sebesar = 74%, dengan data training = 100, yaitu data belum matang = 50, dan matang = 50 Data Testing = 3, Dengan Hasil kematangan matang.

# Tabel 5.1 Hasil Uji Coba Nilai K = 2

|  |  |  |  |
| --- | --- | --- | --- |
| Belum Matang | | Matang | |
| 0°,1,2 | 84% | 0°,1,2 | X |
| 0°,2,2 | 77% | 0°,2,2 | 77% |
| 0°,3,2 | 78% | 0°,3,2 | X |
| 45°,1,2 | X | 45°,1,2 | 80% |
| 45°,2,2 | 74% | 45°,2,2 | 74% |
| 45°,3,2 | X | 45°,3,2 | 71% |
| 90°,1,2 | 71% | 90°,1,2 | X |
| 90°,2,2 | X | 90°,2,2 | 71% |
| 90°,3,2 | 69% | 90°,3,2 | X |
| 135°,1,2 | X | 135°,1,2 | 81% |
| 135°,2,2 | 75% | 135°,2,2 | 75% |
| 135°,3,2 | 77% | 135°,3,2 | 74% |

|  |  |  |  |
| --- | --- | --- | --- |
| Tabel 5.2 Hasil Uji Coba Nilai K = 3 | | | |
| Belum Matang | | Matang | |
| 0°,1,3 | X | 0°,1,3 | 81% |
| 0°,2,3 | 76% | 0°,2,3 | 75% |
| 0°,3,3 | 77% | 0°,3,3 | X |
| 45°,1,3 | 80% | 45°,1,3 | 80% |
| 45°,2,3 | X | 45°,2,3 | 78% |
| 45°,3,3 | 76% | 45°,3,3 | 71% |
| 90°,1,3 | 71% | 90°,1,3 | 71% |
| 90°,2,3 | 65% | 90°,2,3 | 65% |
| 90°,3,3 | X | 90°,3,3 | 74% |
| 135°,1,3 | 81% | 135°,1,3 | X |
| 135°,2,3 | 75% | 135°,2,3 | 75% |
| 135°,3,3 | 75% | 135°,3,3 | X |

Pada tahap pengujian metode terdapat kendalah pada tahap pengenalan kematangan, karena penggunaan sampel buah pisang harus diperhatikan dari contras dan posisi dari data sampel, karenan ada beberapa sampel data yang tidak biasa dikenali atau tidak terdeteksi. Arah dan jarak GLCM sangan berpengaruh untuk pengenalan kematangan pisang, hasil yang di dapatkan untuk metode GLCM dengan K-NN dengan arah 0o dan jarak 1 lebih banyak mengenali kematangan buah pisang dan akurasi tertinggi. Hasil yang di dapatkan dari pengujian metode akurasi yang diperoleh belum cukup baik, hasil dari perhitungan akurasi di atas bahwa semakin banyak nilai K semakin rendah nilai akurasi yang diperoleh, dan sebaliknya jumlah K 2 lebih banyak mengenali pengenalan kematangan pisang ketimbang jumlah K 3. Dari hasil perhitungan akurasi untuk pengujian model di dapatkan nilai sebesar 84%, dan tahap pengujian akurasi Klasifikasi hanya mendaptkan nilai sebesar 76%. Pada penelitian ini penggunaan metode K-NN kurang baik untuk tahap klasifikasi.

## Pembahasan Sistem

Untuk menjalankan sebuah program, maka kita harus menyelesaikan pembuatan program dan selanjutnya akan muncul tampilan menu program yang sedang dijalankan tersebut.



**Gambar 5.1** Tampilan Awal Aplikasi

Setelah selesai pembuatan program, maka akan muncul tampilan browse data training. Pada halaman tampilan proses load data training ini adalah tampilan jika pengguna menekan tombol browse untuk membuka data training yang ingin diuji.



**Gambar 5.2** Tampilan Proses Load Data Training

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



**Gambar 5.3** Tampilan Proses Load Data Testing

Setelah Pemilihan *Image Testing* Folder *Traning* dan data *Testing,* tampilan ini akan memunculkan nama folder yang telah dipilih pengguna.



# Gambar 5.4 Tampilan Nama Folder Yang Telah Dipilih

Pada Hasil Proses Pengenalan Kematangan, akan muncul tampilan saat pengguna mengklik/memilih tombol atau button *proses* setelah memilih gambar yang akan diuji.



**Gambar 5.5** Tampilan Hasil Pengenalan

# BAB VI

**PENUTUP**

## Kesimpulan

Berdasarkan hasil penelitian yang dilakukan pada tingkat kematangan pisang dan pembahasan yang telah diuraikan sebelumnya maka dapat ditarik suatu kesimpulan bahwa hasil penerapan K-NN dan GLCM yang diimplementasikan dalam proses pengenalan kematangan buah pisang mampu mendeteksi dengan nilai akurasi yang diukur dengan *confusion matrix* untuk pengujian klasifikasi 76%.

## Saran

Setelah melakukan penelitian pengenalan kematangan buah pisang, ada beberapa saran yang perlu di perhatikan untuk mencapai tujuan yang di harapkan, yaitu Penulis berharap untuk penelitian selanjutnya agar mendapatkan hasil akurasi yang lebih baik.

# DAFTAR PUSTAKA

[1] M. D. Y. Ambarita, E. S. Bayu, and H. Setiado, “Identifikasi Karakter Morfologis Pisang (Musa Spp.) Di Kabupaten Deli Serdang,” *J. Online Agroekoteknologi*, vol. 4, no. 1, pp. 1911–1924, 2016.

[2] Kementan, “Renstra Kementrian Pertanian Pertanian Tahun 2015 - 2019,” *Hari Aids Sedunia 2014*, pp. 1–339, 2014.

[3] R. U. Saputra, “Pengaruh Jenis Bonggol Dan Konsentrasi BA (BENZILADENIN)Terhadap Pertumbuhan Vegetatif Tanaman Pisang (Musa Paradisiaca Linn) Kepok Kuning,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[4] 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.

[5] Y. Whantdono, “Klasifikasi Buah Pisang Berdasarkan Warna Dan Bentuk Dengan Metode K-Nearest Neighbor Menggunakan Deteksi Tepi Canny Oleh : Yoggy Whantdono Dibimbing oleh : Universitas Nusantara PGRI Kediri Surat Pernyataan Artikel Skripsi Tahun 2017,” *Simki-Techsain*, vol. 01, no. 08, pp. 1–11, 2017.

[6] F. S. Ni’mah, T. Sutojo, and D. R. I. M. Setiadi, “Identifikasi Tumbuhan Obat Herbal Berdasarkan Citra Daun Menggunakan Algoritma Gray Level Co-occurence Matrix dan K-Nearest Neighbor,” *J. Teknol. dan Sist. Komput.*, vol. 6, no. 2, p. 51, 2018.

[7] I. A. Halela, B. Nurhadiyono, and F. Z. Rahmanti, “Identifikasi Jenis Buah Apel Menggunakan Algoritma K-Nearest Neighbor (KNN) dengan Ekstraksi Fitur Histogram,” pp. 1–8, 2016.

[8] G. Ramadhan, E. C. Djamal, and T. Darmanto, “Klasifikasi Identitas Wajah Untuk Otorisasi Menggunakan Deteksi Tepi dan LVQ,” *Semin. Nas. Apl. Teknol. Inf. 2016 Yogyakarta*, no. November, pp. 37–41, 2016.

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

[10] R. Anggraini, B. Hidayat, and S. Darana, “Klasifikasi Jenis Kualitas Keju Dengan Menggunakan Metode Gray Level Co-occurrence Matrix (GLCM) dan Support Vector Machine (SVM) Pada Citra Digital,” *e-Proceeding Eng.*, vol. 4, no. 2, pp. 2035–2042, 2017.

[11] D. D. Prihatin, B. Hidayat, and S. Saidah, “Deteksi Batik Bojonegoro Menggunakan Metode Gray Level Co-occurence Matrix (GLCM) Dan Naive Bayes Detection Of Batik Bojonegoro Using Gray Level Co-occurence Matrix (GLCM) And Naive Bayes Desi Dwi Prihatin [ 1 ], Dr . Ir . Bambang Hidayat , DEA .[,” vol. 5, no. 3, pp. 4650–4657, 2018.

[12] S. A. Wibowo, B. Hidayat, and U. Sunarya, “Simulasi dan Analisis Pengenalan Citra Daging Sapi dan Daging Babi dengan Metode GLCM dan KNN,” *Semin. Nas. Inov. Dan Apl. Teknol. Di Ind. 2016*, pp. 338–343, 2016.

[13] S. Mutro N, A. Izzah, A. Kurniawardhani, and M. Masrur, “Optimasi Teknik Klasifikasi Modified K- Nearest Neighbor Menggunakan Algoritma Genetika Optimization Techniques Modi ed k Nearest Neighbor Classi cation Using Genetic Algorithm,” *Gamma*, vol. 10, no. September, pp. 130–134, 2014.

[14] Kasrina and A. Zulaikha, “Pisang Buah (Musa Spp): Keragaman Dan Etnobotaninya Pada Masyarakat Di Desa Sri Kuncoro Kecamatan Pondok Kelapa Kabupaten Bengkulu Tengah,” *Pros. Semirata FMIPA Univ. Lampung*, vol. 1, no. 1995, pp. 33–40, 2013.

[15] A. Purno and W. Wibowo, “Implementasi Teknik Computer Vision Dengan Metode Colored Markers Trajectory Secara Real Time,” *J. Tek. Inform.*, vol. 8, no. 1, pp. 45–48, 2016.

[16] S. N. Afifah, “Image Processing,” vol. 166, no. 1, pp. 196–223, 2018.

[17] S. Hermawan and A. T. Handoko, “Pengertian – pengertian Dasar Image Processing,” pp. 1–7, 2013.

[18] N. Ahmad, “Pengolahan Citra Digitsl (DIGITAL IMAGE PROCESSING) Deskripsi Mata Kuliah Kompetensi Matakuliah,” 2013.

[19] C. Iswahyudi, “Implementasi Citra Digital Pada Berbagai Bidang,” 2013.

[20] I. Amalia, “Pengenalan Citra Tanda Tangan Menggunakan Gray Level Co-occurence Matrix (GLCM) Dan Probabilistic Neural Network (PNN).”

[21] D. S. Tobias, “Pada Citra Fundus Retina Dengan Metode K-Nearest Neighbor Glaucoma Detection in Retinal Fundus Image Using K- Nearest Neighbor Method Faculty of Science and Technology,” p. 72, 2016.

[22] A. Azwar, “Integrasi Ekstraksi Fitur Local Binary Pattern Dan Gray-Level Cooccurence Metrix Untuk Pengenalan Ekspresi Mulut Pembelajar,” *Ilk. J. Ilm.*, vol. 9, no. 1, p. 17, 2017.

[23] E. K. Sharma, E. Priyanka, E. A. Kalsh, and E. K. Saini, “GLCM and its Features,” *Int. J. Adv. Res. Electron. Commun. Eng.*, vol. 4, no. 8, pp. 2180–2182, 2015.

[24] A. Nurida, “Implementasi Algoritma K-Nearest Neighbor Dalam Sistem Case Based Reasoning Untuk Pembentukan Identitas Jawaban Otomatis Dan Pencari Kemiripan Jawaban Dari Soal-soal Algoritma,” 2016.

[25] I. Jaya, A. Hizriadi, and E. S. Purba, “Klasifikasi Surat Laporan Kehilangan Kepolisian Menggunakan Algoritma K – Nearest Neighbor,” *TECHSI - J. Tek. Inform.*, vol. 10, no. 2, p. 120, 2018.

[26] M. Paramita and R. Ely, “Prediksi Nilai Proyek Akhir Mahasiswa Menggunakan Algoritma Klasifikasi Data Mining,” *Semin. Nas. Sist. Inf. Indones. 2015*, vol. 11, no. November, pp. 1–7, 2015.

[27] M. A. Banjarsari, I. Budiman, and A. Farmadi, “Penerapan K-Optimal Pada Algoritma Knn Untuk Prediksi Kelulusan Tepat Waktu Mahasiswa Program Studi Ilmu Komputer Fmipa Unlam Berdasarkan Ip Sampai Dengan Semester 4,” *Klik - Kumpul. J. Ilmu Komput.*, vol. 2, no. 2, pp. 159–173, 2016.

[28] A. Saifudin, “Metode Data Mining Untuk Seleksi Calon Mahasiswa Pada Penerimaan Mahasiswa Baru Di Universitas Pamulang Aries,” vol. 10, no. 1, pp. 2–4, 2018.

[29] A. Hendini, “Pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro ZHEZHA PONTIANAK),” *KHATULISTIWA Inform.*, vol. 4, no. 2, pp. 107–116, 2016.

[30] Haviluddin, “Memahami Penggunaan UML (Unified Modelling Language) HAVILUDDIN,” vol. 6, no. 1, pp. 1–15, 2011.

[31] T. R. Perkasa, “Rancang Bangun Pendeteksi Gerak Menggunakan Metode Image Substraction Pada Single Board Computer (SBC),” *J. Control Netw. Syst.*, vol. 3, no. 2, pp. 90–97, 2014.

[32] H. H. Setiawan, “Klasifikasi Jenis Buah Pisang Dengan Image Processing Menggunakan Methode Classification of Type of Banana Fruits With Image Processing Using Backpropagation Method Faculty of Science and Technology,” 2018.

## *Lampiran 1 Coding*

**Main Coding (Python 3.7)**

1. **Koding Program**
2. Proses Pengolahan

def prapengolahan(self, image):

image = cv2.resize(image, (200, 200))

gbrGray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

gbrHist = cv2.equalizeHist(gbrGray)

ret, thresh = cv2.threshold(gbrHist,0,255, cv2.THRESH\_BINARY\_INV+cv2.THRESH\_OTSU)

im\_floodfill = thresh.copy()

h,w = thresh.shape[:2]

mask = np.zeros((h+2, w+2), np.uint8)

cv2.floodFill(im\_floodfill, mask, (0,0), 255);

im\_floodfill\_inv = cv2.bitwise\_not(im\_floodfill)

im\_out = thresh | im\_floodfill\_inv

kernel = np.ones((3,3), np.uint8)

opening = cv2.morphologyEx(im\_out,cv2.MORPH\_OPEN,kernel, iterations = 2)

sure\_fg = cv2.dilate(opening,kernel,iterations=1)

dist\_transform = cv2.distanceTransform (opening,cv2.DIST\_L2,0)

ret,sure\_fg = cv2.threshold (dist\_transform,0.2\*dist\_transform.max(),255,0)

images, contour, hier = cv2.findContours(im\_out, cv2.RETR\_LIST, cv2.CHAIN\_APPROX\_SIMPLE)

mx = (0,0,0,0)

mx\_area = 0

for cnt in contour :

x,y,w,h = cv2.boundingRect(cnt)

area = w\*h

if area > mx\_area:

mx = x,y,w,h

mx\_area = area

x,y,w,h = mx

roi = im\_out[y:y+h,x:x+w]

return gbrGray, gbrHist, roi

1. Proses Fitur GLCM

def glcmfitur(self, img, neigh, angle):

glcm = feature.greycomatrix(img,[neigh],[angle], symmetric=False, normed=True)

asm = feature.greycoprops(glcm,'ASM')[0,0]

con = feature.greycoprops(glcm,'contrast')[0,0]

ene = feature.greycoprops(glcm,'energy')[0,0]

hmg = feature.greycoprops(glcm,'homogeneity')[0,0]

cor = feature.greycoprops(glcm,'correlation')[0,0]

dis = feature.greycoprops(glcm, 'dissimilarity')[0,0]

return [asm, con, ene, hmg, cor, dis]

1. Proses Klasifikasi KNN

def proses\_klasifikasiKNN(self, k):

y = self.labeltrain()

x = self.buildtraindata()

xEncode = self.encoderData(x)

clf = KNeighborsClassifier(n\_neighbors=k, weights='uniform', algorithm='auto', metric='minkowski')

clf.fit(xEncode, y)

return clf

## *Lampiran 2 Riwayat Hidup Mahasiswa*

**RIWAYAT HIDUP MAHASISWA**



**Nama** : Amin Dwi Satrio

**NIM** : T3115144

**Tempat, Tanggal Lahir** : Gorontalo, 04 November 1997

**Agama** : Islam

**Email** : [amindwisatrio88@gmail.com](mailto:amindwisatrio88@gmail.com)

**Riwayat Pendidikan** :

1. Tahun 2009, menyelesaikan Pendidikan di Sekolah Dasar Negeri 24 Dungingi
2. Tahun 2012, menyelesaikan Pendidikan di Sekolah Menengah Pertama 8 Kota Gorontalo
3. Tahun 2015, menyelesaikan Pendidikan di Sekolah Menengah Kejuruan Negeri 1 Kota Gorontalo
4. Tahun 2015, telah diterima menjadi Mahasiswa di Universitas Ichsan Gorontalo

## *Lampiran 3 Surat Rekomendasi Penelitian*

**KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI**

**UNIVERSITAS ICHSAN GORONTALO**

**FAKULTAS ILMU KOMPUTER**

**SK MENDIKNAS NOMOR 84/D/O/2001**

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

**SURAT REKOMENDASI**

Hal : -

KepadaYth:

Ketua Prodi S1 Teknik Informatika

Di Tempat.

Dengan hormat,

Yang bertanda tangan di bawah ini :

Nim : T3115144

Nama : Amin Dwi Satrio

Jurusan : Teknik Informatika

Judul Skripsi : Pengenalan Kematangan Buah Pisang Menggunakan Metode Gray Level Co-occurrence Matrix Dan K–Nearest Neighbor

Benar telah melakukan penelitian, adapun Dataset Penelitian menggunakan data public dengan nama dataset yang di unduh dari <https://data.mendeley.com/datasets/zk3tkxndjw/>. Adapun Dataset terlampir di halaman berikutnya.

Demikian surat penelitian ini saya ajukan, atas perhatiannya diucapkan terima kasih.

Gorontalo, 26 November 2019

Amin Dwi Satrio  
T3115144

## *Lampiran 4 Surat Rekomendasi Bebas Plagiasi*

## *Lampiran 5 Hasil Cek Plagiasi Turnitin*