

**SEGMENTASI KEMATANGAN BUAH JERUK
BERDASARKAN KEMIRIPAN WARNA MENGGUNAKAN
ALGORITMA K-MEANS**

SKRIPSI

**ATIQA H AULIA
0701163098**



**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN
2021**

**SEGMENTASI KEMATANGAN BUAH JERUK
BERDASARKAN KEMIRIPAN WARNA MENGGUNAKAN
ALGORITMA K-MEANS**

SKRIPSI

Diajukan Untuk Memenuhi Syarat Mencapai Gelar Sarjana Komputer

**ATIQAHA AULIA
0701163098**



**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN
2021**

PERSETUJUAN SKRIPSI

Hal : Surat Persetujuan Skripsi

Lamp : -

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

Assalamu'alaikum Wr. Wb

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

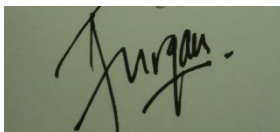
Nama : Atiqah Aulia
Nomor Induk Mahasiswa : 0701163098
Program Study : Ilmu Komputer
Judul : Segmentasi Kematangan Buah Jeruk Berdasarkan
Kemiripan Warna Menggunakan Algoritma
K-means

Dapat disetujui untuk segera diMunaqasyahkan. Atas perhatiannya kami ucapkan terimakasih.

Medan, 28 Januari 2021
14 Jumadil Awal 1442 H

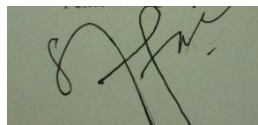
Komisi Pembimbing,

Pembimbing I,



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

Pembimbing II,



Sriani, S.Kom., M.Kom
NIB. 1100000108

SURAT PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertandatangan di bawah ini:

Nama : Atiqah Aulia
Nomor Induk Mahasiswa : 0701163098
Program Studi : Ilmu Komputer
Judul : Segmentasi Kematangan Buah Jeruk Berdasarkan
Kemiripan Warna Menggunakan Algoritma
K-means

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

Medan, 29 Januari 2021



Atiqah Aulia
NIM. 0701163098



KEMENTERIAN AGAMA REPUBLIK INDONESIA
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA MEDAN
FAKULTAS SAINS DAN TEKNOLOGI
Jl. IAIN No. 1 Medan 20235
Telp. (061) 6615683-6622925, Fax. (061) 6615683
Url: <http://saintek.uinsu.ac.id>, E-mail: saintek@uinsu.ac.id

PENGESAHAN SKRIPSI

Nomor: 016/ST/ST.V.I/PP.01.1/02/2021

Judul : Segmentasi Kematangan Buah Jeruk Berdasarkan
Kemiripan Warna Menggunakan Algoritma *K-means*
Nama : Atiqah Aulia
Nomor Induk Mahasiswa : 0701163098
Program Studi : Ilmu Komputer
Fakultas : Sains dan Teknologi

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

Pada hari/tanggal : Kamis, 28 Januari 2021
Tempat/media : Via Zoom Meeting

Tim Ujian Munaqasyah,
Ketua,

Ilka Xufria, M.Kom
NIP. 198506042015031006

Dewan Penguji,

Penguji I,

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

Penguji III,

Muhammad Ikhsan, ST., M.Kom
NIP. 198304152011011008

Penguji II,

Sriani, S.Kom, M.Kom
NIB. 1100000108

Penguji IV,

Armansyah, M.Kom
NIB. 1100000074

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

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

ABSTRAK

Kondisi kematangan buah jeruk dapat ditentukan dengan cara melihat beberapa parameter salah satunya dengan warna, pori-pori semakin besar serta kulit buah kuning merata. Identifikasi tingkat kematangan buah jeruk selama ini oleh petani dan konsumen dengan menggunakan cara manual, seperti pengamatan pada warna, pori-pori dan kulit buah jeruk. Identifikasi seperti itu akan membutuhkan waktu relatif lama dan menghasilkan kematangan yang beragam karena manusia memiliki keterbatasan visual dalam mengidentifikasi, tingkat kelelahan dan perbedaan pendapat tentang kematangan yang baik. Kekurangan dari metode secara manual diperlukan suatu teknologi yang mampu melakukan proses identifikasi kematangan secara objektif, konsisten dan hasil yang lebih jelas. Salah satunya adalah dengan proses segmentasi menggunakan algoritma *k-means*. Proses segmentasi bertujuan untuk membagi atau memisahkan citra ke dalam beberapa daerah (region) berdasarkan kemiripan atribut yang dimilikinya. Algoritma *k-means* akan mengclusterkan data yang memiliki karakteristik yang sama dikelompokkan ke dalam satu kelompok yang sama dan data yang mempunyai karakteristik yang berbeda dikelompokkan ke dalam kelompok yang lainnya. Dari hasil pengambilan gambar dari 6 sudut, yaitu depan, belakang, atas, bawah, dan kanan kiri dengan menggunakan dataset sebanyak 8 maka menghasilkan gambar sebanyak 48, dan dengan pengujian metode tersebut hasil pengclusteran kematangan buah jeruk menghasilkan kematangan sebanyak 6 dan mentah sebanyak 2.

Kata Kunci: Identifikasi, Algoritma *K-means*, Segmentasi, *Clustering*.

ABSTRACT

The condition of the ripeness of citrus fruit can be determined by looking at several parameters, one of which is the color, larger pores and evenly yellow fruit skin. Farmers and consumers alike identify the level of maturity of citrus fruits using manual methods, such as observing the color, pores and skin of citrus fruits. Such identification will take a relatively long time and produce a variety of maturity because humans have visual limitations in identifying, the level of fatigue and disagreements about good maturity. The weakness of the manual method requires a technology capable of carrying out the maturity identification process objectively, consistently and with clearer results. One of them is the segmentation process using the k-means algorithm. The segmentation process aims to divide or separate the image into several regions (regions) based on the similarity of the attributes it has. The k-means algorithm will group data that have the same characteristics into the same group and data that have different characteristics grouped into other groups. From the results of taking pictures from 6 angles, namely front, back, top, bottom, and right left using a dataset of 8, it produces 48 images, and by testing this method the results of clustering the ripeness of citrus fruit yields 6 ripeness and raw 2.

Keywords: Identification, Algoritme K-means, Segmentation, Clustering

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Syukur Alhamdulillah penulis sampaikan kepada Allah SWT yang telah memberikan kemampuan untuk menyelesaikan skripsi ini. Shalawat beriring salam atas junjungan nabi Muhammad SAW, semoga kita mendapatkan syafa'atnya kelak di kemudian hari, Amiiin.

Ini berjudul “Segmentasi Kematangan Buah Jeruk Berdasarkan Kemiripan Warna Menggunakan Algoritma K-Means”. Disusun dalam rangka memenuhi tugas-tugas dan melengkapi syarat-syarat untuk memperoleh gelar sarjana dalam Ilmu Komputer pada Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan.

Pada awalnya sungguh banyak hambatan yang penulis hadapi dalam penulisan skripsi ini. Namun berkat adanya pengarahan, bimbingan dan bantuan yang diterima akhirnya semuanya dapat diatasi dengan baik.

Pada kesempatan ini penulis menyampaikan banyak terima kasih kepada:

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

6. Bapak Rakhmat Kurniawan R, S.T., M.Kom selaku dosen pembimbing Akademik sekaligus Kepala Laboratorium Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan.
7. Seluruh tenaga pengajar dan pegawai Program Studi S1 Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan.
8. Teristimewa ucapan terima kasih kepada orang tua tercinta yaitu Arsyad Lubis dan Maslia Nasution yang telah memberikan bantuan moril maupun materil, semangat dan doa yang begitu besar kepada penulis.
9. Teman-teman kelas Ilmu Komputer 3 yang selalu memberikan dukungan serta arahan kepada penulis, penulis berharap kepada teman-teman ilmu komputer 3 pada masa pandemi ini tetap bersemangat dalam menjalankan aktivitas serta perjuangan untuk meraih gelar sarjananya.
10. Kepada abang, kakak, dan adik kandung penulis, Munawir Lubis, Nurwahidah Lubis, Namlis Lubis, Mikhwar Zeni Lubis, Yusril Mahendra Lubis, dan Amiril Mulia Lubis terima kasih untuk dukungan, doa dan semangat, serta bantuan baik moril maupun materil yang selalu diberikan kepada penulis.
11. Dan semua pihak yang telah membantu penulis namun tidak dapat disebutkan satu persatu.

Penulis telah berusaha dengan segala upaya yang penulis lakukan dalam penyelesaian skripsi ini. Penulis menyadari sepenuhnya bahwa dalam pembuatan skripsi ini masih jauh dari kesempurnaan, untuk itu kritik dan saran yang bersifat membangun sangat penulis harapkan dari pembaca demi sempurnanya skripsi ini. Semoga isi skripsi ini bermanfaat dalam memperkaya khasanah ilmu pengetahuan bagi pembaca. Amiiin Ya Rabbal'alamin.

Medan, 28 Januari 2021

Penulis,

Atiqah Aulia

DAFTAR ISI

	Halaman
ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	v
DAFTAR TABEL	vii
DAFTAR GAMBAR.....	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat penelitian	4
BAB II TINJAUAN PUSTAKA	5
2.1 Tanaman Jeruk.....	5
2.2 Citra	5
2.3 Jenis-jenis Citra	6
2.3.1 Citra Analog	6
2.3.2 Citra Digital	7
2.4 Format File Citra	12
2.4.1 JPEG	12
2.5 Pengolahan Citra.....	12
2.6 Segmentasi Citra.....	13
2.7 Algoritma <i>K-means</i>	13
2.8 Jarak <i>Euclidean</i>	15
2.9 Segmentasi Menggunakan <i>K-means Clustering</i>	15
2.10 <i>Flowchart</i>	15
2.11 Matlab	17
2.12 Penelitian Terkait.....	18

BAB III METODE PENELITIAN.....	20
3.1 Tempat dan Waktu Penelitian.....	20
3.1.1 Tempat Penelitian.....	20
3.1.2 Waktu Penelitian	20
3.2 Bahan dan Alat Penelitian	20
3.2.1 Perangkat Keras.....	21
3.2.2 Perangkat Lunak.....	21
3.3 Prosedur Kerja	21
3.3.1 Teknik Pengumpulan Data	21
3.3.2 Analisis Kebutuhan	22
3.3.3 Perancangan.....	23
3.3.4 Pengujian	24
3.3.5 Penerapan/Penggunaan.....	24
BAB IV PEMBAHASAN DAN HASIL.....	26
4.1 Pembahasan	26
4.1.1 Analisis Data	26
4.1.2 Representasi Data	28
4.1.3 Hasil Analisis.....	28
4.1.4 Perancangan Sistem.....	53
4.2 Hasil.....	59
4.2.1 Pengujian	59
4.2.2 Penerapan	68
BAB V KESIMPULAN DAN SARAN.....	70
5.1 Kesimpulan.....	70
5.2 Saran	70
DAFTAR PUSTAKA	71
LAMPIRAN-LAMPIRAN	

DAFTAR GAMBAR

Gambar	Judul Gambar	Halaman
2.1	Koordinat Citra Digital	7
2.2	Ilustrasi Citra Digital.....	7
2.3	Citra Biner.....	8
2.4	Citra <i>Grayscale</i>	9
2.5	Contoh Penyimpanan Citra Warna di Dalam Memori.....	10
3.1	Tahap-tahap Prosedur Kerja.....	21
3.2	Diagram Perencanaan Algoritma <i>K-means</i>	23
4.1	Citra Asli	28
4.2	Size 4x4	28
4.3	Citra <i>Grayscale</i> 4x4	29
4.4	Citra <i>Grayscale</i>	29
4.5	Citra hasil segmentasi dengan metode <i>threshold</i>	30
4.6	Citra <i>Grayscale</i> 4x4	30
4.7	Citra <i>Grayscale</i>	31
4.8	Citra hasil segmentasi dengan metode <i>threshold</i>	31
4.9	<i>Flowchart</i> Sistem	53
4.10	<i>Flowchart</i> Algoritma <i>K-means</i>	54
4.11	Perancangan Antar Muka.....	56
4.12	Tampilan Aplikasi.....	59
4.13	Tampilan Inputan Citra	60
4.14	Tampilan Input Citra 1	61
4.15	Tampilan Input Citra 2	61
4.16	Tampilan Input Citra 3	62
4.17	Tampilan Input Citra 4	63
4.18	Tampilan Input Citra 5	63
4.19	Tampilan Input Citra 6.....	64
4.20	Tampilan proses <i>k-means clustering</i>	65

DAFTAR TABEL

Tabel	Judul Tabel	Halaman
2.1	Simbol <i>Flowchart</i>	16
3.1	Waktu Penelitian	20
3.2	Daftar Pertanyaan.....	22
4.1	Dataset buah jeruk.....	26
4.2	Dataset.....	31
4.3	Jarak Ke <i>Cluster</i>	37
4.4	Jarak Ke <i>Cluster</i>	43
4.5	Jarak Ke <i>Cluster</i>	49
4.6	Hasil Pengclusteran.....	51
4.7	Hasil Pengujian	65

DAFTAR LAMPIRAN

Lampiran	Judul Lampiran
1.	Tools Matlab
2.	Listing Program
3.	Uji Coba Tingkat Keberhasilan
4.	Daftar Riwayat Hidup

BAB I

PENDAHULUAN

1.1.Latar Belakang

Tanaman jeruk merupakan salah satu produk tanaman yang mendapat perhatian khusus dari pemerintah Indonesia. Pemerintah Indonesia melalui Kementerian mengatakan bahwa tanaman jeruk merupakan tanaman tahunan yang sekitar 70-80% diciptakan di Indonesia dan secara konsisten mengalami kemajuan dalam perkembangannya termasuk luas lahan, jumlah penghasil bahkan minat pasar (Febrinanto, 2018).

Sebagaimana yang dijelaskan dalam surah Ash Shu'ara:7 yang berbunyi:

أَوَلَمْ يَرَوْا إِلَى الْأَرْضِ كَمْ أَنْبَتْنَا فِيهَا مِنْ كُلِّ زَوْجٍ كَرِيمٍ (v)

Artinya: “Dan apakah mereka tidak memusatkan perhatiannya pada bumi? Berapa jumlah kita di bumi ini berbagai macam tanaman yang baik?”.

Ayat tersebut berisi perintah kepada manusia untuk mengembangkan pandangannya sejauh kemampuannya, dengan fokus pada apa pun yang telah Allah SWT ciptakan di bumi mulai dari tanah, air, udara, cahaya, makhluk hingga hamparan tanaman hijau.

Proses atau lamanya waktu pemetikan buah jeruk cukup lama yaitu mulai dari berbunga sampai mengalami kematangan itu berkisar 9 bulan lamanya. Untuk mendapatkan hasil yang diinginkan petani harus tetap merawat dan memupuk tanaman jeruk secara rutin. Pemupukan yang dilakukan yaitu sekali dalam 6 bulan agar tanaman jeruk mengalami pertumbuhan sesuai yang diharapkan ataupun yang diinginkan (Boneran, 2020).

Kondisi kematangan buah jeruk dapat dilakukan dengan melihat beberapa parameter salah satunya dengan warna, pori-pori semakin besar serta kulit buah kuning merata. Kematangan jeruk akan terlihat dari warnanya yang berwarna kuning akan menghasilkan rasa yang manis, sedangkan buah jeruk yang mentah berwarna hijau akan menghasilkan rasa asam pekat. Warna, pori-pori, serta kulit

buah jeruk dapat digunakan untuk mengidentifikasi buah tersebut masih mentah dan matang (Boneran, 2020).

Identifikasi tingkat kematangan buah jeruk selama ini oleh petani dan konsumen dengan menggunakan cara manual, seperti pengamatan pada warna, pori-pori dan kulit buah jeruk. Identifikasi seperti ini akan membutuhkan waktu relatif lama dan menghasilkan kematangan yang beragam karena manusia memiliki keterbatasan visual dalam mengidentifikasi, tingkat kelelahan dan perbedaan pendapat tentang kematangan yang baik. Konsumen biasanya melihat dari bentuk luarnya saja seperti warna buah jeruk yang mencolok dan ukuran yang besar namun tidak dapat membedakan bagaimana tingkat kematangan dan gizi yang baik pada buah yang baik untuk dikonsumsi. Oleh karena itu, diperlukan suatu teknologi yang dapat membantu proses segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means* menggunakan matlab.

K-Means merupakan salah satu teknik yang digunakan dalam segmentasi citra berbasis *clustering*. Algoritme *K-Means* termasuk *partitioning clustering* yang memisahkan data ke *k* daerah bagian yang terpisah. Algoritma *K-Means* sangat terkenal karena kemudahan dan kemampuannya untuk mengcluster data besar dan data *outlier* dengan sangat cepat (Kusuma dan Ellyana, 2018).

Seperti Penelitian yang dilakukan oleh Kusuma dan Ellyana (2018) dalam judul penelitiannya yaitu Penerapan Citra terkompresi pada Segmentasi Citra Menggunakan Algoritma *K-Means*. Dalam penelitiannya citra yang diolah adalah citra MRI otak dan citra payudara. Sebelum citra-citra tersebut disegmentasi sebelumnya dikompresikan terlebih dahulu. Pada penelitiannya ini menghasilkan kesimpulan bahwa jenis-jenis ekstensi citra yang menjadi input aplikasi *K-Means*. Dimana citra yang berekstensi BMP yang memakan waktu lama dalam proses segmentasi. Sedangkan citra yang berekstensi JPG yang paling cepat.

Berdasarkan uraian diatas, maka penulis memutuskan untuk mengambil judul **“Segmentasi Kematangan Buah Jeruk Berdasarkan Kemiripan Warna Menggunakan Algoritma *K-means*”**.

1.2. Rumusan Masalah

Rumusan masalah dalam penelitian ini yaitu sebagai berikut:

1. Bagaimana menganalisis tingkat kematangan buah jeruk berdasarkan kemiripan warna dengan menggunakan proses segmentasi citra.
2. Bagaimana mengidentifikasikan kematangan buah jeruk berdasarkan kemiripan warna dengan menerapkan algoritma *K-means*.
3. Bagaimana menghasilkan sistem yang dapat melakukan segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *K-means*.

1.3. Batasan Masalah

Berdasarkan rumusan masalah diatas untuk segmentasi tingkat kematangan buah jeruk berdasarkan kemiripan warna maka yang menjadi batasan masalahnya sebagai berikut:

1. Penelitian ini hanya mendeteksi kematangan buah jeruk berdasarkan kemiripan warnanya.
2. Algoritma yang digunakan adalah *K-means*.
3. Buah jeruk yang digunakan adalah jenis jeruk madu.
4. Data buah jeruk sebanyak 8 sample jeruk.
5. Format citra yang digunakan *.jpg dan dimensi 124 x 165.

1.4. Tujuan Penelitian

Tujuan penelitian ini yaitu sebagai berikut:

1. Menerapkan proses segmentasi citra dalam menganalisis tingkat kematangan buah jeruk berdasarkan kemiripan warna.
2. Mengetahui identifikasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *K-means*.
3. membuat sebuah sistem yang bisa menghasilkan proses segmentasi kematangan buah jeruk berdasarkan kemiripan warna dengan menggunakan algoritma *K-means*.

1.5. Manfaat Penelitian

Manfaat penelitian ini yaitu sebagai berikut:

1. Sebagai bahan pengujian bagi pelajar maupun mahasiswa dalam menentukan kematangan buah jeruk.
2. Mengurangi adanya resiko buah jeruk yang masih mentah.
3. Penelitian ini diharapkan menghasilkan akurasi yang tinggi terhadap pengenalan kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means* yang didapatkan dari citra buah jeruk.

BAB II

TINJAUAN PUSTAKA

2.1. Tanaman Jeruk

Tanaman jeruk adalah tanaman produk alami tahunan yang berasal dari Asia. Cina dipercaya sebagai tempat jeruk pertama kali dikembangkan. Sejak bertahun-tahun sebelumnya, jeruk sudah tumbuh di Indonesia baik secara normal atau berkembang. Tanaman jeruk yang ada di Indonesia merupakan tradisi orang Belanda, yaitu jeruk manis dan keprok dari Amerika dan Itali.

Jenis jeruk lokal yang dikembangkan di Indonesia adalah jeruk Keprok (*Citrus reticulata/nobilis L.*), jeruk Siem (*C. microcarpa L.* dan *C. sinensis L.*) yang terdiri dari Siem Pontianak, Siem Garut, Siem Lumajang, jeruk manis (*C. auranticum L.* dan *C. sinensis L.*), jeruk sitrun/lemon (*C. medica*), jeruk besar (*C. maxima Herr.*) yang terdiri dari jeruk Nambangan-Madium dan jeruk Bali. Jeruk untuk bumbu masak terdiri dari jeruk nipis (*C. aurantifolia*), jeruk Purut (*C. hystrix*) dan jeruk sambal (*C. hystrix ABC*).

Penyajian jenis jeruk yang umumnya ditanam adalah jenis Lemon dan Grapefruit. Sedangkan pada lingkungan adalah jenis jeruk siem, jeruk anak, keprok medan, jeruk bali, jeruk nipis dan jeruk purut (Pontianak et al, 1952).

2.2. Citra

Dalam arti sebenarnya, citra (*image*) adalah gambar pada bidang dwimatra (duadimensi). Ditinjau dari sudut pandang matematis, citra adalah fungsi menerus (*continue*) dari intensitas cahaya pada bidang dua dimensi. Sumber cahaya menerangi objek yang memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh gadget, seperti mata manusia, kamera, pemindai (*scanner*), dan lain-lainya, sehingga gambar benda tersebut terekam (Munir, 2004:2).

Citra sebagai hasil dari suatu sistem perekaman data dapat bersifat :

1. optik berupa foto,
2. analog berupa sinyal video seperti gambar pada monitor televisi,

3. digital yang dapat langsung disimpan pada suatu pita magnetik.

Citra dibagi menjadi 2 yaitu citra sebenarnya (*still image*) dan citra bergerak (*moving image*). Citra sebenarnya (*still image*) adalah citra tunggal yang tidak bergerak. Citra bergerak (*moving images*) merupakan perkembangan dari citra sebenarnya yang ditampilkan secara berurutan sehingga memberi kesan pada mata kita sebagai gambar yang bergerak. Setiap citra di dalam rangkaian itu disebut *frame*. Gambar-gambar yang muncul di film layar lebar atau televisi pada dasarnya terdiri dari ratusan hingga ribuan frame.

Dalam jurnal yang berbeda, definisi citra adalah suatu representasi, kemiripan, atau imitasi dari suatu obyek atau benda. Sebuah citra berisi informasi tentang obyek yang sedang dibahas. Citra dapat dilihat sebagai gambar yang nyata dan gambar yang tidak terlihat. Agar terlihat oleh mata manusia, gambar yang tidak terdeteksi harus diubah menjadi gambar yang nyata, misalnya dengan menampilkannya di monitor, dicetak di kertas dan lain-lain. Salah satu gambar yang tidak terdeteksi adalah gambar yang terkomputerisasi. Citra juga dapat diartikan sebagai gambar dua dimensi yang disampaikan dari gambar analog dua dimensi yang konstan menjadi gambar diskrit melalui siklus pengujian. Gambar analog dibagi menjadi N baris dan M kolom bagian sehingga menjadi gambar diskrit. Persilangan antara baris dan kolom tertentu disebut dengan *piksel* (Sriani dkk, 2017).

2.3. Jenis-Jenis Citra

2.3.1. Citra Analog

Citra analog adalah citra yang bersifat kontinu, seperti gambar pada monitor televisi, foto sinar-X, foto yang tercetak di kertas foto, lukisan, pemandangan, hasil CT scan, gambar-gambar yang terekam pada pita kaset, dan lain-lain sebagainya. Citra analog tidak dapat ditangani di komputer sehingga tidak bisa langsung ditampilkan di komputer. Oleh karena itu, agar citra ini dapat diproses di komputer, langkah perubahan analog ke digital harus diselesaikan terlebih dahulu. Citra analog disampaikan dari gadget analog, video kamera

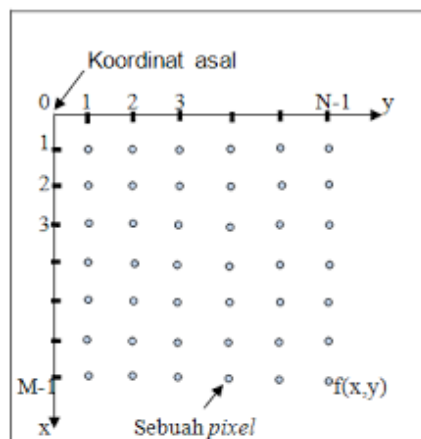
analog, kamera foto analog, *Web Cam*, *CT scan*, *sensor ultrasound* pada *system* *USG*, dan lain-lain.

2.3.2. Citra Digital

Citra Digital adalah klaster (pameran) yang berisi kualitas asli atau kompleks yang ditangani oleh pengelompokan potongan tertentu. Gambar terkomputerisasi dicirikan sebagai kapasitas $f(x,y)$ dengan ukuran garis M dan segmen N , di mana x dan y adalah arah spasial, dan kelimpahan f pada arah (x,y) dikenal sebagai tingkat daya atau redup dari gambar saat itu. Jika sisi atas dari x , y , dan nilai kelimpahan dari f terbatas dan diskrit, dapat dikatakan bahwa gambar tersebut adalah gambar yang terkomputerisasi (Putra, 2010:19).

Pengolahan Citra digital adalah menangani gambar menggunakan PC agar gambar dapat diuraikan secara efektif oleh orang atau mesin. Citra digital merupakan Gambar lanjutan adalah gambar yang dapat ditangani secara langsung di PC, isi gambar yang terkomputerisasi adalah nilai kompleks yang ditangani dengan menggunakan deretan bit tertentu. (Furqan dkk, 2020).

Gambar 2.1 menunjukkan posisi koordinat citra digital. Gambar 2.2 menunjukkan ilustrasi dari citra digital.



Gambar 2.1 Koordinat citra digital



Gambar 2.2 Ilustrasi citra digital

Sumber: (Putra, 2010).

1. Jenis-jenis Citra Digital

Nilai sebuah piksel memiliki nilai dalam jangkauan tertentu, dari nilai dasar hingga nilai terbesar. Jangkauan yang digunakan berbeda bergantung pada jenis warnanya. Namun, secara keseluruhan jangkauannya adalah $0 - 255$.

Gambar dengan penggambaran semacam ini adalah citra integer. Berikut adalah jenis gambar yang bergantung pada nilai pikselnya (Putra, 2010:39).

a. Citra Biner (Monokrom)

Citra biner adalah citra digital yang hanya memiliki dua nilai piksel potensial, untuk lebih spesifiknya sangat kontras. Citra biner juga disebut sebagai gambar B&W (kontras tinggi) atau gambar monokrom. Hanya perlu 1 bit untuk mengatasi nilai setiap piksel dari citra biner (Putra, 2010:40). Citra biner secara teratur muncul karena penanganan siklus seperti segmentasi, pengembangan, morfologi, dan dithering. Gambar 2.3 menunjukkan contoh citra biner.



2.3 Gambar Citra Biner

Sumber: (Putra, 2010).

b. Citra *Grayscale* (Skala Keabuan)

Citra *Grayscale* adalah gambar terkomputerisasi yang hanya memiliki satu saluran insentif untuk setiap piksel, seperti bagian Merah = Hijau = Biru. Nilai ini digunakan untuk menunjukkan tingkat intensitas. warna yang mereka miliki adalah hitam, abu-abu, dan putih. Tingkat hitam adalah warna abu-abu dengan tingkat dari hitam hingga mendekati putih. Citra grayscale yang menyertainya memiliki kedalaman 8 buah (256 campuran warna abu-abu) (Putra, 2010:40).



Gambar 2.4 Citra *Grayscale*

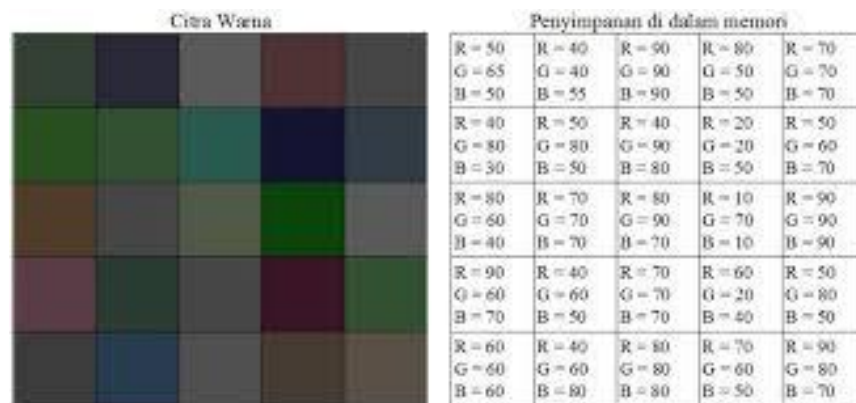
Sumber: (Putra, 2010).

c. Citra Warna (*True Color*)

Sebuah gambar dapat dicirikan sebagai kapasitas dua dimensi $f(x,y)$ di mana x dan y adalah fasilitas ruang dan amplitudo $f(x,y)$ dalam beberapa arah dikenal sebagai intensitas citra. Istilah tingkat keabuan biasanya digunakan dalam gambar monokrom. Citra warna dibingkai oleh campuran gambar individu. Misalnya, dalam kerangka warna RGB, citra warna terdiri dari tiga gambar monokrom yang mengandung intensitas warna R (Merah), G (Hijau), dan B (Biru).

Setiap piksel dalam citra warna menunjukkan warna yang merupakan perpaduan dari tiga warna dasar (Merah, Hijau, Biru). Setiap warna dasar menggunakan 8 buah = 1 byte kapasitas, yang berarti bahwa setiap warna memiliki derajat 255 warna. Ini menyiratkan bahwa setiap piksel memiliki campuran warna lebih dari 16 juta warna. Itulah sebabnya format ini disebut *true color* karena memiliki jumlah warna yang cukup banyak sehingga dapat dikatakan mencakup hampir semua warna di alam..

Penyimpanan citra *true color* dalam memori tidak sama dengan gambar skala *grayscale*. Setiap piksel dari gambar skala abu-abu 256 derajat diwakili oleh 1 byte. Sedangkan 1 piksel gambar alam asli diwakili oleh 3 byte, dimana setiap byte alamat (Merah), (Hijau), dan (Biru).. Gambar 3 menunjukkan contoh penyimpanan citra warna di dalam memori.



Gambar 2.5 Contoh penyimpanan citra warna di dalam memori

Sumber: (Angga dkk, 2018).

2. Elemen-Elemen Citra Digital

Citra digital mengandung berbagai elemendasar. Elemen-elemen dasar ini dikendalikan dalam penanganan citra dan selanjutnya dimanfaatkan dalam visi komputer. Elemen-elemen dasar yang penting diantaranya adalah (Munir, 2004:25).

a. Kecerahan (*Brightness*)

Kecerahan adalah satu kata lain untuk intensitas cahaya. Seperti dijelaskan di segmen awal, kecerahan pada suatu titik (piksel) pada gambar bukanlah intensitas asli, tetapi sebenarnya intensitas normal dari area sekitarnya. Kerangka visual manusia dapat mengubah dirinya sendiri ke tingkat kecerahan mulai dari yang paling rendah hingga yang paling tinggi dengan cakupan 1010.

b. Kontras (*contrast*)

Kontras menyatakan sebaran terang (*lightness*) dan gelap (*darkness*) di dalam sebuah gambar. Gambar kontras rendah digambarkan oleh sebagian besar komposisi citranya menjadi terang atau umumnya gelap. Dalam gambar dengan perbedaan besar, komposisi gelap dan terang diedarkan secara merata.

c. Kontur (*contour*)

Kontur adalah keadaan yang disebabkan oleh perubahan daya pada piksel yang berdampingan. Karena perubahan kontur adalah kondisi yang disebabkan oleh perubahan intensitas piksel yang berdampingan. Mengingat penyesuaian

intensitas ini, mata kita dapat mengidentifikasi tepi item dalam gambar. Pada gaya inilah cahaya membedakan tepi benda-benda dalam gambar.

d. Warna (*color*)

Warna adalah wawasan yang dirasakan oleh pengaturan visual manusia dari frekuensi cahaya yang dipantulkan oleh objek. Setiap warna memiliki frekuensi alternatif (λ) yang berbeda. Merah memiliki frekuensi paling penting, sedangkan ungu memiliki frekuensi paling sedikit.

warna-warna yang didapat oleh mata (kerangka visual manusia) adalah efek lanjutan dari campuran cahaya dengan berbagai frekuensi. Hasil penelitian menunjukkan bahwa paduan warna yang memberikan rentang warna paling luas adalah merah (R), hijau (G), dan biru (B)..

Pandangan kerangka visual manusia tentang shading sangat relatif karena dipengaruhi oleh berbagai standar, salah satunya disebabkan oleh transformasi yang menyebabkan mutilasi. Misalnya, bercak-bercak abu-abu di sekitar warna hijau akan tampak keunguan (mutilasi ruang), atau sebaliknya jika mata melihat hijau dan segera melihat abu-abu, mata melihat warna gelap sebagai warna ungu (mutilasi terhadap waktu).

e. Bentuk (*Shape*)

Bentuk adalah properti bawaan dari item tiga dimensi, dengan kesepakatan bahwa bentuk adalah properti yang sangat melekat untuk kerangka visual manusia.. Orang lebih sering memasang objek dengan bentuknya daripada komponen yang berbeda (shading misalnya). Biasanya, citra yang dibentuk oleh mata adalah citra dua dimensi (2 ukuran), sedangkan benda yang dilihat umumnya tiga dimensi (3 dimensi). informasi bentuk objek dapat diekstraksi dari citra ke awal pra-pengolahan dan segmentasi citra. Salah satu tantangan utama dalam visi komputer adalah mengatasi bentuk, atau bagian penting dari bentuk.

f. Tekstur (*Texture*)

Tekstur digambarkan sebagai sirkulasi spasial tingkat keabuan di dalam sekelompok piksel yang berdampingan. Dengan demikian, tekstur tidak dapat dicirikan untuk piksel. Kerangka visual manusia pada dasarnya tidak mendapatkan data citra secara bebas dari setiap piksel, kecuali citra dianggap

sebagai satu kesatuan. Tujuan di balik citra yang sedang dipertimbangkan ditentukan oleh skala di mana permukaannya terlihat. Misalnya, jika kita melihat gambar lantai keramik dari cara yang baik, kita melihat bahwa permukaan dibentuk oleh posisi ubin secara keseluruhan, bukan dari tampilan contoh di dalam ubin yang sebenarnya. Namun, jika kita melihat gambar yang sama dari jarak dekat, beberapa ubin terlihat di bidang pandang, sehingga kita melihat bahwa permukaan dibingkai oleh posisi contoh tertentu yang membentuk setiap ubin.

2.4. Format File Citra

Format file citra standar yang digunakan saat ini terdiri dari beberapa jenis. Format-format ini digunakan untuk menyimpan citra dalam file. Setiap format memiliki kualitasnya sendiri. Berikut adalah penjelasan format umum yang digunakan saat ini (Putra, 2010:58).

2.4.1. JPEG (.jpg)

(.jpg) adalah format yang sangat umum digunakan saat ini terutama untuk transmisi citra. format ini digunakan untuk menyimpan citra hasil menggunakan strategi JFEG.

2.5. Pengolahan Citra

Pengolahan citra adalah suatu karya yang dibuat untuk mengubah citra menjadi citra lain yang lebih menakjubkan sehingga hasil selanjutnya sesuai keinginan (Furqan dkk, 2020).

Pengolahan citra bertujuan untuk lebih mengembangkan kualitas gambar dengan tujuan agar dapat diuraikan secara efektif oleh orang atau mesin (untuk hal ini komputer). Teknik-teknik pengolahan citra mengubah citra menjadi citra yang berbeda. Jadi, masukannya adalah citra dan hasilnya juga citra, namun citra yang dihasilkan memiliki kualitas yang lebih baik daripada citra masukan. Selain itu yang diingat untuk bidang ini adalah tekanan citra (image compression) (Munir, 2004:5). Teknik pengolahan citra juga dapat dilakukan dengan mengidentifikasi

tekstur pada suatu objek dengan menggunakan metode statistik secara jelas (Furqan dkk, 2020).

2.6. Segmentasi citra

segmentasi adalah teknik yang memiliki arti memiliki pilihan untuk mengisolasi gambar menjadi beberapa lokal dan setiap wilayah memiliki karakteristik yang sama (Furqan dkk, 2020).

Teknik untuk membagi atau memisahkan suatu gambaran menjadi beberapa region (wilayah) dengan melihat kesamaan atribut-nya disebut segmentasi. segmentasi juga dianggap sebagai siklus yang memisahkan gambar menjadi berbagai bagian atau objek. segmentasi tentu saja bukan interaksi tunggal yang diselesaikan dalam pengolahan citradigital. Meskipun demikian, segmentasi adalah siklus penting dalam pengolahan citra digital. Dalam interaksi segmentasi, objek yang berubah menjadi tujuan akan diambil untuk sistem berikutnya. Ada dua kualitas nilai derajat citra dalam teknik pembagian ini, khususnya *discontinuity* dan *similarity*. Pada *discontinuity*, citra diisolasi tergantung pada perubahan mencolok dalam tingkat kecerahannya, biasanya diterapkan pada penemuan tepi, garis, daerah dan sisi gambar. Sementara *similarity* citra akan diisolasi tergantung pada *thresholding*, pengembangan area dan pemisahan dan penggabungan lokal, umumnya diterapkan pada citra statis dan dinamis. (M. Taufik, 2019).

2.7. Algoritma K-Means

K-Means adalah teknik pengelompokan data *non-hierarki* yang berusaha untuk mengelompokkan data yang ada menjadi satu kelompok atau sekumpulan sehingga data dengan kualitas yang sebanding dikumpulkan ke dalam kelompok yang sama dan data dengan berbagai atribut dikumpulkan ke dalam kelompok yang lain.

K-Means adalah metode pengelompokan berbasis jarak yang membagi data ke dalam berbagai kelompok dan perhitungan ini hanya berurusan dengan *atribut numerik*. Algoritma *K-Means* menggabungkan *partitioning clustering* yang

memisahkan data ke dalam k subwilayah yang terpisah. Algoritma *K-Means* sangat terkenal karena kemudahannya dan kapasitasnya untuk mengumpulkan data besar dan data *outlier* dengan cepat. Dalam algoritma *K-Means*, setiap data harus termasuk ke *cluster* tertentu dan bisa dimungkinkan bagi setiap data yang termasuk *cluster* tertentu pada suatu tahapan proses, pada tahapan berikutnya berpindah ke *cluster* lainnya.

Pada dasarnya, pemanfaatan algoritma dalam sistem *clustering* bergantung pada data yang ada dan tujuan yang akan dicapai. Oleh karena itu, algoritma *K-Means* digunakan untuk membuat prinsip-prinsip sebagai berikut:

1. Jumlah *Cluster* perlu diinputkan.
2. Hanya memiliki atribut bertipe *numeric*.

Algoritma *K-Means* merupakan metode *non-hierarki* yang pada awalnya mengambil sebagian dari bagian populasi untuk dimanfaatkan sebagai komunitas *cluster* yang mendasarinya. Pada tahap ini pusat *cluster* diambil secara sembarangan dari sekumpulan populasi data. Selanjutnya *K-Means* menguji setiap bagian dalam populasi data dan memasukkan bagian tersebut ke salah satu komunitas *cluster* yang telah didefinisikan berdasarkan jarak minimum antar bagian dan setiap *cluster*. Posisi pusat *cluster* akan dihitung ulang sampai semua bagian data diatur ke dalam setiap tempat *cluster* dan terakhir akan dibentuk posisi pusat *cluster* yang baru.

Algoritma *K-Means* pada dasarnya melakukan dua proses, khususnya metode yang terlibat dengan mengidentifikasi area fokus setiap kelompok dan cara paling umum untuk mencari individu dari setiap kelompok. (Darmi dan Setiawan, 2016). Cara kerja algoritma *K-Means*:

1. Tentukan k sebagai jumlah *cluster* yang ingin dibentuk.
2. Bangkitkan k *centroid* (titik pusat *cluster*) awal secara random.
3. Hitung jarak setiap data ke masing-masing *centroid*.
4. Setiap data memilih *centroid* yang terdekat.
5. Tentukan posisi *centroid* yang baru dengan cara menghitung nilai rata-rata dari data-data yang terletak pada *centroid* yang sama.

6. Kembali ke langkah-3 jika posisi *centroid* baru dengan *centroid* yang lama tidak sama.

2.8. Jarak Euclidean (Euclidean Distance)

Untuk mengukur jarak antara data dan titik fokus *cluster* digunakan jarak Euclidean, maka pada titik tersebut akan diperoleh matriks jarak sebagai berikut:

Rumus euclidian distance :

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

X = Pusat cluster

Y = data

2.9. Segmentasi Menggunakan K-Means clustering

Dalam proses segmentasi menggunakan K-Means, langkah awal adalah menentukan jumlah *cluster* pada citra yang telah ditangani sebelumnya dan menghitung jumlah *centroid* secara acak. Kemudian, pada saat itu, pastikan jarak piksel ke *centeroid* dan kumpulkan piksel-piksel yang bergantung pada jarak terdekat. Setelah nilai piksel gambar dikumpulkan berdasarkan jarak terdekatnya, pusat dihitung ulang sebagai pusat massa lain dengan menghitung piksel normal per *cluster* sebagai *centeroid* lain dan menyatukan piksel-piksel sesuai *centeroid*. Jika masih ada piksel yang berpindah *cluster*, maka dilakukan perhitungan *centeroid* baru, namun jika tidak ada nilai piksel yang berpindah *cluster*, sistem pengelompokan selesai. (Febrinanto dkk, 2018).

2.10. Flowchart

Flowchart adalah penyajian sistematis tentang proses dan logika yang dari penanganan informasi atau penggambaran grafis dari langkah-langkah dan pengaturan teknik suatu program.. *Flowchart* membantu para ahli dan pengembang untuk memisahkan masalah menjadi fragmen yang lebih sederhana

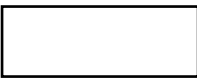
dan membantu dalam membedah opsi yang berbeda dalam pengoperasian. (Anharku, 2009).

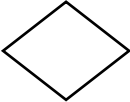
Aturan pembuatan *flowchart* Jika seorang ahli dan *software engineer* akan membuat *flowchart*, ada beberapa aturan yang harus diperhatikan, misalnya,:

1. *Flowchart* digambarkan dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari aktivitas harus diuraikan dengan menggunakan deskripsi kata kerja, misalkan Melakukan penggandaan diri.
5. Setiap langkah dari aktivitas harus berada pada urutan yang benar.
6. Lingkup dan *range* dari aktifitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada *flowchart* yang sama. Simbol konektor harus digunakan dan percabangannya diletakan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.
7. Gunakan simbol-simbol *flowchart* yang standar.

Simbol-simbol *flowchart* yang biasanya digunakan adalah simbol *flowchart* standar yang diberikan oleh ANSI dan ISO. Tabel 2.1 adalah bagan alir dari beberapa gambar *flowchart* yang digunakan untuk menggambarkan *flowchart* :

Tabel 2.1 Simbol *flowchart*

Simbol	Nama	Fungsi
	<i>TERMINATOR</i>	Permulaan/ akhir program
	GARIS ALIR (<i>FLOW LINE</i>)	Arah aliran program
	<i>PREPARATION</i>	Proses inisialisasi/ pemberian harga awal
	PROSES	Proses perhitungan/ proses pengolahan data

	<i>INPUT/OUTPUT DATA</i>	Proses input/ output data, parameter, informasi
	<i>PREDEFINED PROCESS (SUB PROGRAM)</i>	Permulaan sub program/ proses menjalankan subprogram
	<i>DECISION</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	<i>ON PAGE CONNECTOR</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman
	<i>OFF PAGE CONNECTOR</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda

Sumber: (Anharku, 2009).

2.11. Matlab

Matlab menggunakan konsep *array*/matriks sebagai variabel komponen standar tanpa persyaratan untuk pendeklarasian *array* seperti dalam bahasa yang lainnya. Selain itu juga dapat dikoordinasikan dengan aplikasi luar dan bahasa pemrograman seperti *C*, *Java*, *.NET* dan *Microsoft Excel*. Pemrograman Matlab memiliki berbagai aplikasi, terutama pada aplikasi yang membutuhkan perhitungan secara numerik. Sadarilah bahwa Matlab meakukan semua perhitungan numerik dalam bentk matriks.

Semua aktivitas numerik di Matlab adalah operasi matriks. Matlab dapat menampilkan hasil komputasi dalam bentuk grafik dan dapat direncanakan sesuai keinginan kita dengan memanfaatkan GUI yang kita buat sendiri (Hutagalung, 2018).

2.12. Penelitian Terkait

Preeti Panwar, Girdhar Gopal, dan Rakesh Kumar membandingkan pembagian berdasarkan bayangan dan perhitungan clustering dan *thresholding K-Means*. Metode *K-Means* menggunakan *cluster* partisi. Algoritma *K-Means clustering* digunakan untuk membagi suatu citra menjadi *k cluster*. *K-Means clustering dan thresholding* digunakan dalam ulasan ini untuk pemeriksaan hasil segmentasi. Korelasi kedua strategi ini tergantung pada batas divisi seperti MSE, PSNR dan NSR. MSE dan PSNR secara luas digunakan untuk mengukur tingkat liku-liku citra karena mereka dapat mengatasi tingkat umum kesalahan tingkat *grai-level* yang terkandung dalam keseluruhan citra.

Dengan menggunakan dua teknik ini, kinerja untuk berbagai citra dibagi menggunakan batasan seperti MSE, PSNR, dan SNR. Dari persepsi yang ditampilkan dalam ulasan ini, dapat diduga bahwa teknik *thresholding* memberikan hasil dari dua bagian. Namun, dalam metode *K-Means*, hasil porsi yang berbeda berhubungan dengan ukuran *cluster*. Kinerja meningkat dengan ukuran *cluster*. Lebih banyak ukuran *cluster* adalah tingkat akurasi. MSE dan PSNR digunakan untuk mengukur kualitas rekreasi. Nilai PSNR keempat citra dalam *K-Means clustering* lebih tinggi dari *thresholding* dan nilai MSE lebih rendah (Angga, 2018).

Review yang berjudul Pengenalan Tingkat Kenaikan Paya Rabo Pepaya Menggunakan Pengolahan Citra Berbasis Warna RGB Dengan *K-Means Clustering*. Dalam tinjauan ini teknik pengolahan citra dilakukan dengan memanfaatkan kualitas merah, hijau, biru sebagai sumber perspektif untuk menganalisa kondisi kematangan pepaya.

penentuanklasifikasi denganmetode *K-Means Clustering* yang memanfaatkan perbedaan jarak Euclidian sebagai sumber perspektif. Hasil yang diperoleh adalah 60% kelompok pepaya muda dipersepsikan sebagai pepaya muda, 90% kelompok pepaya mentah dipersepsikan pepaya mentah, sedangkan 100% kelompok pepaya masak penuh dibedakan sebagai pepaya matang penuh. (Eliyani dkk, 2013).

Seperti *review* oleh Andika dan Hafiz pada tahun 2018 memanfaatkan segmentasi citra memanfaatkan *K-Means* dan *Fuzzy C-Means*. Dalam *review* ini, kami membandingkan dua metode. Menurut hasil *review* oleh Andika dan Hafiz, metode *k-means* lebih unggul daripada metode *Fuzzy C-means*. (Mira & Taufik, 2019).

Pengujian yang dilakukan oleh Kusuma dan Ellyana pada tahun 2018 dengan judul penelitiannya yaitu Penerapan Citra terkompresi pada Segmentasi Citra Menggunakan Algoritma K-Means. citra yang ditangani adalah citra MRI otak dan citra payudara. Sebelum citra di segmetasi, dikompresi dulu. Pengujian ini mengasumsikan bahwa jenis ekstensi citra yang diinput untuk aplikasi *K-Means* adalah di mana citra dengan ekspansi BMP menghabiskan sebagian besar hari dalam interaksi divisi. Sementara citra dengan ekspansi JPG adalah yang tercepat (Orisa & Hidayat, 2019).

Hasil penelitian menggunakan algoritma *K-Means* untuk pembelahan sel darah putih, titik fokus pembelahan adalah pada pelepasan inti dan sitoplasma sel. Penggunaan *substrat* citra untuk menghasilkan bagian yang ideal setelah siklus segmentasi menggunakan algoritma *K-Means*. Konsekuensi dari tinjauan ini menghasilkan akurasi 95,7% untuk segmentasi nukleus dan 93,1% untuk segmentasi sitoplasma (Febrinanto, et al., 2018).

BAB III

METODE PENELITIAN

3.1. Tempat dan Waktu Penelitian

3.1.1. Tempat Penelitian

Tempat penelitian ini dilaksanakan di Kebun Jeruk Simpang Banyak Kecamatan Ulupungkut Kabupaten Mandailing Natal.

3.1.2. Waktu Penelitian

Waktu penelitian ini dilaksanakan pada semester genap tahun ajaran 2019/2020 yaitu antara bulan Maret sampai dengan bulan Juni 2020 dengan tabel sebagai berikut:

Tabel 3.1 Waktu Penelitian

No	Kegiatan	Bulan 2020			
		Maret	April	Mei	juni
1.	Tahap persiapan penelitian				
	a. Penyusunan dan pengajuan judul				
	b. Pengajuan proposal				
	c. Perijinan penelitian				
2.	Tahap pelaksanaan				
	a. Pengumpulan data				
	b. Analisis data				
3.	Tahap penyusunan laporan				

3.2. Bahan dan Alat Penelitian

Penelitian ini diarahkan dengan memanfaatkan buah jeruk untuk membantu dan mendukung pelaksanaan penelitian.

3.2.1. Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan sistem ini diperlukan sebagai berikut:

1. *Processor* : *Intel(R) Celeron(R) CPU N3060 @1.60GHz*
2. *Memory* : *Random Access Memory (RAM) 4 GB*
3. Penyimpanan : *hardisk 500 GB.*

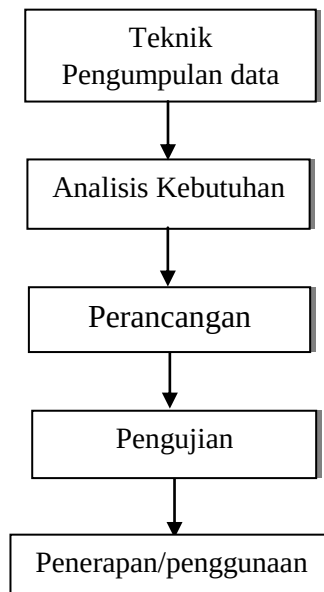
3.2.2. Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan sistem ini diperlukan sebagai berikut :

1. *Operating System Windows 10 Pro 64 bit.*
2. Matlab R2015a

3.3. Prosedur Kerja

Prosedur kerja dalam penelitian ini dilakukan melalui tahapan sebagai berikut:



Gambar 3.1.Tahap – Tahap Prosedur Kerja

3.3.1. Teknik Pengumpulan Data

Teknik pengumpulan data yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Penelitian Kepustakaan

Penelitian ini penulis lakukan untuk memperoleh data dengan meneliti dan berkonsentrasi pada tulisan atau sumber yang diidentifikasi dengan masalah yang diteliti. Penelitian kepustakaan bisa didapat dengan cara mencari jurnal, ebook yang diambil dari internet yang berencana untuk menemukan hipotesis saat ini dengan masalah yang diteliti.

2. Observasi

Observasi adalah pengumpulan informasi yang dilakukan dengan cara menilai atau mengunjungi perkebunan jeruk secara langsung di Desa Simpang Banyak Kecamatan Ulupungkut Kabupaten Mandailing Natal. Sebanyak 8 data gambar buah jeruk yang dikumpulkan dan dibagi menjadi 1 data sample dan 8 data uji. Cara pengambilan gambar dengan memetik buah jeruk dan diletakkan di atas kertas berwarna putih, setelah itu pengambilan gambar pada buah jeruk menggunakan kamera smartphone redmi 6A dengan spesifikasi kamera 13MP.

Setelah itu dilakukan pengambilan gambar sebanyak 1 gambar sebagai data uji buah jeruk. Pengambilan gambar yang dilakukan menggunakan 6 sudut yaitu dari sudut atas, bawah, kiri, kanan, depan dan belakang. Jadi, dalam 1 gambar memiliki jumlah sample sebanyak 6 buah gambar.

3. Wawancara

Wawancara adalah teknik pengumpulan data yang dilakukan melalui tatap muka dan tanya jawab langsung antara peneliti dan narasumber. Sebelum melakukan wawancara, peneliti menulis terlebih dahulu membuat daftar pertanyaan guna untuk memudahkan proses wawancara dan mendapatkan sebuah data yang lengkap sehingga lebih mudah untuk melakukan analisa data.

Wawancara yang dilakukan dengan tanya jawab kepada petani buah jeruk. Hal ini dilakukan untuk menggali, mengumpulkan, menemukan informasi yang dibutuhkan atau yang berhubungan dengan penelitian.

Berdasarkan data yang didapatkan dari hasil Penelitian Kepustakaan, Observasi dan Wawancara yaitu buah jeruk yang dikumpulkan sebanyak 8 buah jeruk yang diambil menggunakan 6 sudut kemudian diambil 1 gambar sebagai data uji buah jeruk. Jadi dataset buah jeruk yang digunakan dalam pengujian sebanyak 48 dataset buah jeruk.

Berikut daftar pertanyaan dan dataset yang dibuat penulis dalam proses wawancara:

Tabel 3.2. Daftar Pertanyaan

No	Daftar Pertanyaan
1	Bagaimana cara bapak dalam mengenali kematangan dari buah jeruk?
2	Berapa lamakah proses atau waktu pemetikan buah jeruk?
3	Apa jenis buah jeruk yang dibudidayakan disini?
4	Berdasarkan warna buah jeruk, menurut bapak kematangan buah jeruk ada berapa warna?
5	Menurut bapak sulitkah menentukan kematangan buah jeruk?

3.3.2. Analisis Kebutuhan

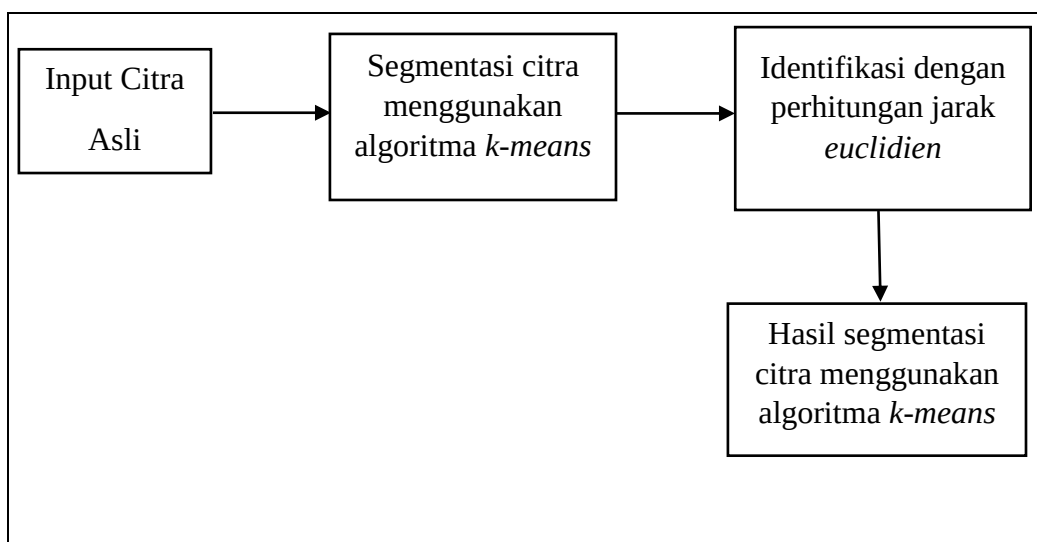
Analisis kebutuhan adalah sebuah proses yang dilakukan untuk mendapatkan sebuah informasi kebutuhan apa saja yang digunakan untuk membangun sistem agar mendukung berjalan semestinya. Kebutuhan tersebut mencakup perangkat keras dan perangkat lunak.

Algoritma *K-means* adalah metode yang dipilih dalam pengclusteran karena algoritma *K-means* mampu mempartisi data ke dalam *cluster* sehingga data yang memiliki karakteristik yang sama dikelompokkan ke dalam satu *cluster* yang sama dan data yang mempunyai karakteristik yang berbeda di kelompokkan ke dalam *cluster* yang lain. Pengclusteran yang dilakukan yaitu terdiri dari 2 jenis *cluster* yaitu matang dan mentah. Selain itu, algoritma ini mudah untuk diimplementasikan dan dijalankan serta waktu yang digunakan untuk proses pembelajaran relatif cepat.

3.3.3. Perancangan

Perancangan adalah langkah pertama dalam fase pengembangan rekayasa produk atau sistem. Perancangan itu adalah proses penerapan berbagai teknik dan prinsip yang bertujuan untuk mendefinisikan sebuah peralatan, satu proses atau satu sistem secara detail yang membolehkan dilakukan realisasi fisik. Fase ini adalah inti teknis dari proses rekayasa perangkat lunak.

Adapun diagram perencanaan dalam algoritma *k-means* adalah sebagai berikut:



Gambar 3.2. Diagram Perencanaan Algoritma *K-Means*

Gambar di atas merupakan alur dari sebuah sistem dalam melakukan sebuah segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *K-means*. Gambar buah jeruk yang telah diambil sebanyak 8 buah jeruk yaitu 1 data sample dan 8 data uji menggunakan 2 jenis cluster yaitu matang dan mentah dengan warna hijau, hijau kekuning-kuningan, dan kuning dimulai dari menginput citra asli dengan format .jpg, kemudian segmentasi citra menggunakan algoritma *k-means*. Selanjutnya tahap identifikasi dengan perhitungan jarak *euclidean* sebagai acuan dalam menentukan kematangan buah jeruk. Kemudian perhitungan jarak *euclidean* dilakukan secara berulang-ulang hingga lengkap meliputi seluruh hasil nilai data dalam perpindahan satu *cluster* ke *cluster* lainnya. Maka hasil segmentasi citra menggunakan algoritma *k-means* pun didapatkan dan proses segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means* selesai.

3.3.4. Pengujian

Pengujian sistem ditujukan untuk menguji kinerja sistem serta mengetahui secara cepat dan lebih efektif langkah apa yang harus dilakukan oleh penguji dalam memilih kematangan dari buah jeruk. Pengujian dilakukan oleh penguji dengan cara menggunakan sistem tersebut secara langsung. Sistem akan diujikan dari beberapa buah jeruk yang dipilih secara acak pada setiap pengujian dengan menggunakan 8 sample buah jeruk. Pengujian akan dimulai dari menentukan jumlah pengclusteran dari 2 jenis *cluster* yang masing-masing memiliki 3 warna dari buah jeruk dengan penjelasan tentang segmentasi kematangan dan kemiripan warna buah jeruk menggunakan algoritma *K-means*. Setelah program pengclusteran selesai maka penguji akan diminta menginput hasil *cluster* untuk menentukan apakah buah tersebut matang ataupun mentah, setelah itu muncullah hasil dari *pengclusteran* dan program sistem pun selesai.

3.3.5. Penerapan / Penggunaan

Penerapan/ penggunaan sistem ini adalah untuk mengetahui tingkat kematangan buah jeruk secara cepat. Dengan menggunakan segmentasi dan menerapkan algoritma *K-means* dapat mempercepat pemilihan buah jeruk bagi penguji. Serta membantu penguji dalam meminimalisir kesalahan dalam memilih tingkat kematangan buah jeruk.

BAB IV

PEMBAHASAN DAN HASIL

4.1. Pembahasan

Beberapa tahapan yang akan dibahas pada penelitian ini yaitu Analisis data, Reprsntasi data, hasil analisis data dan Perancangan sistem yaitu sebagai berikut.

4.1.1. Analisis Data

Pada analisis data untuk segmentasi kematangan buah jeruk berdasarkan kemiripan warna dilakukan pengumpulan data dan analisis kebutuhan. Pengumpulan data yaitu, dilakukan untuk memperoleh beberapa informasi yang berkaitan dalam pembuatan sistem pengujian menentukan kematangan buah jeruk berdasarkan kemiripan warna yaitu berupa data set dari buah jeruk. Analisis kebutuhan terdiri dari kebutuhan proses, kebutuhan masukan dan kebutuhan keluaran. Analisis kebutuhan proses, yaitu menjelaskan bagaimana sistem akan bekerja, proses-proses apa yang digunakan, mulai dari masuknya data input yang kemudian diproses oleh sistem hingga menjadi data output (tampilan akhir sistem).

Berikut dataset buah jeruk yang digunakan dalam proses segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means*:

Tabel 4.1 dataset buah jeruk

No	Dataset					
	Atas	Bawah	Depan	Belakang	Kanan	Kiri

1						
2						
3						
4						
5						
6						
7						



Implementasi matlab menggunakan algoritma *k-means* untuk menganalisis kematangan buah jeruk menggunakan matlab 2015. Ada beberapa langkah program yang dilakukan yaitu :

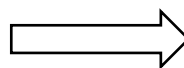
- Menginput citra buah jeruk.
- Segmentasi citra menggunakan algoritma *k-means*.
- Identifikasi jenis buah jeruk melalui perhitungan jarak *euclidean*.
- Hasil segmentasi citra menggunakan algoritma *k-means*

4.1.2. Representasi Data

Data yang digunakan berupa citra buah jeruk berjenis *grayscale* dan akan dilakukan proses segmentasi citra buah jeruk dengan metode *thresholding* menggunakan algoritma *k-means*, untuk melakukan pengujian *sample* penulis menggunakan citra buah jeruk berjenis *grayscale* dengan ukuran 4 x 4 *pixel*. Gambar 4.1.3. merupakan *sample* nilai intensitas *pixel* citra buah jeruk berjenis *grayscale* dengan ukuran citra 4 x 4 *pixel*.



Gambar 4.1 Citra Asli



194	200	211	211
181	91	155	200
100	1	78	194
81	106	183	189

Gambar 4.2 Size 4 x 4

Sample citra diatas merupakan citra buah jeruk yang memiliki nilai disetiap *pixel*, citra tersebut berjenis *grayscale* yang memiliki rentang warna 0- 255. *Sample* citra tersebut terdiri dari 4 baris dan 4 kolom yang akan digunakan untuk melakukan proses segmentasi citra buah jeruk dengan metode *thresholding* menggunakan algoritma *k-means*.

4.1.3. Hasil Analisis Data

Pada proses implementasi metode *thresholding* menggunakan algoritma *k-means* dalam melakukan proses segmentasi citra buah jeruk penulis menggunakan dataset citra buah jeruk berjenis *grayscale* dengan ukuran citra 4 x 4 *pixel* dengan kedalaman 8 bit dengan format *.jpg. Untuk gambar jeruk 1 dari nilai *grayscale* sebagai berikut:

194	200	211	211
181	91	155	200
100	1	78	194
81	106	183	189

Gambar 4.3 Citra *Grayscale* 4x4

Langkah-langkah segmentasi citra *grayscale* dengan algoritma *k-means* adalah sebagai berikut:

$$f(x, y) = 1, \text{ jika } f(x, y) \geq T$$

$$f(x, y) = 0, \text{ jika } f(x, y) \leq T$$

Dengan metode ini, nilai *Threshold* T adalah:

$$T = \frac{f_{maks} + f_{min}}{2}$$

$$T = \frac{211 + 1}{2}$$

$$T = 106$$

Dari hasil penentuan nilai *thresholding* dalam menentukan nilai biner, maka sebagai acuannya ditentukan dengan nilai dari *threshold*nya. Jika nilai matriks lebih kecil dari nilai *threshold* maka hasilnya bernilai 0 dan jika nilai matriks lebih besar dari nilai *threshold* maka hasilnya bernilai 1. Dapat dilihat seperti gambar 4.1.3.2. citra *Grayscale* dan gambar 4.3 citra hasil segmentasi dengan algoritma *k-means*.

194	200	211	211
-----	-----	-----	-----

181	91	155	200
100	1	78	194
81	106	183	189

Gambar 4.4 Citra *Grayscale*

↓ T = 106

1	1	1	1
1	0	1	1
0	0	0	1
0	0	1	1

Gambar 4.5 Citra hasil segmentasi dengan algoritma *k-means*

Untuk gambar jeruk 2 dari nilai *grayscale* sebagai berikut:

200	213	215	217
171	64	145	212
84	27	30	180
70	82	180	198

Gambar 4.6 Citra *Grayscale 4x4*

Langkah-langkah segmentasi citra *grayscale* dengan algoritma *k-means* adalah sebagai berikut:

$$f(x, y) = 1, \text{ jika } f(x, y) \geq T$$

$$f(x, y) = 0, \text{ jika } f(x, y) \leq T$$

Dengan metode ini, nilai *Threshold* T adalah:

$$T = \frac{f_{maks} + f_{min}}{2}$$

$$T = \frac{217 + 27}{2}$$

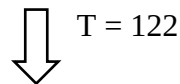
$$T = 122$$

Dari hasil penentuan nilai *thresholding* dalam menentukan nilai biner, maka sebagai acuannya ditentukan dengan nilai dari *threshold*nya. Jika nilai matriks

lebih kecil dari nilai *threshold* maka hasilnya bernilai 0 dan jika nilai matriks lebih besar dari nilai *threshold* maka hasilnya bernilai 1. Dapat dilihat seperti gambar 4.5 citra *Grayscale* dan gambar 4.6 citra hasil segmentasi dengan algoritma *k-means*.

200	213	215	217
171	64	145	212
84	27	30	180
70	82	180	198

Gambar 4.7Citra *Grayscale*



1	1	1	1
1	0	1	1
0	0	0	1
0	0	1	1

Gambar 4.8Citra hasil segmentasi dengan algoritma *k-means*

Langkah-langkah proses identifikasi citra buah jeruk dengan perhitungan jarak *euclidean* menggunakan algoritma *k-means* sebagai berikut:

1. Dataset

Tabel 4.2 merupakan tabel dataset dari 8 buah jeruk yang diambil dari 6 sudut dan menghasilkan gambar sebanyak 48 gambar. Dari 48 gambar buah jeruk tersebut akan dikelompokkan menjadi 2 bagian yaitu matang dan mentah. Hasil pengujian dataset segmentasi dengan *thresholding* menggunakan algoritma *k-means* dapat dilihat pada tabel 4.2 dataset.

Tabel 4.2Dataset

No	Dataset	Citra Hasil Segmentasi Dengan <i>k-means</i>
----	---------	--

1	Jeruk1	106
2	Jeruk2	122
3	Jeruk3	114
4	Jeruk4	115,5
5	Jeruk5	114,5
6	Jeruk6	110,5
7	Jeruk7	113,5
8	Jeruk8	136
9	Jeruk9	136
10	Jeruk10	121,5
11	Jeruk11	127
12	Jeruk12	114,5
13	Jeruk13	112
14	Jeruk14	117
15	Jeruk15	119,5
16	Jeruk16	114,5
17	Jeruk17	114
18	Jeruk18	116
19	Jeruk19	123
20	Jeruk20	127,5
21	Jeruk21	113,5
22	Jeruk22	116,5
23	Jeruk23	114,5
24	Jeruk24	125
25	Jeruk25	121
26	Jeruk26	126,5
27	Jeruk27	117,5
28	Jeruk28	116,5
29	Jeruk29	118,5
30	Jeruk30	113

31	Jeruk31	110,5
32	Jeruk32	113
33	Jeruk33	111
34	Jeruk34	113,5
35	Jeruk35	111,5
36	Jeruk36	115,5
37	Jeruk37	113,5
38	Jeruk38	119
39	Jeruk39	113
40	Jeruk40	112
41	Jeruk41	113
42	Jeruk42	111,5
43	Jeruk43	112,5
44	Jeruk44	115
45	Jeruk45	118
46	Jeruk46	119,5
47	Jeruk47	116,5
48	Jeruk48	118,5

2. Tentukan jumlah *cluster* yang akan dibentuk.

Cluster 1(C1) = Matang

Cluster 2 (C2) = Mentah

3. Tetapkan C pusat *cluster* awal secara random. Dalam menentukan C pusat *cluster* awal secara random yang diambil dari dataset buah jeruk yaitu pada jeruk 1 dan jeruk 2.

C1 = 106

C2 = 122

4. Alokasikan semua data/obyek ke dalam *cluster* terdekat. Berikut hasil dari alokasi data ke jarak *cluster*. Adapun hasil dari jarak ke *cluster* diperoleh dari perhitungan dengan rumus:

$$d(x, y) = \sqrt{\sum_{i=1}^n (xi - yi)^2}$$

Iterasi 1:

$$d_{1,1} = \sqrt{(106 - 106)^2} = 0$$

$$d_{1,2} = \sqrt{(122 - 106)^2} = 16$$

$$d_{2,1} = \sqrt{(106 - 122)^2} = -16$$

$$d_{2,2} = \sqrt{(122 - 122)^2} = 0$$

$$d_{3,1} = \sqrt{(106 - 114)^2} = -8$$

$$d_{3,2} = \sqrt{(122 - 114)^2} = 8$$

$$d_{4,1} = \sqrt{(106 - 115,5)^2} = -9,5$$

$$d_{4,2} = \sqrt{(122 - 115,5)^2} = 6,5$$

$$d_{5,1} = \sqrt{(106 - 114,5)^2} = -8,5$$

$$d_{5,2} = \sqrt{(122 - 114,5)^2} = 7,5$$

$$d_{6,1} = \sqrt{(106 - 110,5)^2} = -4,5$$

$$d_{6,2} = \sqrt{(122 - 110,5)^2} = 11,5$$

$$d_{7,1} = \sqrt{(106 - 113,5)^2} = -7,5$$

$$d_{7,2} = \sqrt{(122 - 113,5)^2} = 8,5$$

$$d_{8,1} = \sqrt{(106 - 136)^2} = -30$$

$$d_{8,2} = \sqrt{(122 - 136)^2} = -14$$

$$d_{(9,1)} = \sqrt{(106 - 136)^2} = -30$$

$$d_{(9,2)} = \sqrt{(122 - 136)^2} = -14$$

$$d_{(10,1)} = \sqrt{(106 - 121,5)^2} = -15,5$$

$$d_{(10,2)} = \sqrt{(122 - 121,5)^2} = 0,5$$

$$d_{(11,1)} = \sqrt{(106 - 127)^2} = -21$$

$$d_{(11,2)} = \sqrt{(122 - 127)^2} = -5$$

$$d_{(12,1)} = \sqrt{(106 - 114,5)^2} = -8,5$$

$$d_{(12,2)} = \sqrt{(122 - 114,5)^2} = 7,5$$

$$d(13,1) = \sqrt{(106 - 112)^2} = -6$$

$$d(13,2) = \sqrt{(122 - 112)^2} = 10$$

$$d(14,1) = \sqrt{(106 - 117)^2} = -11$$

$$d(14,2) = \sqrt{(122 - 117)^2} = 5$$

$$d(15,1) = \sqrt{(106 - 119,5)^2} = -13,5$$

$$d(15,2) = \sqrt{(122 - 119,5)^2} = 2,5$$

$$d(16,1) = \sqrt{(106 - 114,5)^2} = -8,5$$

$$d(16,2) = \sqrt{(122 - 114,5)^2} = 7,5$$

$$d(17,1) = \sqrt{(106 - 114)^2} = -8$$

$$d(17,2) = \sqrt{(122 - 114)^2} = 8$$

$$d(18,1) = \sqrt{(106 - 116)^2} = -10$$

$$d(18,2) = \sqrt{(122 - 116)^2} = 6$$

$$d(19,1) = \sqrt{(106 - 123)^2} = -17$$

$$d(19,2) = \sqrt{(122 - 123)^2} = -1$$

$$d(20,1) = \sqrt{(106 - 127,5)^2} = -21,5$$

$$d(20,2) = \sqrt{(122 - 127,5)^2} = -5,5$$

$$d(21,1) = \sqrt{(106 - 113,5)^2} = -7,5$$

$$d(21,2) = \sqrt{(122 - 113,5)^2} = 8,5$$

$$d(22,1) = \sqrt{(106 - 116,5)^2} = -10,5$$

$$d(22,2) = \sqrt{(122 - 116,5)^2} = 5,5$$

$$d(23,1) = \sqrt{(106 - 114,5)^2} = -8,5$$

$$d(23,2) = \sqrt{(122 - 114,5)^2} = 7,5$$

$$d(24,1) = \sqrt{(106 - 125)^2} = -19$$

$$d(24,2) = \sqrt{(122 - 125)^2} = -3$$

$$d(25,1) = \sqrt{(106 - 121)^2} = -15$$

$$d(25,2) = \sqrt{(122 - 121)^2} = 1$$

$$d(26,1) = \sqrt{(106 - 126,5)^2} = -20,5$$

$$d(26,2) = \sqrt{(122 - 126,5)^2} = -4,5$$

$$d(27,1) = \sqrt{(106 - 117,5)^2} = -11,5$$

$$d(27,2) = \sqrt{(122 - 117,5)^2} = 4,5$$

$$d(28,1) = \sqrt{(106 - 116,5)^2} = -10,5$$

$$d(28,2) = \sqrt{(122 - 116,5)^2} = 5,5$$

$$d(29,1) = \sqrt{(106 - 118,5)^2} = -12,5$$

$$d(29,2) = \sqrt{(122 - 118,5)^2} = 3,5$$

$$d(30,1) = \sqrt{(106 - 113)^2} = -7$$

$$d(30,2) = \sqrt{(122 - 113)^2} = 9$$

$$d(31,1) = \sqrt{(106 - 110,5)^2} = -4,5$$

$$d(31,2) = \sqrt{(122 - 110,5)^2} = 11,5$$

$$d(32,1) = \sqrt{(106 - 113)^2} = -7$$

$$d(32,2) = \sqrt{(122 - 113)^2} = 9$$

$$d(33,1) = \sqrt{(106 - 111)^2} = -5$$

$$d(33,2) = \sqrt{(122 - 111)^2} = 11$$

$$d(34,1) = \sqrt{(106 - 113,5)^2} = -7,5$$

$$d(34,2) = \sqrt{(122 - 113,5)^2} = 8,5$$

$$d(35,1) = \sqrt{(106 - 111,5)^2} = -5,5$$

$$d(35,2) = \sqrt{(122 - 111,5)^2} = 10,5$$

$$d(36,1) = \sqrt{(106 - 115,5)^2} = -9,5$$

$$d(36,2) = \sqrt{(122 - 115,5)^2} = 6,5$$

$$d(37,1) = \sqrt{(106 - 113,5)^2} = -7,5$$

$$d(37,2) = \sqrt{(122 - 113,5)^2} = 8,5$$

$$d(38,1) = \sqrt{(106 - 119)^2} = -13$$

$$d(38,2) = \sqrt{(122 - 119)^2} = 3$$

$$d(39,1) = \sqrt{(106 - 113)^2} = -7$$

$$d(39,2) = \sqrt{(122 - 113)^2} = 9$$

$$d(40,1) = \sqrt{(106 - 112)^2} = -6$$

$$d(40,2) = \sqrt{(122 - 112)^2} = 10$$

$$d(41,1) = \sqrt{(106 - 113)^2} = -7$$

$$d(41,2) = \sqrt{(122 - 113)^2} = 9$$

$$d(42,1) = \sqrt{(106 - 111,5)^2} = -5,5$$

$$d(42,2) = \sqrt{(122 - 111,5)^2} = 10,5$$

$$d(43,1) = \sqrt{(106 - 112,5)^2} = -6,5$$

$$d(43,2) = \sqrt{(122 - 112,5)^2} = 9,5$$

$$d(44,1) = \sqrt{(106 - 115)^2} = -9$$

$$d(44,2) = \sqrt{(122 - 115)^2} = 7$$

$$d(45,1) = \sqrt{(106 - 118)^2} = -12$$

$$d(45,2) = \sqrt{(122 - 118)^2} = 4$$

$$d(46,1) = \sqrt{(106 - 119,5)^2} = -13,5$$

$$d(46,2) = \sqrt{(122 - 119,5)^2} = 2,5$$

$$d(47,1) = \sqrt{(106 - 116,5)^2} = -10,5$$

$$d(47,2) = \sqrt{(122 - 116,5)^2} = 5,5$$

$$d(48,1) = \sqrt{(106 - 118,5)^2} = -12,5$$

$$d(48,2) = \sqrt{(122 - 118,5)^2} = 3,5$$

Setelah melakukan perhitungan iterasi 1 maka didapatkan hasil seperti berikut ini:

Tabel 4.3Jarak Ke *Cluster*

No	Dataset	Jarak Ke <i>Cluster</i>		Hasil
		C1	C2	
1	Jeruk1	0	16	1
2	Jeruk2	-16	0	1

3	Jeruk3	-8	8	1
4	Jeruk4	-9,5	6,5	1
5	Jeruk5	-8,5	7,5	1
6	Jeruk6	-4,5	11,5	1
7	Jeruk7	-7,5	8,5	1
8	Jeruk8	-30	-14	1
9	Jeruk9	-30	-14	1
10	Jeruk10	-15,5	0,5	1
11	Jeruk11	-21	-5	1
12	Jeruk12	-8,5	7,5	1
13	Jeruk13	-6	10	1
14	Jeruk14	-11	5	1
15	Jeruk15	-13,5	2,5	1
16	Jeruk16	-8,5	7,5	1
17	Jeruk17	-8	8	1
18	Jeruk18	-10	6	1
19	Jeruk19	-17	-1	1
20	Jeruk20	-21,5	-5,5	1
21	Jeruk21	-7,5	8,5	1
22	Jeruk22	-10,5	5,5	1
23	Jeruk23	-8,5	7,5	1
24	Jeruk24	-19	-3	1
25	Jeruk25	-15	1	1
26	Jeruk26	-20,5	-4,5	1
27	Jeruk27	-11,5	4,5	1
28	Jeruk28	-10,5	5,5	1
29	Jeruk29	-12,5	3,5	1
30	Jeruk30	-7	9	1
31	Jeruk31	-4,5	11,5	1
32	Jeruk32	-7	9	1

33	Jeruk33	-5	11	1
34	Jeruk34	-7,5	8,5	1
35	Jeruk35	-5,5	10,5	1
36	Jeruk36	-9,5	6,5	1
37	Jeruk37	-7,5	8,5	1
38	Jeruk38	-13	3	1
39	Jeruk39	-7	9	1
40	Jeruk40	-6	10	1
41	Jeruk41	-7	9	1
42	Jeruk42	-5,5	10,5	1
43	Jeruk43	-6,5	9,5	1
44	Jeruk44	-9	7	1
45	Jeruk45	-12	4	1
46	Jeruk46	-13,5	2,5	1
47	Jeruk47	-10,5	5,5	1
48	Jeruk48	-12,5	3,5	1

5. Tentukan kembali titik pusat *cluster* yang baru berdasarkan rata-rata *cluster* baru tersebut didapat dari rumus: nilai hasil / banyak hasil.

$$\begin{aligned}
 C1 = & (106 + 122 + 114 + 115,5 + 114,5 + 110,5 + 113,5 + 136 + 136 + \\
 & 121,5 + 127 + 114,5 + 112 + 117 + 119,5 + 114,5 + 114 + 116 + 123 \\
 & + 127,5 + 113,5 + 116,5 + 114,5 + 125 + 121 + 126,5 + 117,5 + 116,5 \\
 & + 118,5 + 113 + 110,5 + 113 + 111 + 113,5 + 111,5 + 115,5 + 113,5 \\
 & + 119 + 113 + 112 + 113 + 111,5 + 112,5 + 115 + 118 + 119,5 + \\
 & 116,5 + 118,5) / 48
 \end{aligned}$$

$$= 5614 / 48 = 116,96$$

$$C2 = 0$$

6. Lakukan kembali langkah 4 hingga titik pusat dari setiap *cluster* tidak berubah.

Iterasi 2:

$$d_{1,1} = \sqrt{(116,96 - 106)^2} = 10,96$$

$$d_{1,2} = \sqrt{(0 - 106)^2} = -106$$

$$d_{2,1} = \sqrt{(116,96 - 122)^2} = -5,04$$

$$d_{2,2} = \sqrt{(0 - 122)^2} = -122$$

$$d_{3,1} = \sqrt{(116,96 - 114)^2} = 2,96$$

$$d_{3,2} = \sqrt{(0 - 114)^2} = -114$$

$$d_{4,1} = \sqrt{(116,96 - 115,5)^2} = 1,46$$

$$d_{4,2} = \sqrt{(0 - 115,5)^2} = -115,5$$

$$d_{5,1} = \sqrt{(116,96 - 114,5)^2} = 2,46$$

$$d_{5,2} = \sqrt{(0 - 114,5)^2} = -114,5$$

$$d_{6,1} = \sqrt{(116,96 - 110,5)^2} = 6,46$$

$$d_{6,2} = \sqrt{(0 - 110,5)^2} = -110,5$$

$$d_{7,1} = \sqrt{(116,96 - 113,5)^2} = 3,46$$

$$d_{7,2} = \sqrt{(0 - 113,5)^2} = -113,5$$

$$d_{8,1} = \sqrt{(116,96 - 136)^2} = -19,04$$

$$d_{8,2} = \sqrt{(0 - 136)^2} = -136$$

$$d_{(9,1)} = \sqrt{(116,96 - 136)^2} = -19,04$$

$$d_{(9,2)} = \sqrt{(0 - 136)^2} = -136$$

$$d_{(10,1)} = \sqrt{(116,96 - 121,5)^2} = -4,54$$

$$d_{(10,2)} = \sqrt{(0 - 121,5)^2} = -121,5$$

$$d_{(11,1)} = \sqrt{(116,96 - 127)^2} = -10,04$$

$$d_{(11,2)} = \sqrt{(0 - 127)^2} = -127$$

$$d_{(12,1)} = \sqrt{(116,96 - 114,5)^2} = 2,46$$

$$d_{(12,2)} = \sqrt{(0 - 114,5)^2} = -114,5$$

$$d_{(13,1)} = \sqrt{(116,96 - 112)^2} = 4,96$$

$$d_{(13,2)} = \sqrt{(0 - 112)^2} = -112$$

$$d(14,1) = \sqrt{(116,96 - 117)^2} = -0,04$$

$$d(14,2) = \sqrt{(0 - 117)^2} = -117$$

$$d(15,1) = \sqrt{(116,96 - 119,5)^2} = -2,54$$

$$d(15,2) = \sqrt{(0 - 119,5)^2} = -119,5$$

$$d(16,1) = \sqrt{(116,96 - 114,5)^2} = 2,46$$

$$d(16,2) = \sqrt{(0 - 114,5)^2} = -114,5$$

$$d(17,1) = \sqrt{(116,96 - 114)^2} = 2,96$$

$$d(17,2) = \sqrt{(0 - 114)^2} = -114$$

$$d(18,1) = \sqrt{(116,96 - 116)^2} = 0,96$$

$$d(18,2) = \sqrt{(0 - 116)^2} = -116$$

$$d(19,1) = \sqrt{(116,96 - 123)^2} = -6,04$$

$$d(19,2) = \sqrt{(0 - 123)^2} = -123$$

$$d(20,1) = \sqrt{(116,96 - 127,5)^2} = -10,54$$

$$d(20,2) = \sqrt{(0 - 127,5)^2} = -127,5$$

$$d(21,1) = \sqrt{(116,96 - 113,5)^2} = 3,46$$

$$d(21,2) = \sqrt{(0 - 113,5)^2} = -113,5$$

$$d(22,1) = \sqrt{(116,96 - 116,5)^2} = 0,46$$

$$d(22,2) = \sqrt{(0 - 116,5)^2} = -116,5$$

$$d(23,1) = \sqrt{(116,96 - 114,5)^2} = 2,46$$

$$d(23,2) = \sqrt{(0 - 114,5)^2} = -114,5$$

$$d(24,1) = \sqrt{(116,96 - 125)^2} = -8,04$$

$$d(24,2) = \sqrt{(0 - 125)^2} = -125$$

$$d(25,1) = \sqrt{(116,96 - 121)^2} = -4,04$$

$$d(25,2) = \sqrt{(0 - 121)^2} = -121$$

$$d(26,1) = \sqrt{(116,96 - 126,5)^2} = -9,54$$

$$d(26,2) = \sqrt{(0 - 126,5)^2} = -126,5$$

$$d(27,1) = \sqrt{(116,96 - 117,5)^2} = -0,54$$

$$d(27,2) = \sqrt{(0 - 117,5)^2} = -117,5$$

$$d(28,1) = \sqrt{(116,96 - 116,5)^2} = 0,46$$

$$d(28,2) = \sqrt{(0 - 116,5)^2} = -116,5$$

$$d(29,1) = \sqrt{(116,96 - 118,5)^2} = -1,54$$

$$d(29,2) = \sqrt{(0 - 118,5)^2} = -118,5$$

$$d(30,1) = \sqrt{(116,96 - 113)^2} = 3,96$$

$$d(30,2) = \sqrt{(0 - 113)^2} = -113$$

$$d(31,1) = \sqrt{(116,96 - 110,5)^2} = 6,46$$

$$d(31,2) = \sqrt{(0 - 110,5)^2} = -110,5$$

$$d(32,1) = \sqrt{(116,96 - 113)^2} = 3,96$$

$$d(32,2) = \sqrt{(0 - 113)^2} = -113$$

$$d(33,1) = \sqrt{(116,96 - 111)^2} = 5,96$$

$$d(33,2) = \sqrt{(0 - 111)^2} = -111$$

$$d(34,1) = \sqrt{(116,96 - 113,5)^2} = 3,46$$

$$d(34,2) = \sqrt{(0 - 113,5)^2} = -113,5$$

$$d(35,1) = \sqrt{(116,96 - 111,5)^2} = 5,46$$

$$d(35,2) = \sqrt{(0 - 111,5)^2} = -111,5$$

$$d(36,1) = \sqrt{(116,96 - 115,5)^2} = 1,46$$

$$d(36,2) = \sqrt{(0 - 115,5)^2} = -115,5$$

$$d(37,1) = \sqrt{(116,96 - 113,5)^2} = 3,46$$

$$d(37,2) = \sqrt{(0 - 113,5)^2} = -113,5$$

$$d(38,1) = \sqrt{(116,96 - 119)^2} = -2,04$$

$$d(38,2) = \sqrt{(0 - 119)^2} = -119$$

$$d(39,1) = \sqrt{(116,96 - 113)^2} = 3,96$$

$$d(39,2) = \sqrt{(0 - 113)^2} = -113$$

$$d(40,1) = \sqrt{(116,96 - 112)^2} = 4,96$$

$$d(40,2) = \sqrt{(0 - 112)^2} = -112$$

$$d(41,1) = \sqrt{(116,96 - 113)^2} = 3,96$$

$$d(41,2) = \sqrt{(0 - 113)^2} = -113$$

$$d(42,1) = \sqrt{(116,96 - 111,5)^2} = 5,46$$

$$d(42,2) = \sqrt{(0 - 111,5)^2} = -111,5$$

$$d(43,1) = \sqrt{(116,96 - 112,5)^2} = 4,46$$

$$d(43,2) = \sqrt{(0 - 112,5)^2} = -112,5$$

$$d(44,1) = \sqrt{(116,96 - 115)^2} = 1,96$$

$$d(44,2) = \sqrt{(0 - 115)^2} = -115$$

$$d(45,1) = \sqrt{(116,96 - 118)^2} = -1,04$$

$$d(45,2) = \sqrt{(0 - 118)^2} = -118$$

$$d(46,1) = \sqrt{(116,96 - 119,5)^2} = -2,54$$

$$d(46,2) = \sqrt{(0 - 119,5)^2} = -119,5$$

$$d(47,1) = \sqrt{(116,96 - 116,5)^2} = 0,46$$

$$d(47,2) = \sqrt{(0 - 116,5)^2} = -116,5$$

$$d(48,1) = \sqrt{(116,96 - 118,5)^2} = -1,54$$

$$d(48,2) = \sqrt{(0 - 118,5)^2} = -118,5$$

Setelah melakukan perhitungan iterasi 2 maka didapatkan hasil seperti berikut ini:

Tabel 4.4Jarak Ke *Cluster*

No	Dataset	Jarak Ke <i>Cluster</i>		Hasil
		C1	C2	
1	Jeruk1	10,96	-106	2
2	Jeruk2	-5,04	-122	2
3	Jeruk3	2,96	-114	2
4	Jeruk4	1,46	-115,5	2

5	Jeruk5	2,46	-114,5	2
6	Jeruk6	6,46	-110,5	2
7	Jeruk7	3,46	-113,5	2
8	Jeruk8	-19,04	-136	2
9	Jeruk9	-19,04	-136	2
10	Jeruk10	-4,54	-121,5	2
11	Jeruk11	-10,04	-127	2
12	Jeruk12	2,46	-114,5	2
13	Jeruk13	4,96	-112	2
14	Jeruk14	-0,04	-117	2
15	Jeruk15	-2,54	-119,5	2
16	Jeruk16	2,46	-114,5	2
17	Jeruk17	2,96	-114	2
18	Jeruk18	0,96	-116	2
19	Jeruk19	-6,04	-123	2
20	Jeruk20	-10,54	-127,5	2
21	Jeruk21	3,46	-113,5	2
22	Jeruk22	0,46	-116,5	2
23	Jeruk23	2,46	-114,5	2
24	Jeruk24	-8,04	-125	2
25	Jeruk25	-4,04	-121	2
26	Jeruk26	-9,54	-126,5	2
27	Jeruk27	-0,54	-117,5	2
28	Jeruk28	0,46	-116,5	2
29	Jeruk29	-1,54	-118,5	2
30	Jeruk30	3,96	-113	2
31	Jeruk31	6,46	-110,5	2
32	Jeruk32	3,96	-113	2
33	Jeruk33	5,96	-111	2
34	Jeruk34	3,46	-113,5	2

35	Jeruk35	5,46	-111,5	2
36	Jeruk36	1,46	-115,5	2
37	Jeruk37	3,46	-113,5	2
38	Jeruk38	-2,04	-119	2
39	Jeruk39	3,96	-113	2
40	Jeruk40	4,96	-112	2
41	Jeruk41	3,96	-113	2
42	Jeruk42	5,46	-111,5	2
43	Jeruk43	4,46	-112,5	2
44	Jeruk44	1,96	-115	2
45	Jeruk45	-1,04	-118	2
46	Jeruk46	-2,54	-119,5	2
47	Jeruk47	0,46	-116,5	2
48	Jeruk48	-1,54	-118,5	2

7. Tentukan kembali titik pusat *cluster* yang baru berdasarkan rata-rata *cluster* baru tersebut didapat dari rumus: nilai hasil / banyak hasil.

$$C1 = 0$$

$$C2 = (106 + 122 + 114 + 115,5 + 114,5 + 110,5 + 113,5 + 136 + 136 + 121,5 + 127 + 114,5 + 112 + 117 + 119,5 + 114,5 + 114 + 116 + 123 + 127,5 + 113,5 + 116,5 + 114,5 + 125 + 121 + 126,5 + 117,5 + 116,5 + 118,5 + 113 + 110,5 + 113 + 111 + 113,5 + 111,5 + 115,5 + 113,5 + 119 + 113 + 112 + 113 + 111,5 + 112,5 + 115 + 118 + 119,5 + 116,5 + 118,5) / 48$$

$$= 5614 / 48 = 116,96$$

8. Lakukan kembali langkah 4 hingga titik pusat dari setiap *cluster* tidak berubah.

Iterasi 3:

$$d1,1 = \sqrt{(0 - 106)^2} = -106$$

$$d1,2 = \sqrt{(116,96 - 106)^2} = 10,96$$

$$d_{2,1} = \sqrt{(0 - 122)^2} = -122$$

$$d_{2,2} = \sqrt{(116,96 - 122)^2} = -5,04$$

$$d_{3,1} = \sqrt{(0 - 114)^2} = -114$$

$$d_{3,2} = \sqrt{(116,96 - 114)^2} = 2,96$$

$$d_{4,1} = \sqrt{(0 - 115,5)^2} = -115,5$$

$$d_{4,2} = \sqrt{(116,96 - 115,5)^2} = 1,46$$

$$d_{5,1} = \sqrt{(0 - 114,5)^2} = -114,5$$

$$d_{5,2} = \sqrt{(116,96 - 114,5)^2} = 2,46$$

$$d_{6,1} = \sqrt{(0 - 110,5)^2} = -110,5$$

$$d_{6,2} = \sqrt{(116,96 - 110,5)^2} = 6,46$$

$$d_{7,1} = \sqrt{(0 - 113,5)^2} = -113,5$$

$$d_{7,2} = \sqrt{(116,96 - 113,5)^2} = 3,46$$

$$d_{8,1} = \sqrt{(0 - 136)^2} = -136$$

$$d_{8,2} = \sqrt{(116,96 - 136)^2} = -19,04$$

$$d_{9,1} = \sqrt{(0 - 136)^2} = -136$$

$$d_{9,2} = \sqrt{(116,96 - 136)^2} = -19,04$$

$$d_{10,1} = \sqrt{(0 - 121,5)^2} = -121,5$$

$$d_{10,2} = \sqrt{(116,96 - 121,5)^2} = -4,54$$

$$d_{11,1} = \sqrt{(0 - 127)^2} = -127$$

$$d_{11,2} = \sqrt{(116,96 - 127)^2} = -10,04$$

$$d_{12,1} = \sqrt{(0 - 114,5)^2} = -114,5$$

$$d_{12,2} = \sqrt{(116,96 - 114,5)^2} = 2,46$$

$$d_{13,1} = \sqrt{(0 - 112)^2} = -112$$

$$d_{13,2} = \sqrt{(116,96 - 112)^2} = 4,96$$

$$d_{14,1} = \sqrt{(0 - 117)^2} = -117$$

$$d_{14,2} = \sqrt{(116,96 - 117)^2} = -0,04$$

$$\begin{aligned}
d(15,1) &= \sqrt{(0 - 119,5)^2} = -119,5 \\
d(15,2) &= \sqrt{(116,96 - 119,5)^2} = -2,54 \\
d(16,1) &= \sqrt{(0 - 114,5)^2} = -114,5 \\
d(16,2) &= \sqrt{(116,96 - 114,5)^2} = 2,46 \\
d(17,1) &= \sqrt{(0 - 114)^2} = -114 \\
d(17,2) &= \sqrt{(116,96 - 114)^2} = 2,96 \\
d(18,1) &= \sqrt{(0 - 116)^2} = -116 \\
d(18,2) &= \sqrt{(116,96 - 116)^2} = 0,96 \\
d(19,1) &= \sqrt{(0 - 123)^2} = -123 \\
d(19,2) &= \sqrt{(116,96 - 123)^2} = -6,04 \\
d(20,1) &= \sqrt{(0 - 127,5)^2} = -127,5 \\
d(20,2) &= \sqrt{(116,96 - 127,5)^2} = -10,54 \\
d(21,1) &= \sqrt{(0 - 113,5)^2} = -113,5 \\
d(21,2) &= \sqrt{(116,96 - 113,5)^2} = 3,46 \\
d(22,1) &= \sqrt{(0 - 116,5)^2} = -116,5 \\
d(22,2) &= \sqrt{(116,96 - 116,5)^2} = 0,46 \\
d(23,1) &= \sqrt{(0 - 114,5)^2} = -114,5 \\
d(23,2) &= \sqrt{(116,96 - 114,5)^2} = 2,46 \\
d(24,1) &= \sqrt{(0 - 125)^2} = -125 \\
d(24,2) &= \sqrt{(116,96 - 125)^2} = -8,04 \\
d(25,1) &= \sqrt{(0 - 121)^2} = -121 \\
d(25,2) &= \sqrt{(116,96 - 121)^2} = -4,04 \\
d(26,1) &= \sqrt{(0 - 126,5)^2} = -126,5 \\
d(26,2) &= \sqrt{(116,96 - 126,5)^2} = -9,54 \\
d(27,1) &= \sqrt{(0 - 117,5)^2} = -117,5 \\
d(27,2) &= \sqrt{(116,96 - 117,5)^2} = -0,54
\end{aligned}$$

$$\begin{aligned}
d(28,1) &= \sqrt{(0 - 116,5)^2} = -116,5 \\
d(28,2) &= \sqrt{(116,96 - 116,5)^2} = 0,46 \\
d(29,1) &= \sqrt{(0 - 118,5)^2} = -118,5 \\
d(29,2) &= \sqrt{(116,96 - 118,5)^2} = -1,54 \\
d(30,1) &= \sqrt{(0 - 113)^2} = -113 \\
d(30,2) &= \sqrt{(116,96 - 113)^2} = 3,96 \\
d(31,1) &= \sqrt{(0 - 110,5)^2} = -110,5 \\
d(31,2) &= \sqrt{(116,96 - 110,5)^2} = 6,46 \\
d(32,1) &= \sqrt{(0 - 113)^2} = -113 \\
d(32,2) &= \sqrt{(116,96 - 113)^2} = 3,96 \\
d(33,1) &= \sqrt{(0 - 111)^2} = -111 \\
d(33,2) &= \sqrt{(116,96 - 111)^2} = 5,96 \\
d(34,1) &= \sqrt{(0 - 113,5)^2} = -113,5 \\
d(34,2) &= \sqrt{(116,96 - 113,5)^2} = 3,46 \\
d(35,1) &= \sqrt{(0 - 111,5)^2} = -111,5 \\
d(35,2) &= \sqrt{(116,96 - 111,5)^2} = 5,46 \\
d(36,1) &= \sqrt{(0 - 115,5)^2} = -115,5 \\
d(36,2) &= \sqrt{(116,96 - 115,5)^2} = 1,46 \\
d(37,1) &= \sqrt{(0 - 113,5)^2} = -113,5 \\
d(37,2) &= \sqrt{(116,96 - 113,5)^2} = 3,46 \\
d(38,1) &= \sqrt{(0 - 119)^2} = -119 \\
d(38,2) &= \sqrt{(116,96 - 119)^2} = -2,04 \\
d(39,1) &= \sqrt{(0 - 113)^2} = -113 \\
d(39,2) &= \sqrt{(116,96 - 113)^2} = 3,96 \\
d(40,1) &= \sqrt{(0 - 112)^2} = -112 \\
d(40,2) &= \sqrt{(116,96 - 112)^2} = 4,96
\end{aligned}$$

$$d(41,1) = \sqrt{(0 - 113)^2} = -113$$

$$d(41,2) = \sqrt{(116,96 - 113)^2} = 3,96$$

$$d(42,1) = \sqrt{(0 - 111,5)^2} = -111,5$$

$$d(42,2) = \sqrt{(116,96 - 111,5)^2} = 5,46$$

$$d(43,1) = \sqrt{(0 - 112,5)^2} = -112,5$$

$$d(43,2) = \sqrt{(116,96 - 112,5)^2} = 4,46$$

$$d(44,1) = \sqrt{(0 - 115)^2} = -115$$

$$d(44,2) = \sqrt{(116,96 - 115)^2} = 1,96$$

$$d(45,1) = \sqrt{(0 - 118)^2} = -118$$

$$d(45,2) = \sqrt{(116,96 - 118)^2} = -1,04$$

$$d(46,1) = \sqrt{(0 - 119,5)^2} = -119,5$$

$$d(46,2) = \sqrt{(116,96 - 119,5)^2} = -2,54$$

$$d(47,1) = \sqrt{(0 - 116,5)^2} = -116,5$$

$$d(47,2) = \sqrt{(116,96 - 116,5)^2} = 0,46$$

$$d(48,1) = \sqrt{(0 - 118,5)^2} = -118,5$$

$$d(48,2) = \sqrt{(116,96 - 118,5)^2} = -1,54$$

Setelah melakukan perhitungan iterasi 2 maka didapatkan hasil seperti berikut ini:

Tabel 4.5Jarak Ke *Cluster*

No	Dataset	Jarak Ke <i>Cluster</i>		Hasil
		C1	C2	
1	Jeruk1	-106	10,96	1
2	Jeruk2	-122	-5,04	1
3	Jeruk3	-114	2,96	1
4	Jeruk4	-115,5	1,46	1
5	Jeruk5	-114,5	2,46	1
6	Jeruk6	-110,5	6,46	1

7	Jeruk7	-113,5	3,46	1
8	Jeruk8	-136	-19,04	1
9	Jeruk9	-136	-19,04	1
10	Jeruk10	-121,5	-4,54	1
11	Jeruk11	-127	-10,04	1
12	Jeruk12	-114,5	2,46	1
13	Jeruk13	-112	4,96	1
14	Jeruk14	-117	-0,04	1
15	Jeruk15	-119,5	-2,54	1
16	Jeruk16	-114,5	2,46	1
17	Jeruk17	-114	2,96	1
18	Jeruk18	-116	0,96	1
19	Jeruk19	-123	-6,04	1
20	Jeruk20	-127,5	-10,54	1
21	Jeruk21	-113,5	3,46	1
22	Jeruk22	-116,5	0,46	1
23	Jeruk23	-114,5	2,46	1
24	Jeruk24	-125	-8,04	1
25	Jeruk25	-121	-4,04	1
26	Jeruk26	-126,5	-9,54	1
27	Jeruk27	-117,5	-0,54	1
28	Jeruk28	-116,5	0,46	1
29	Jeruk29	-118,5	-1,54	1
30	Jeruk30	-113	3,96	1
31	Jeruk31	-110,5	6,46	1
32	Jeruk32	-113	3,96	1
33	Jeruk33	-111	5,96	1
34	Jeruk34	-113,5	3,46	1
35	Jeruk35	-111,5	5,46	1
36	Jeruk36	-115,5	1,46	1

37	Jeruk37	-113,5	3,46	1
38	Jeruk38	-119	-2,04	1
39	Jeruk39	-113	3,96	1
40	Jeruk40	-112	4,96	1
41	Jeruk41	-113	3,96	1
42	Jeruk42	-111,5	5,46	1
43	Jeruk43	-112,5	4,46	1
44	Jeruk44	-115	1,96	1
45	Jeruk45	-118	-1,04	1
46	Jeruk46	-119,5	-2,54	1
47	Jeruk47	-116,5	0,46	1
48	Jeruk48	-118,5	-1,54	1

Hasil dari tahapan iterasi ke-1 dan iterasi ke-3 tidak berubah, maka hasil sudah sesuai dengan pengelompokan *cluster*. Berikut adalah hasil dari pengelompokan tersebut.

Tabel 4.6 hasil pengclusteran

Dataset	Keterangan
Jeruk1	Matang
Jeruk2	Matang
Jeruk3	Matang
Jeruk4	Matang
Jeruk5	Matang
Jeruk6	Matang
Jeruk7	Matang
Jeruk8	Matang
Jeruk9	Matang
Jeruk10	Matang
Jeruk11	Matang

Jeruk12	Matang
Jeruk13	Matang
Jeruk14	Matang
Jeruk15	Matang
Jeruk16	Matang
Jeruk17	Matang
Jeruk18	Matang
Jeruk19	Matang
Jeruk20	Matang
Jeruk21	Matang
Jeruk22	Matang
Jeruk23	Matang
Jeruk24	Matang
Jeruk25	Matang
Jeruk26	Matang
Jeruk27	Matang
Jeruk28	Matang
Jeruk29	Matang
Jeruk30	Matang
Jeruk31	Matang
Jeruk32	Matang
Jeruk33	Matang
Jeruk34	Matang
Jeruk35	Matang
Jeruk36	Matang
Jeruk37	Matang
Jeruk38	Matang
Jeruk39	Matang
Jeruk40	Matang
Jeruk41	Matang

Jeruk42	Matang
Jeruk43	Matang
Jeruk44	Matang
Jeruk45	Matang
Jeruk46	Matang
Jeruk47	Matang
Jeruk48	Matang

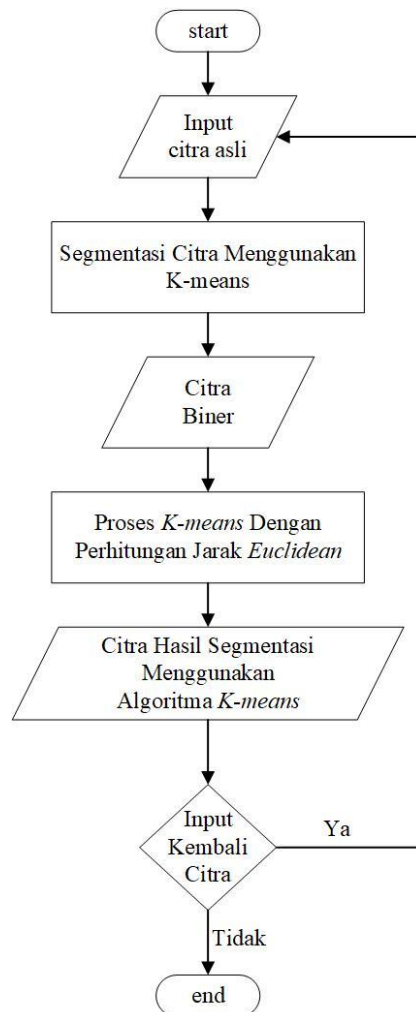
4.1.4. Perancangan Sistem

Pada penelitian ini untuk merancang sistem yang digunakan untuk merancang sebuah sistem ataupun gambaran dari sistem yang akan dibangun, berikut tampilan rancangan sistem pengujian kematangan buah jeruk:

1. Flowchart

Flowchart adalah bagan atau gambar yang menunjukkan aliran proses dan hubungan dari suatu program. *Flowchart* dibutuhkan untuk menjelaskan alur program yang dibuat dalam bentuk grafis agar orang lain dapat memahami alur yang telah dibuat. Rancangan *flowchart* yang akan digunakan untuk identifikasi kematangan buah jeruk terdapat pada gambar di bawah ini :

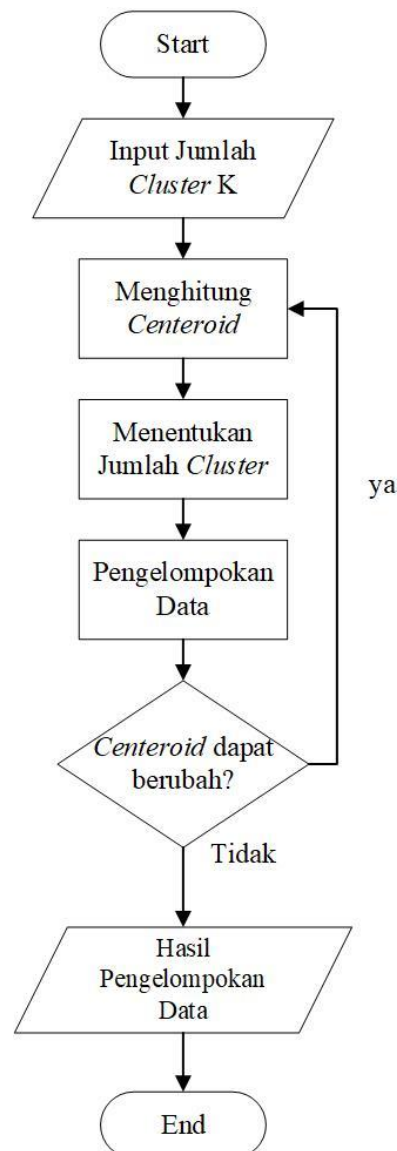
a. Perancangan *Flowchart* Sistem



Gambar 4.9 Flowchart Sistem

Gambar di atas merupakan alur dari *Flowchart* sistem dalam melakukan identifikasi kematangan buah jeruk menggunakan algoritma *k-means*. Dimulai dari menginput citra asli, kemudian proses segmentasi warna dengan metode *threshold* menggunakan algoritma *k-means*. Setelah didapatkan hasil segmentasi warna yang menghasilkan citra *grayscale*. Selanjutnya citra *grayscale* akan diidentifikasi dengan jarak *euclidean* menggunakan algoritma *k-means* sebagai acuan untuk kematangan buah jeruk, setelah didapat hasil dari perhitungan jarak *euclidean*, maka proses algoritma *k-means* selesai.

b. *flowchart* Algoritma *K-means*

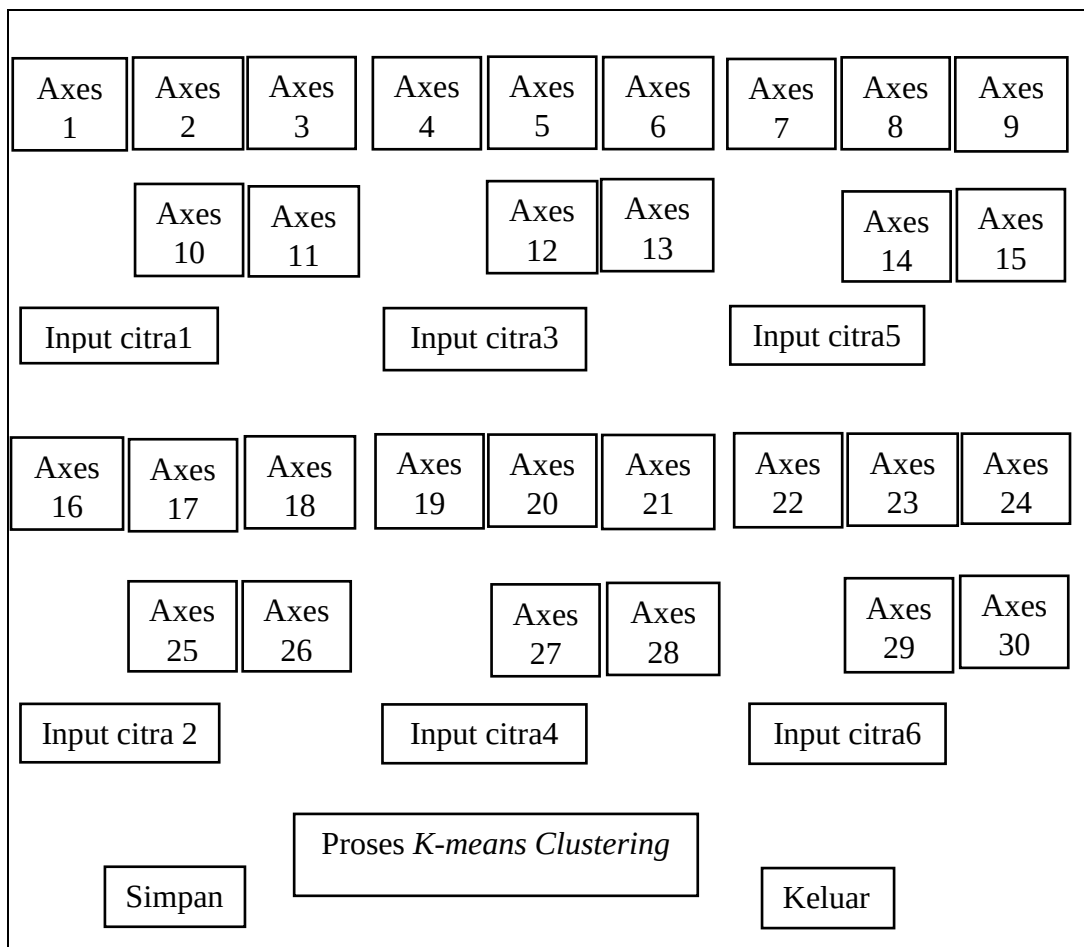


Gambar 4.10 Flowchart Algoritma *K-means*

Gambar di atas merupakan alur dari *Flowchart Algoritma K-means* dalam melakukan pengclusteran dataset buahjeruk untuk menentukan kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means*. Dimulai dari menentukan jumlah *cluster* *K*, kemudian asumsikan titik pusat *cluster*. Selanjutnya menghitung jarak objek dengan pusat cluster dan kemudian mengelompokkan objek berdasarkan jarak minimum, jika ada objek yang berpindah maka kembali ke langkah menghitung jarak objek dengan pusat cluster, jika tidak ada objek yang berpindah maka proses selesai.

2. Perancangan Antar Muka

Perancangan antar muka, sistem ini dirancang dengan menggunakan bahasa pemrograman Matlab. Perancangan antar muka bertujuan untuk memudahkan pengguna (*user*) dalam menggunakan sistem yang telah dibuat. Perancangan antar muka ini akan berpengaruh pada spesifikasi komputer yang digunakan, agar bisa berjalan dengan baik maka spesifikasi hardware harus sesuai. Adapun bentuk rancangan antar muka ini digunakan untuk menginput



gambar citra buah jeruk yang akan diproses menggunakan segmentasi menggunakan algoritma *k-means*.

Gambar 4.11 Perancangan Antar Muka

3. Algoritma Pseudocode

Pseudocode merupakan deskripsi tingkat tinggi informal dan ringkas atas algoritma pemrograman komputer yang menggunakan konvensi struktural atas suatu bahasa pemrograman dan ditujukan untuk dibaca oleh manusia dan bukan oleh mesin.

a. Pseudocode menghitung jarak *euclidean*

Program_menghitung_jarak_euclidean_menggunakan_algoritma_k-means;

Deklarasi:

n: class: area;

Deskripsi:

For n < - 1 : numberofclasses

class (:, :, n) < - classimage < - < - n;

area (n) < - sum (sum (class (:, :, n)));

Write (area);

Read (area);

[~, min_area] < - min (area);

Write (area);

b. Pseudocode menghitung *k-means clustering*

Program_menghitung_k-means_clustering;

Matang untuk nilai > 127, mentah untuk nilai < 127;

Deklarasi :

Nilai_clus : nilai_maks : nilai_min : nilai_threshold : ctr_hsl_seg_algo_kmeans : a
: ctr_hsl_kmeans_clus

Deskripsi:

Nilai_clus < - 124 * 165;


```

Nilai_maks < - 0-255;

Nilai_min < - 0-255;

Nilai_threshold < - (nilai_maks + nilai_min) / 2;

ctr_hsl_seg_algo_kmeans < - int2str nilai_threshold;

a < - T;

ctr_hsl_kmeans_clus < - strcat (a, ctr_hsl_seg_algo_kmeans);

Read (ctr_hsl_kmeans_clus);

If (nilai_threshold > 127) and ctr_hsl_kmeans_clus “matang”;
Else (nilai_threshold < 127) and ctr_hsl_kmeans_clus “mentah”;
EndIf : ctr_hsl_kmeans_clus error;

Write : ctr_hsl_kmeans_clus;

```

c. Pseudocode menghitung rata-rata *k-means clustering*

Program_menghitung_rata-rata_hasil_k-means_clustering;

Matang untuk nilai > 3, mentah untuk nilai <= 3;

Deklarasi:

```

ctr_hsl_kmeans_clus_1 : ctr_hsl_kmeans_clus_2 : ctr_hsl_kmeans_clus_3 :
ctr_hsl_kmeans_clus_4 : ctr_hsl_kmeans_clus_5 : ctr_hsl_kmeans_clus_6 :
tot_nilai;

```

Deskripsi:

```

ctr_hsl_kmeans_clus_1 < - mentah;

ctr_hsl_kmeans_clus_2 < - matang;

ctr_hsl_kmeans_clus_3 < - matang;

ctr_hsl_kmeans_clus_4 < - matang;

```

```

ctr_hsl_kmeans_clus_5 < - matang;

ctr_hsl_kmeans_clus_6 < - mentah;

tot_nilai < - ctr_hsl_kmeans_clus_1 + ctr_hsl_kmeans_clus_2 +
ctr_hsl_kmeans_clus_3 + ctr_hsl_kmeans_clus_4 + ctr_hsl_kmeans_clus_5 +
ctr_hsl_kmeans_clus_6;

Read (tot_nilai);

If (tot_nilai > 3) and ctr_hsl_kmeans_clus “matang”;

Else (tot_nilai <= 3) and ctr_hsl_kmeans_clus “mentah”;

EndIf : tot_nilai error;

Write : tot_nilai;

```

4.2. Hasil

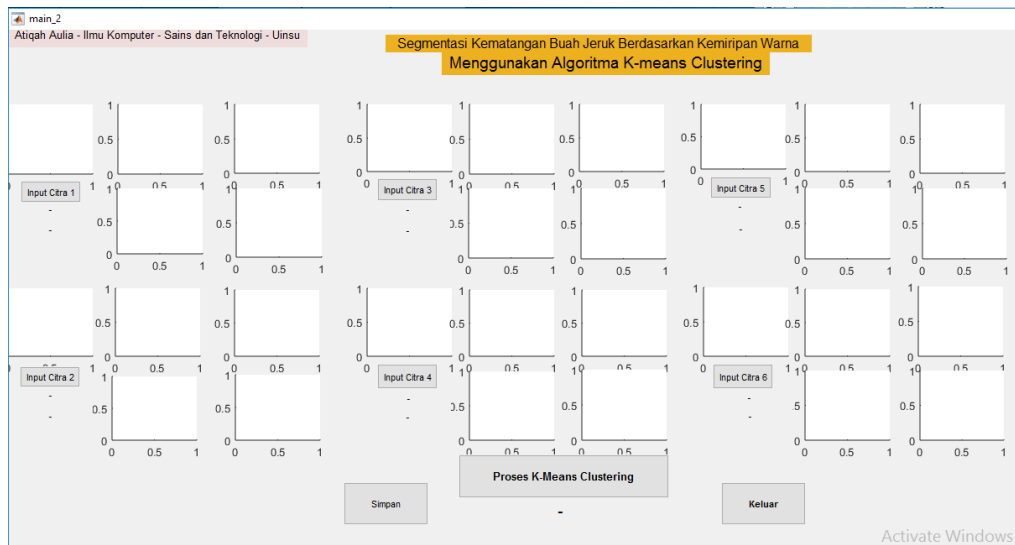
Beberapa tahapan yang akan dibahas mengenai hasil yang didapat pada penelitian ini yaitu pengujian dan penerapan sebagai berikut.

4.2.1. Pengujian

Setelah dilakukan perancangan dan pembuatan sebuah sistem, maka selanjutnya dilakukan pengujian. Pengujian bertujuan untuk melihat sejauh mana sistem yang telah dibangun apakah sudah sesuai dengan yang diharapkan, contoh hasil segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means* dapat dilihat sebagai berikut ini:

1. Tampilan awal aplikasi

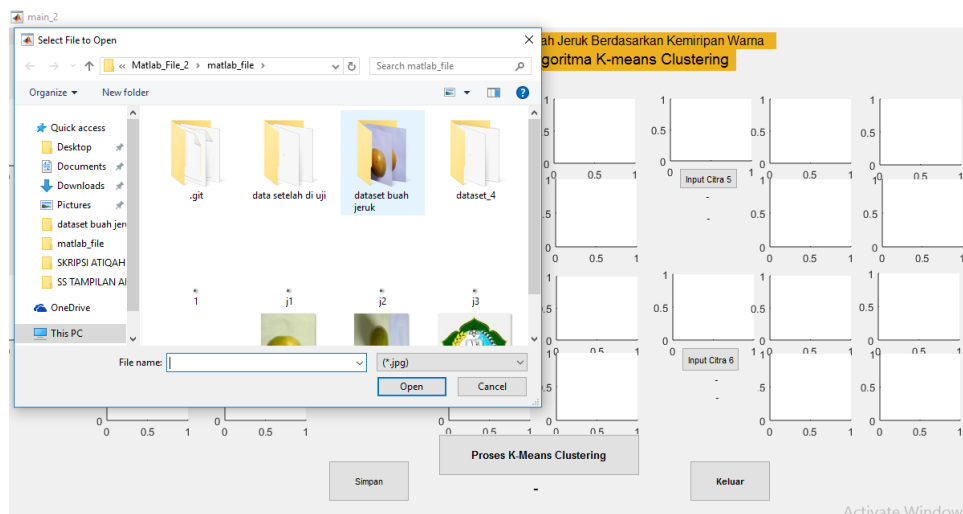
Tampilan dibawah ini adalah tampilan aplikasi segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means*. dimana terdapat button input citra 1 sampai button input citra 6 yang akan diproses dengan algoritma *k-means*. Fungsi input citra ini untuk mengetahui kematangan buah jeruk dari 6 sudut. Selanjutnya terdapat button proses *k-means clustering*, button simpan dan button keluar. Seperti gambar 4.12 tampilan aplikasi di bawah ini.



Gambar 4.12Tampilan Aplikasi

2. Tampilan Inputan Citra

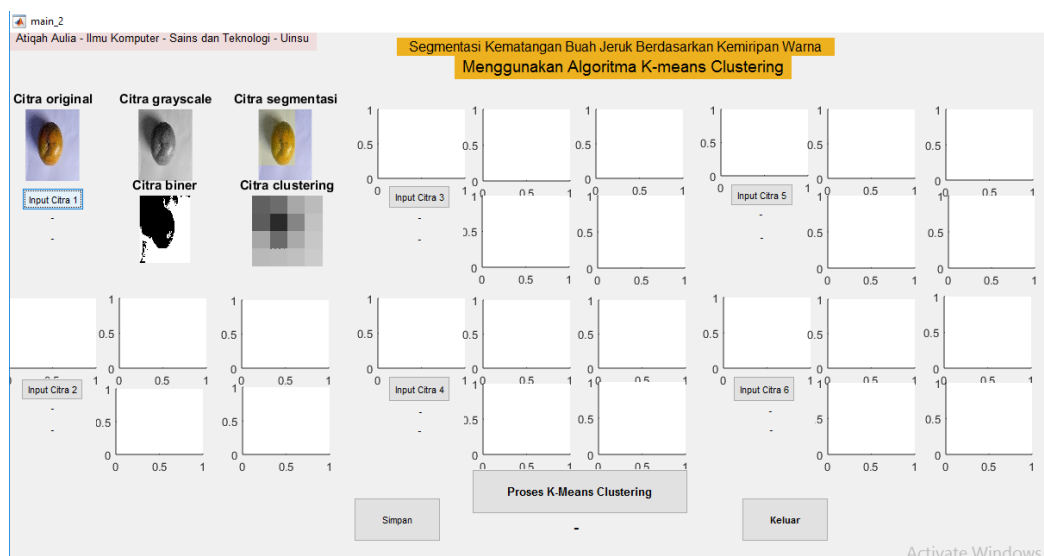
Setelah itu, klik button input citra 1 ataupun button input citra 2 sampai button input citra 6 dari file dataset buah jeruk untuk mengambil gambar citra dari sudut atas, bawah, kanan, kiri, depan dan belakang, dengan nama file jeruk1 *.jpg sampai jeruk48 *.jpg dataset dengan ukuran citra 124 x 165 *pixel*. Setelah itu klik open seperti gambar dibawah ini.



Gambar 4.13Tampilan Inputan Citra

3. Tampilan input citra 1

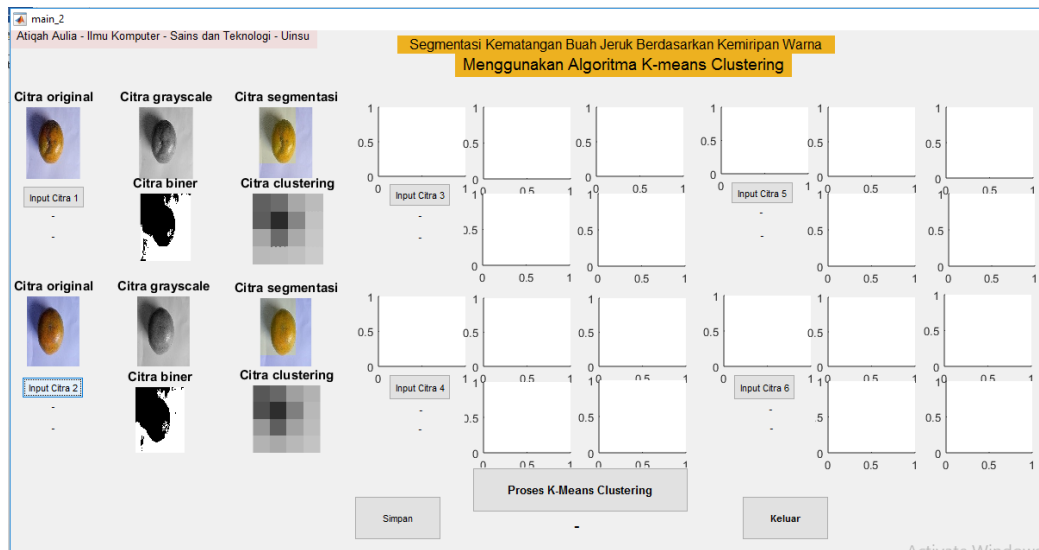
Berikut tampilan input citra 1, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *clusterkan* berdasarkan kematangannya.



Gambar 4.14Tampilan input citra 1

4. Tampilan input citra 2

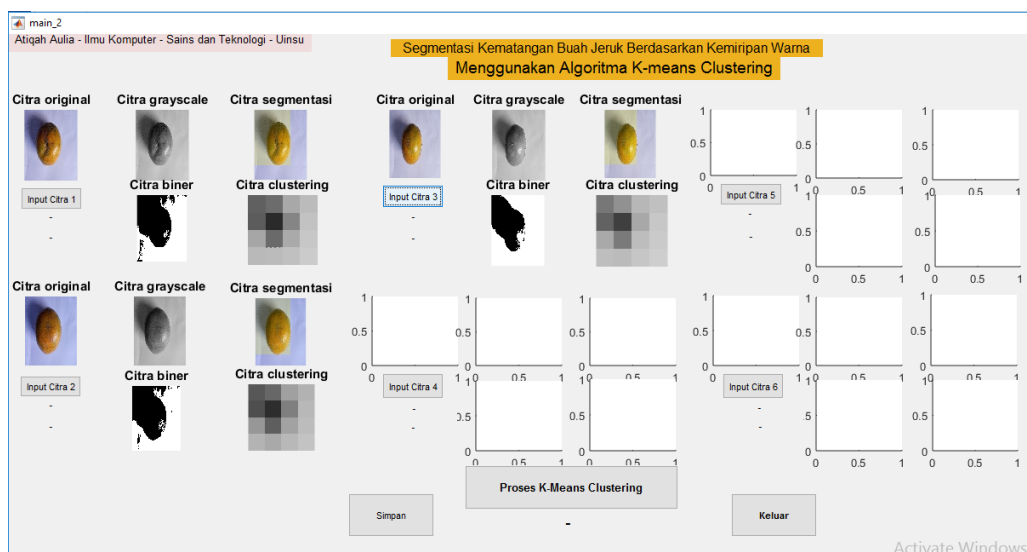
Selanjutnya tampilan input citra 2, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *clusterkan* berdasarkan kematangannya.



Gambar 4.15Tampilan input citra 2

5. Tampilan input citra 3

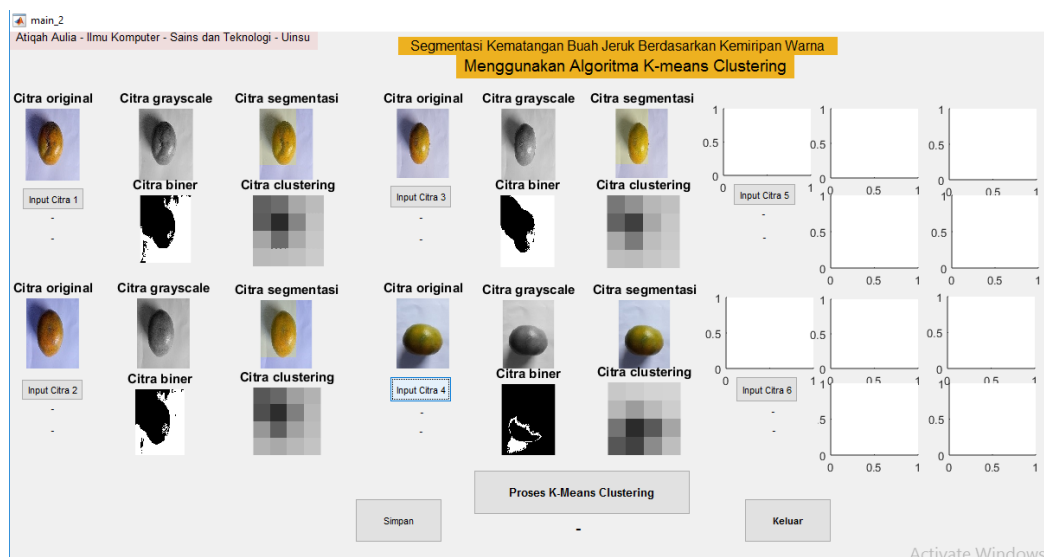
Selanjutnya tampilan input citra 3, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *clusterkan* berdasarkan kematangannya.



Gambar 4.16Tampilan input citra 3

6. Tampilan input citra 4

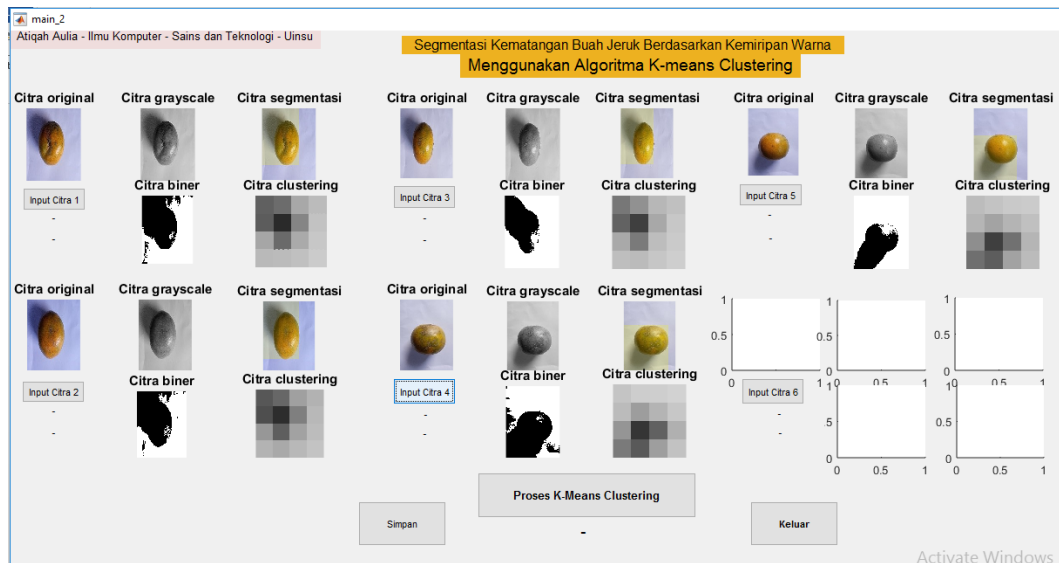
Selanjutnya tampilan input citra 4, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *clusterkan* berdasarkan kematangannya.



Gambar 4.17 Tampilan input citra 4

7. Tampilan input citra 5

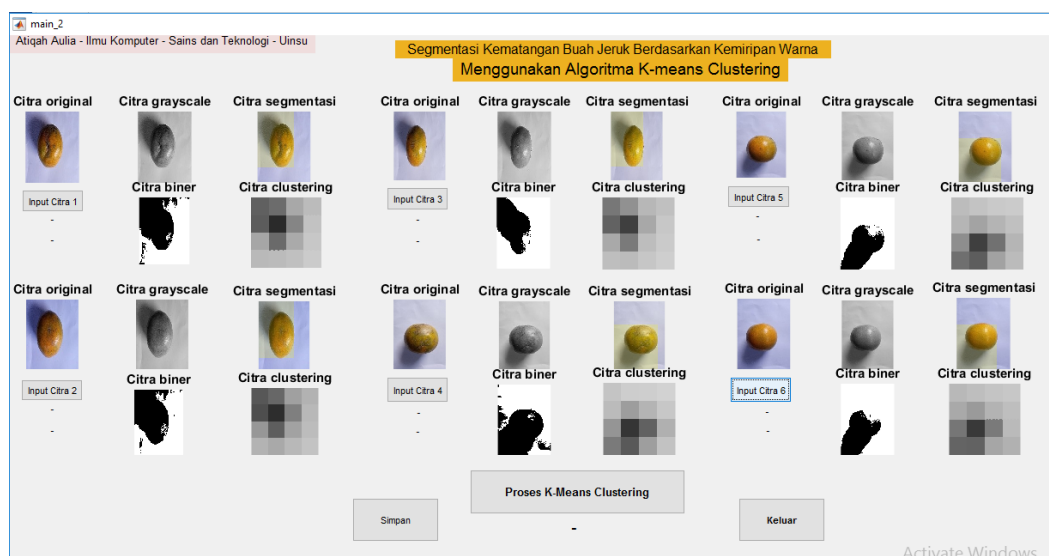
Selanjutnya tampilan input citra 5, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *clusterkan* berdasarkan kematangannya.



Gambar 4.18Tampilan input citra 5

8. Tampilan input citra 6

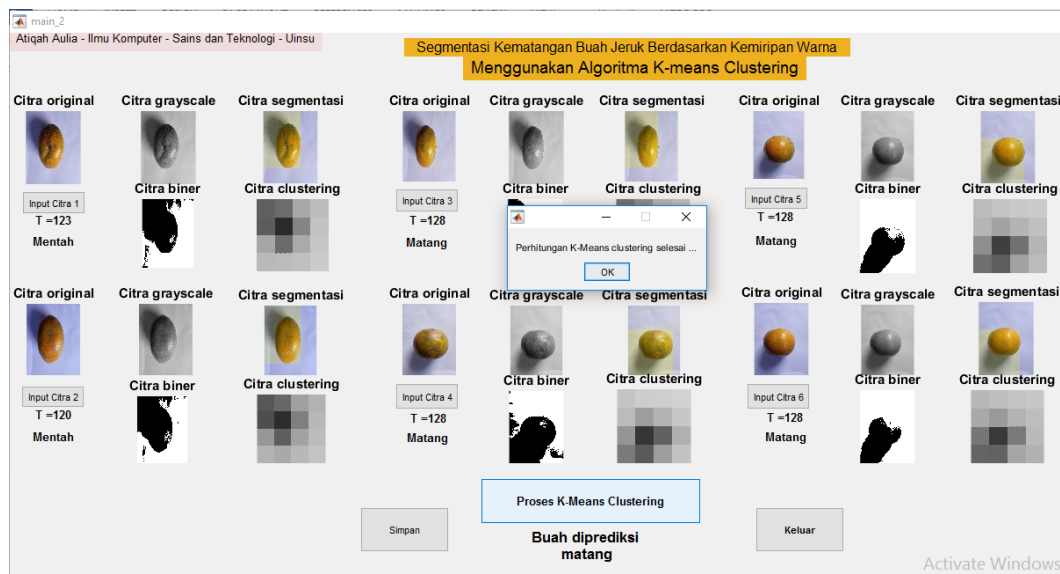
Selanjutnya tampilan input citra 6, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *cluster* berdasarkan kematangannya.



Gambar 4.19Tampilan input citra 6

9. Tampilan proses *k-means clustering*

Berikut tampilan proses *k-means clustering* buah jeruk yang diukur dari 6 sudut. Sudut atas, bawah, kanan, kiri, depan dan belakang, dengan nama file jeruk1 *.jpg sampai jeruk6 *.jpg dengan ukuran citra 124 x 165 pixel, nilai T masing-masing didapatkan secara otomatis dari sistem proses segmentasi menggunakan algoritma *k-means* yang menyatakan bahwa buah jeruk diprediksi matang ataupun mentah dari masing-masing input citra. Selanjutnya pada proses *k-means clustering* proses inputan citra 1 sampai inputan citra 6 akan disatukan, apabila nilai matang lebih besar dari 3 maka dinyatakan matang. Akan tetapi, apabila nilai matang lebih kecil atau sama dengan 3 maka dinyatakan mentah. Setelah hasil citra dari 6 sudut diproses dengan proses *k-means clustering*, klik button simpan untuk menyimpan gambar proses *k-means clustering*, dan selanjutnya klik button keluar jika ingin keluar maka proses *k-means clustering* selesai. Seperti gambar 4.2.1.9. Tampilan proses *k-means clustering*.



Gambar 4.20 Tampilan proses *k-means clustering*

10. Hasil Pengujian sistem terhadap masing-masing citra buah jeruk dengan algoritma *k-means*

Dibawah ini merupakan tabel hasil pengujian terhadap masing-masing citra buah jeruk dengan format *.jpg dan ukuran 124 x 165 *pixel*, dan nilai segmentasi menggunakan *k-means* didapatkan secara otomatis dari sistem.

Tabel 4.7 Hasil Pengujian

Dataset	nilai segmentasi menggunakan <i>k-means</i> (T)	Hasil proses <i>k-means</i>	Hasil akhir proses <i>k-means</i>
Jeruk1	124	Mentah	Matang
Jeruk2	128	Matang	
Jeruk3	128	Matang	
Jeruk4	128	Matang	
Jeruk5	128	Matang	
Jeruk6	122	Mentah	
Jeruk7	125	Mentah	Matang
Jeruk8	128	Matang	
Jeruk9	128	Matang	
Jeruk10	128	Matang	
Jeruk11	128	Matang	
Jeruk12	128	Matang	
Jeruk13	128	Matang	
Jeruk14	128	Matang	

Jeruk15	128	Matang	Matang
Jeruk16	128	Matang	
Jeruk17	127	Matang	
Jeruk18	128	Matang	
Jeruk19	126	Mentah	Matang
Jeruk20	128	Matang	
Jeruk21	126	Mentah	
Jeruk22	128	Matang	
Jeruk23	128	Matang	
Jeruk24	128	Matang	
Jeruk25	124	Mentah	Matang
Jeruk26	128	Matang	
Jeruk27	128	Matang	
Jeruk28	128	Matang	
Jeruk29	128	Matang	
Jeruk30	128	Matang	
Jeruk31	123	Mentah	
Jeruk32	120	Mentah	
Jeruk33	128	Matang	

Jeruk34	128	Matang	Matang
Jeruk35	128	Matang	
Jeruk36	128	Matang	
Jeruk37	125	Mentah	Mentah
Jeruk38	128	Matang	
Jeruk39	128	Matang	
Jeruk40	123	Mentah	
Jeruk41	125	Mentah	
Jeruk42	125	Mentah	
Jeruk43	120	Mentah	Mentah
Jeruk44	123	Mentah	
Jeruk45	125	Mentah	
Jeruk46	124	Mentah	
Jeruk47	126	Mentah	
Jeruk48	125	Mentah	

Berdasarkan hasil pengujian segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means* hasil yang diperoleh dari proses tahapan-tahapan yang dilakukan yaitu menggunakan dataset 8 buah jeruk yang diambil dari 6 sudut. Sudut atas, bawah, kanan, kiri, depan dan belakang dan menghasilkan gambar sebanyak 48 dataset yang akan diuji. Dengan format *.jpg dan ukuran 124 x 165pixel. Nilai T masing-masing didapatkan secara

otomatis dari sistem, selanjutnya akan diproses dengan *k-means clustering* pada masing-masing input citra 1 sampai input citra 6, dari input citra 1 sampai input citra 6 akan disatukan, apabila nilai matang lebih besar dari 3 maka dinyatakan matang. Akan tetapi, apabila nilai matang lebih kecil atau sama dengan 3 maka dinyatakan mentah. Hasil yang didapatkan dari dataset 8 buah jeruk yang diukur dari 6 sudut menghasilkan dataset sebanyak 48 yaitu data mentah berjumlah 2 dan data matang berjumlah 6.

4.2.2. Penerapan

Penerapan/ penggunaan sistem ini untuk mengetahui tingkat kematangan buah jeruk. Dengan menggunakan segmentasi citra dan menerapkan algoritma *k-means* bermanfaat untuk membantu mahasiswa maupun pelajar melihat tingkat kematangan buah jeruk. Penerapan sistem ini dapat diterapkan oleh mahasiswa maupun pelajar yang ingin mengetahui tingkat kematangan buah jeruk. Yang mana sistem ini akan mempermudah mahasiswa maupun pelajar dalam menganalisis buah jeruk sehingga kematangan buah jeruk dapat diketahui dengan cara mudah dan dapat meminimalisir kesalahan dalam memilih kematangan buah jeruk.

BAB V

KESIMPULAN DAN SARAN

5.1. KESIMPULAN

Berdasarkan penelitian yang dilakukan penulis dalam proses segmentasi kematangan buah jeruk menggunakan algoritma *k-means*, dapat disimpulkan sebagai berikut.

1. Dari hasil penerapan segmentasi citra, penelitian yang dilakukan penulis dalam menganalisis tingkat kematangan buah jeruk berdasarkan kemiripan warna menghasilkan kualitas citra yang lebih baik dari sebelumnya.
2. Dalam proses identifikasi kematangan buah jeruk berdasarkan kemiripan warna dengan menerapkan algoritma *k-means* menghasilkan kematangan sebanyak 6 dan mentah sebanyak 2.
3. Sistem yang dibuat dalam melakukan segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means* dapat digunakan dalam pemilihan tingkat kematangan dari buah jeruk.

5.2. SARAN

Adapun saran penulis dalam penelitian ini sebagai berikut.

1. Penulis menyadari banyak kesalahan dalam penulisan maupun pembuatan program aplikasi.
2. Penulis berharap agar pembuatan sistem aplikasi ini dapat dikembangkan, agar petani maupun orang awam dapat menggunakannya bukan hanya mahasiswa maupun pelajar.
3. Penulis menyadari dalam pengambilan dataset buah jeruk masih kurang akurat, sehingga penulis menyarankan agar pada penelitian selanjutnya pengambilan dataset lebih akurat lagi.

DAFTAR PUSTAKA

- Anharku, 2009, *flowchart. Ilmu komputer.org*, 1-4.
- Aprianto,B.W., 2018, Sistem Pengelompokan dan Rekomendasi Pengadaan Barang Dengan Menggunakan Metode *K-means*. *Simki-Techsain*, **vol 02**.
- Arief,M., 2019, Klasifikasi Kematangan Buah Jeruk Berdasarkan Fitur Warna Menggunakan Metode SVM. *Jurnal Ilmu Komputer dan Desain Komunikasi Visual*, **vol 4**, 9-16.
- B,M.M., Andriana., Hidayat.A.S., 2017, Implementasi Algoritma GLCM Dan MED pada Aplikasi Pendeteksi Kolesterol Melalui Iris Mata. *MIND Journal*, Vol. 2, halm.23 – 41.
- Darmi,Y., Setiawan,A., 2016, Penerapan Metode *Clustering K-Means* Dalam Pengelompokan Penjualan Produk. *Jurnal Media Infotama*,**Vol. 12**, 148-157.
- Eliyani., Tulus., Fahmi.F., 2013, Pengenalan Tingkat Kematangan Buah Pepaya Paya Rabo Menggunakan Pengolahan Citra Berdasarkan Warna RGB Dengan K-Means Clustering. *SINGUDA ENSIKOM*.
- Febrinanto,F.G., Dewi,C., Wiratno,A.T., 2018,Implementasi Algoritme *K-Means*Sebagai Metode Segmentasi Citra Dalam Identifikasi Penyakit Daun Jeruk. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, **Vol. 2**, 5375-5383.
- Furqan,M., Sriani., Siregar,Y.K., 2020, Perbandingan Algoritma *Contraharmonic Mean Filter* Dan *Arithmetic Mean Filter* Untuk Mereduksi *Exponential Noise*. *JISKa*,**Vol.5**, 107 – 115.
- Furqan,M., Sriani., Sari,I.E.Y., 2020,Penerapan Metode Otsu Dalam Melakukan Segmentasi Citra Pada Citra Naskah Arab.*Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, **Vol.20**, 59-72.
- Furqan,M., Sriani., Harahap,L.S., 2020,Klasifikasi Daun Bugenvil Menggunakan *Gray Level Co-Occurrence Matrix* Dan *K- Nearest Neighbor*. *Jurnal CoreIT*, **Vol.6**.

- Hadiani,S., Riana,D., 2018, Segmentasi Citra Bemisia Tabaci Menggunakan Metode *K-Means*. *Seminar Nasional Inovasi dan Tren (SNIT)*, 2018.
- Hutagalung,S.N., 2018, Pembelajaran Fisika Dasar Dan Elektronika Dasar Menggunakan Aplikasi Matlab Metode Simulink. *Journal of Science and Social Research*, **Vol 4307**, 30-35.
- Kusuma,I.W.A.W, dan Ellyana,R.L, 2018, Penerapan Citra Terkompresi Pada Segmentasi Citra Menggunakan Algoritme K-MEANS.*JUTEI*,**Vol 2**, 65-74.
- Munir,R., 2004, *Pengolahan Citra Digital*. Informatika: Bandung.
- Orisa,M., Hidayat,T., 2019, Analisis Teknik Segmentasi Pada Pengolahan Citra. *Jurnal MNEMONIC*, **Vol 2**, 9-13.
- Prabowo,H., 2017, Deteksi Kondisi Kematangan Buah Jeruk Berdasarkan Kemiripan Warna Pada Ruang Warna Rgb Berbasis Android. *Jurnal Elektronik Sistem Informasi dan Komputer*,**Vol 3**.
- Putra,D., 2010, *Pengolahan Citra Digital*. Andi Offset: Yogyakarta.
- Siem,P., Siem,G., Siem,L., 1952, Jeruk (*Citrus Sp.*). *Ttg Budidaya Pertanian*, 1-16.
- Sriani., Triase., Khairuna., 2017, Pendekomposisian Citra Digital Dengan Algoritma Dwt. *Jurnal Ilmu Komputer dan Informatika*, **Vol 01**.
- <http://studyshut.blogspot.com/2018/12/contoh-perhitungan-manual-menggunakan.html>

LAMPIRAN-LAMPIRAN

LAMPIRAN TOOLS MATLAB

Tools Matlab untuk Segmentasi Kematangan Buah Jeruk Berdasarkan Kemiripan Warna Menggunakan Algoritma *K-means*:

Tools	Fungsi
Img_original_1	Menampilkan citra original
Img_original_2	Menampilkan citra original
Img_original_3	Menampilkan citra original
Img_original_4	Menampilkan citra original
Img_original_5	Menampilkan citra original
Img_original_6	Menampilkan citra original
Img_grayscale_1	Menampilkan citra grayscale
Img_grayscale_2	Menampilkan citra grayscale
Img_grayscale_3	Menampilkan citra grayscale
Img_grayscale_4	Menampilkan citra grayscale
Img_grayscale_5	Menampilkan citra grayscale
Img_grayscale_6	Menampilkan citra grayscale
Img_dec_1	Menampilkan citra biner
Img_dec_2	Menampilkan citra biner
Img_dec_3	Menampilkan citra biner
Img_dec_4	Menampilkan citra biner
Img_dec_5	Menampilkan citra biner
Img_dec_6	Menampilkan citra biner
Img_seg_1	Menampilkan citra segmentasi
Img_seg_2	Menampilkan citra segmentasi
Img_seg_3	Menampilkan citra segmentasi
Img_seg_4	Menampilkan citra segmentasi

Img_seg_5	Menampilkan citra segmentasi
Img_seg_6	Menampilkan citra segmentasi
Citra_clus_1	Menampilkan citra <i>clustering</i>
Citra_clus_2	Menampilkan citra <i>clustering</i>
Citra_clus_3	Menampilkan citra <i>clustering</i>
Citra_clus_4	Menampilkan citra <i>clustering</i>
Citra_clus_5	Menampilkan citra <i>clustering</i>
Citra_clus_6	Menampilkan citra <i>clustering</i>
Btn_input_citra_1	Menampilkan citra original, citra grayscale, citra segmentasi, citra biner, citra clustering
Btn_input_citra_2	Menampilkan citra original, citra grayscale, citra segmentasi, citra biner, citra clustering
Btn_input_citra_3	Menampilkan citra original, citra grayscale, citra segmentasi, citra biner, citra clustering
Btn_input_citra_4	Menampilkan citra original, citra grayscale, citra segmentasi, citra biner, citra clustering
Btn_input_citra_5	Menampilkan citra original, citra grayscale, citra segmentasi, citra biner, citra clustering
Btn_input_citra_6	Menampilkan citra original, citra grayscale, citra segmentasi, citra biner, citra clustering
Btnproses	Menampilkan proses k-means clustering
Btn_save	Menampilkan simpan
Btn_keluar	Menampilkan keluar
Txt_t_1	Menampilkan nilai thresholding
Txt_t_2	Menampilkan nilai thresholding
Txt_t_3	Menampilkan nilai thresholding
Txt_t_4	Menampilkan nilai thresholding
Txt_t_5	Menampilkan nilai thresholding
Txt_t_6	Menampilkan nilai thresholding
Txt_hasil_1	Menampilkan hasil proses <i>k-means clustering</i>

Txt_hasil_2	Menampilkan hasil proses <i>k-means clustering</i>
Txt_hasil_3	Menampilkan hasil proses <i>k-means clustering</i>
Txt_hasil_4	Menampilkan hasil proses <i>k-means clustering</i>
Txt_hasil_5	Menampilkan hasil proses <i>k-means clustering</i>
Txt_hasil_6	Menampilkan hasil proses <i>k-means clustering</i>
Txt_hasil_final	Menampilkan hasil keseluruhan 6 sudut proses <i>k-means clustering</i>

LAMPIRAN

LISTING PROGRAM

```
function varargout = main_2(varargin)
% MAIN_2 MATLAB code for main_2.fig
%     MAIN_2, by itself, creates a new MAIN_2 or raises the
existing
%     singleton*.
%
%     H = MAIN_2 returns the handle to a new MAIN_2 or the handle
to
%     the existing singleton*.
%
%     MAIN_2('CALLBACK',hObject,eventData,handles,...) calls the
local
%     function named CALLBACK in MAIN_2.M with the given input
arguments.
%
%     MAIN_2('Property','Value',...) creates a new MAIN_2 or
raises the
%     existing singleton*. Starting from the left, property
value pairs are
%     applied to the GUI before main_2_OpeningFcn gets called.
An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to main_2_OpeningFcn via
varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help main_2

% Last Modified by GUIDE v2.5 14-Dec-2020 00:08:11
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @main_2_OpeningFcn, ...
                  'gui_OutputFcn',  @main_2_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
```



```

end
% End initialization code - DO NOT EDIT

% --- Executes just before main_2 is made visible.
function main_2_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to main_2 (see VARARGIN)

% Choose default command line output for main_2
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes main_2 wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = main_2_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in btn_input_citra_1.
function btn_input_citra_1_Callback(hObject, eventdata, handles)
% hObject    handle to btn_input_citra_1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
warning('off');
% citra 1
% -> input citra
[filename,pathname] = uigetfile('*.jpg');
img_input_1 = imread(fullfile(pathname,filename));
% -> set gambar input original ke axes
axes(handles.img_original_1);
imshow(img_input_1);
title('Citra original');
% -> ubah gambar ke grayscale
img_gray_1 = rgb2gray(img_input_1);
saveas(gcf, 'citra_uji1.png');
% -> set gambar grayscale ke axes

```

```

axes(handles.img_grayscale_1);
imshow(img_gray_1);
title('Citra grayscale');

% -> tentukan jumlah class
numberOfClasses = 2;
indexes = kmeans(img_gray_1(:), numberOfClasses);
classImage = reshape(indexes, size(img_gray_1));

% h = subplot(2, 2, 3);
axes(handles.img_seg_1);
imshow(classImage, []);
title('Citra biner');
% colormap(h,parula);
class = zeros(size(img_gray_1));
area = zeros(numberOfClasses, 1);

for n = 1:numberOfClasses
    class(:, :, n) = classImage == n;
    area(n) = sum(sum(class(:, :, n)));
end

[~, min_area] = min(area);

object = classImage == min_area;
bw = medfilt2(object, [5 5]);
bw = bwareaopen(bw, 5000);
s = regionprops(bw, 'BoundingBox');
bbox = cat(1, s.BoundingBox);
RGB = insertShape(img_input_1, 'FilledRectangle', bbox, 'Color',
    'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,
    '', 'TextBoxOpacity', 0.9, 'FontSize', 6);
axes(handles.img_dec_1);
imshow(RGB);
title('Citra segmentasi');

% clusterisasi
img_clus_1 = imresize(img_gray_1, [4 4]);
axes(handles.citra_clus_1);
imshow(img_clus_1);
title('Citra clustering');
handles.np1 = img_clus_1;
guidata(hObject, handles);

% --- Executes during object creation, after setting all
properties.
function figure1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to figure1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called
clc;clear;

```

```

% --- Executes on button press in btn_input_citra_2.
function btn_input_citra_2_Callback(hObject, eventdata, handles)
% hObject      handle to btn_input_citra_2 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
warning('off');
% citra 2
% -> input citra
[filename,pathname] = uigetfile('*.jpg');
img_input_2 = imread(fullfile(pathname,filename));
% -> set gambar input original ke axes
axes(handles.img_original_2);
imshow(img_input_2);
title('Citra original');
% -> ubah gambar ke grayscale
img_gray_2 = rgb2gray(img_input_2);
% -> set gambar grayscale ke axes
axes(handles.img_grayscale_2);
imshow(img_gray_2);
title('Citra grayscale');

% -> tentukan jumlah class
numberOfClasses = 2;
indexes = kmeans(img_gray_2(:), numberOfClasses);
classImage = reshape(indexes, size(img_gray_2));

% h = subplot(2, 2, 3);
axes(handles.img_seg_2);
imshow(classImage, []);
title('Citra biner');
% colormap(h,parula);
class = zeros(size(img_gray_2));
area = zeros(numberOfClasses, 1);

for n = 1:numberOfClasses
    class(:, :, n) = classImage == n;
    area(n) = sum(sum(class(:, :, n)));
end

[~,min_area] = min(area);

object = classImage==min_area;
bw = medfilt2(object,[5 5]);
bw = bwareaopen(bw, 5000);
s = regionprops(bw,'BoundingBox');
bbox = cat(1, s.BoundingBox);
RGB = insertShape(img_input_2, 'FilledRectangle', bbox, 'Color',
'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,
'', 'TextBoxOpacity', 0.9, 'FontSize', 11);
axes(handles.img_dec_2);
imshow(RGB);
title('Citra segmentasi');

```

```

% clusterisasi
img_clus_2 = imresize(img_gray_2, [4 4]);
axes(handles.citra_clus_2);
imshow(img_clus_2);
title('Citra clustering');
handles.np2 = img_clus_2;
guidata(hObject, handles);

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

% pertama
np1 = handles.np1;
m1 = max(np1(:));
k1 = min(np1(:));
fnx_1 = (m1 + k1) / 2;
mts1 = int2str(fnx_1);
cp1 = 'T = ';
fincap1 = strcat(cp1, mts1);
set(handles.txt_t_1, 'String', fincap1);
if fnx_1 < 127
    set(handles.txt_hasil_1, 'String', 'Mentah');
    handles.nf1 = 0;
else
    set(handles.txt_hasil_1, 'String', 'Matang');
    handles.nf1 = 1;
end

```



```

% kedua
np2 = handles.np2;
m2 = max(np2(:));
k2 = min(np2(:));
fnx_2 = (m2 + k2) / 2;
mts2 = int2str(fnx_2);
cp2 = 'T = ';
fincap2 = strcat(cp2, mts2);
set(handles.txt_t_2, 'String', fincap2);
if fnx_2 < 127
    set(handles.txt_hasil_2, 'String', 'Mentah');
    handles.nf2 = 0;
else
    set(handles.txt_hasil_2, 'String', 'Matang');
    handles.nf2 = 1;
end

% ketiga
np3 = handles.np3;
m3 = max(np3(:));
k3 = min(np3(:));
fnx_3 = (m3 + k3) / 2;
mts3 = int2str(fnx_3);
cp3 = 'T = ';

```

```

fincap3 = strcat(cp3, mts3);
set(handles.txt_t_3, 'String', fincap3);
if fnx_3 < 127
    set(handles.txt_hasil_3, 'String', 'Mentah');
    handles.nf3 = 0;
else
    set(handles.txt_hasil_3, 'String', 'Matang');
    handles.nf3 = 1;
end

```

```

% keempat
np4 = handles.np4;
m4 = max(np4(:));
k4 = min(np4(:));
fnx_4 = (m4 + k4) / 2;
mts4 = int2str(fnx_4);
cp4 = 'T = ';
fincap4 = strcat(cp4, mts4);
set(handles.txt_t_4, 'String', fincap4);
if fnx_4 < 127
    set(handles.txt_hasil_4, 'String', 'Mentah');
    handles.nf4 = 0;
else
    set(handles.txt_hasil_4, 'String', 'Matang');
    handles.nf4 = 1;
end

```

```

% kelima
np5 = handles.np5;
m5 = max(np5(:));
k5 = min(np5(:));
fnx_5 = (m5 + k5) / 2;
mts5 = int2str(fnx_5);
cp5 = 'T = ';

```

```

fincap5 = strcat(cp5, mts5);
set(handles.txt_t_5, 'String', fincap5);
if fnx_5 < 127
    set(handles.txt_hasil_5, 'String', 'Mentah');
    handles.nf5 = 0;
else
    set(handles.txt_hasil_5, 'String', 'Matang');
    handles.nf5 = 1;
end

% kedua
np6 = handles.np6;
m6 = max(np6(:));
k6 = min(np6(:));
fnx_6 = (m6 + k6) / 2;
mts6 = int2str(fnx_6);
cp6 = 'T = ';
fincap6 = strcat(cp6, mts6);
set(handles.txt_t_6, 'String', fincap6);
if fnx_6 < 127
    set(handles.txt_hasil_6, 'String', 'Mentah');
    handles.nf6 = 0;

```

```

else
    set(handles.txt_hasil_6, 'String', 'Matang');
    handles.nf6 = 1;
end

nf_1 = handles.nf1;
nf_2 = handles.nf2;
nf_3 = handles.nf3;
nf_4 = handles.nf4;
nf_5 = handles.nf5;
nf_6 = handles.nf6;

f = msgbox('Perhitungan K-Means clustering selesai ...');

total_nilai = nf_1 + nf_2 + nf_3 + nf_4 + nf_5 + nf_6;

if total_nilai <= 3
    set(handles.txt_hasil_final,'String', 'Buah diprediksi
mentah');
else
    set(handles.txt_hasil_final,'String', 'Buah diprediksi
matang');
end

disp(total_nilai);

guidata(hObject, handles);

% --- Executes on button press in btn_input_citra_2.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject      handle to btn_input_citra_2 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
warning('off');
% citra 3
% -> input citra
[filename,pathname] = uigetfile('*.jpg');
img_input_3 = imread(fullfile(pathname,filename));
% -> set gambar input original ke axes
axes(handles.img_original_3);
imshow(img_input_3);
title('Citra original');
% -> ubah gambar ke grayscale
img_gray_3 = rgb2gray(img_input_3);

```

```

% -> set gambar grayscale ke axes
axes(handles.img_grayscale_3);
imshow(img_gray_3);
title('Citra grayscale');

% -> tentukan jumlah class
numberOfClasses = 2;
indexes = kmeans(img_gray_3(:), numberOfClasses);
classImage = reshape(indexes, size(img_gray_3));

% h = subplot(2, 2, 3);
axes(handles.img_seg_3);
imshow(classImage, []);
title('Citra biner');
% colormap(h,parula);
class = zeros(size(img_gray_3));
area = zeros(numberOfClasses, 1);

for n = 1:numberOfClasses
    class(:, :, n) = classImage == n;
    area(n) = sum(sum(class(:, :, n)));
end

[~, min_area] = min(area);

object = classImage == min_area;
bw = medfilt2(object, [5 5]);
bw = bwareaopen(bw, 5000);
s = regionprops(bw, 'BoundingBox');
bbox = cat(1, s.BoundingBox);
RGB = insertShape(img_input_3, 'FilledRectangle', bbox, 'Color',
'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,

```

```

'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,
'', 'TextBoxOpacity', 0.9, 'FontSize', 11);
axes(handles.img_dec_3);
imshow(RGB);
title('Citra segmentasi');

% clusterisasi
img_clus_3 = imresize(img_gray_3, [4 4]);
axes(handles.citra_clus_3);
imshow(img_clus_3);
title('Citra clustering');
handles.np3 = img_clus_3;
guidata(hObject, handles);
% --- Executes on button press in input_citra_4.
function input_citra_4_Callback(hObject, eventdata, handles)
% hObject      handle to input_citra_4 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
warning('off');
% citra 4
% -> input citra
[filename,pathname] = uigetfile('*.jpg');

```

```

img_input_4 = imread(fullfile(pathname,filename));
% -> set gambar input original ke axes
axes(handles.img_original_4);
imshow(img_input_4);
title('Citra original');
% -> ubah gambar ke grayscale
img_gray_4 = rgb2gray(img_input_4);
% -> set gambar grayscale ke axes
axes(handles.img_grayscale_4);
imshow(img_gray_4);
title('Citra grayscale');

% -> tentukan jumlah class
numberOfClasses = 2;
indexes = kmeans(img_gray_4(:), numberOfClasses);
classImage = reshape(indexes, size(img_gray_4));

% h = subplot(2, 2, 3);
axes(handles.img_seg_4);
imshow(classImage, []);
title('Citra biner');
% colormap(h,parula);
class = zeros(size(img_gray_4));
area = zeros(numberOfClasses, 1);

for n = 1:numberOfClasses
    class(:, :, n) = classImage == n;
    area(n) = sum(sum(class(:, :, n)));
end

[~,min_area] = min(area);

object = classImage==min_area;

```

```

bw = medfilt2(object,[5 5]);
bw = bwareaopen(bw, 5000);
s = regionprops(bw,'BoundingBox');
bbox = cat(1, s.BoundingBox);
RGB = insertShape(img_input_4, 'FilledRectangle', bbox, 'Color',
'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,
'', 'TextBoxOpacity', 0.9, 'FontSize', 11);
axes(handles.img_dec_4);
imshow(RGB);
title('Citra segmentasi');

% clusterisasi
img_clus_4 = imresize(img_gray_4, [4 4]);
axes(handles.citra_clus_4);
imshow(img_clus_4);
title('Citra clustering');
handles.np4 = img_clus_4;
guidata(hObject, handles);

% --- Executes on button press in input_citra_5.
function input_citra_5_Callback(hObject, eventdata, handles)

% hObject      handle to input_citra_5 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
warning('off');
% citra 5
% -> input citra
[filename,pathname] = uigetfile('*.jpg');
img_input_5 = imread(fullfile(pathname,filename));
% -> set gambar input original ke axes
axes(handles.img_original_5);
imshow(img_input_5);
title('Citra original');
% -> ubah gambar ke grayscale
img_gray_5 = rgb2gray(img_input_5);
% -> set gambar grayscale ke axes
axes(handles.img_grayscale_5);
imshow(img_gray_5);
title('Citra grayscale');

% -> tentukan jumlah class
numberOfClasses = 2;
indexes = kmeans(img_gray_5(:), numberOfClasses);
classImage = reshape(indexes, size(img_gray_5));

% h = subplot(2, 2, 3);
axes(handles.img_seg_5);
imshow(classImage, []);
title('Citra biner');
% colormap(h,parula);
class = zeros(size(img_gray_5));
area = zeros(numberOfClasses, 1);

```



```

for n = 1:numberOfClasses
    class(:, :, n) = classImage == n;
    area(n) = sum(sum(class(:, :, n)));
end

[~, min_area] = min(area);

object = classImage == min_area;
bw = medfilt2(object, [5 5]);
bw = bwareaopen(bw, 5000);
s = regionprops(bw, 'BoundingBox');
bbox = cat(1, s.BoundingBox);
RGB = insertShape(img_input_5, 'FilledRectangle', bbox, 'Color',
'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,
'', 'TextBoxOpacity', 0.9, 'FontSize', 11);
axes(handles.img_dec_5);
imshow(RGB);
title('Citra segmentasi');

% clusterisasi
img_clus_5 = imresize(img_gray_5, [4 4]);

```

```

axes(handles.citra_clus_5);
imshow(img_clus_5);
title('Citra clustering');
handles.np5 = img_clus_5;
guidata(hObject, handles);

% --- Executes on button press in input_citra_6.
function input_citra_6_Callback(hObject, eventdata, handles)
% hObject    handle to input_citra_6 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
warning('off');
% citra 6
% -> input citra
[filename,pathname] = uigetfile('*.jpg');
img_input_6 = imread(fullfile(pathname,filename));
% -> set gambar input original ke axes
axes(handles.img_original_6);
imshow(img_input_6);
title('Citra original');
% -> ubah gambar ke grayscale
img_gray_6 = rgb2gray(img_input_6);
handles.citra_s_g_1 = img_gray_6;
% -> set gambar grayscale ke axes
axes(handles.img_grayscale_6);
imshow(img_gray_6);
title('Citra grayscale');

% -> tentukan jumlah class
numberOfClasses = 2;
indexes = kmeans(img_gray_6(:), numberOfClasses);
classImage = reshape(indexes, size(img_gray_6));
% h = subplot(2, 2, 3);
axes(handles.img_seg_6);
imshow(classImage, []);
title('Citra biner');
% colormap(h,parula);
class = zeros(size(img_gray_6));
area = zeros(numberOfClasses, 1);

for n = 1:numberOfClasses
    class(:, :, n) = classImage == n;
    area(n) = sum(sum(class(:, :, n)));
end

[~,min_area] = min(area);

object = classImage==min_area;
bw = medfilt2(object,[5 5]);
bw = bwareaopen(bw, 5000);
s = regionprops(bw,'BoundingBox');
bbox = cat(1, s.BoundingBox);

```

```

RGB = insertShape(img_input_6, 'FilledRectangle', bbox, 'Color',
'yellow', 'Opacity', 0.2);
% RGB = insertObjectAnnotation(RGB, 'rectangle', bbox,
'', 'TextBoxOpacity', 0.9, 'FontSize', 11);
axes(handles.img_dec_6);
imshow(RGB);
title('Citra segmentasi');

% clusterisasi
img_clus_6 = imresize(img_gray_6, [4 4]);
axes(handles.citra_clus_6);
imshow(img_clus_6);
title('Citra clustering');
handles.np6 = img_clus_6;
guidata(hObject, handles);

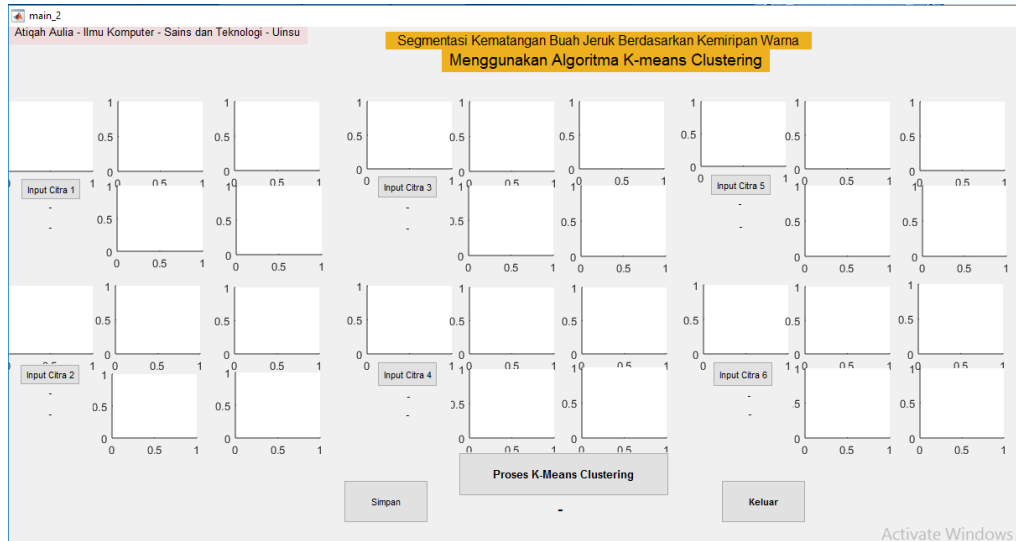
% --- Executes on button press in btn_keluar.
function btn_keluar_Callback(hObject, eventdata, handles)
% hObject      handle to btn_keluar (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
close all force;

% --- Executes on button press in btn_save.
function btn_save_Callback(hObject, eventdata, handles)
% hObject      handle to btn_save (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
saveas(gcf, 'hasil_pengujian.png');

```

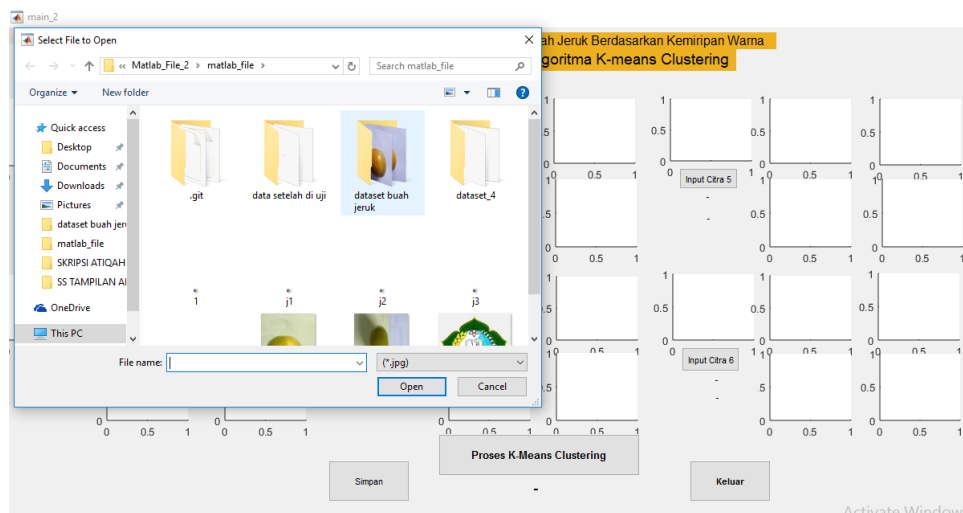
LAMPIRAN

UJI COBA TINGKAT KEBERHASILAN



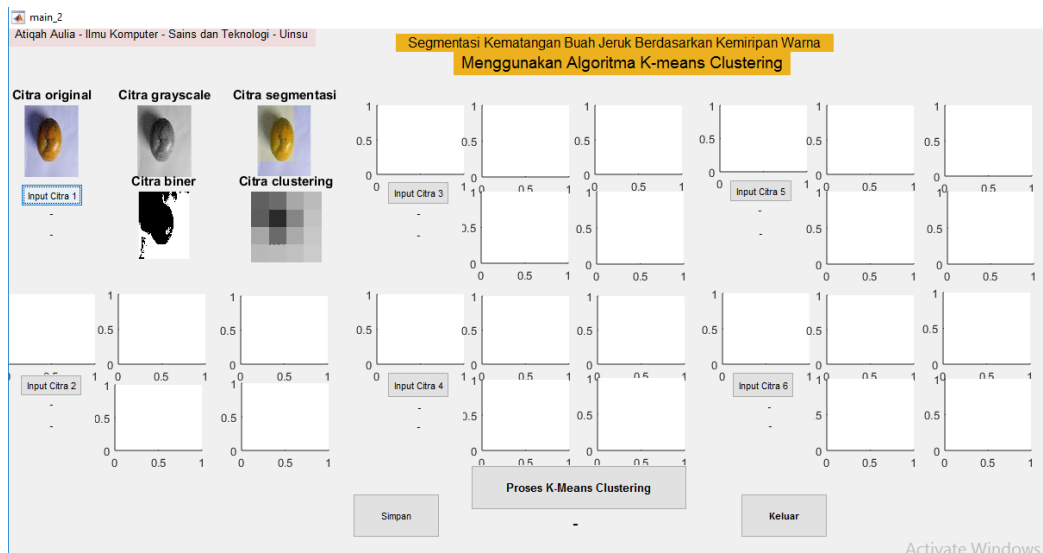
Gambar 1Tampilan Aplikasi

Tampilan diatas adalah tampilan aplikasi segmentasi kematangan buah jeruk berdasarkan kemiripan warna menggunakan algoritma *k-means*. dimana terdapat button input citra 1 sampai button input citra 6 yang akan diproses dengan algortima *k-means*. Fungsi input citra ini untuk mengetahui kematangan buah jeruk dari 6 sudut. Selanjutnya terdapat button proses *k-means clustering*, button simpan dan button keluar.



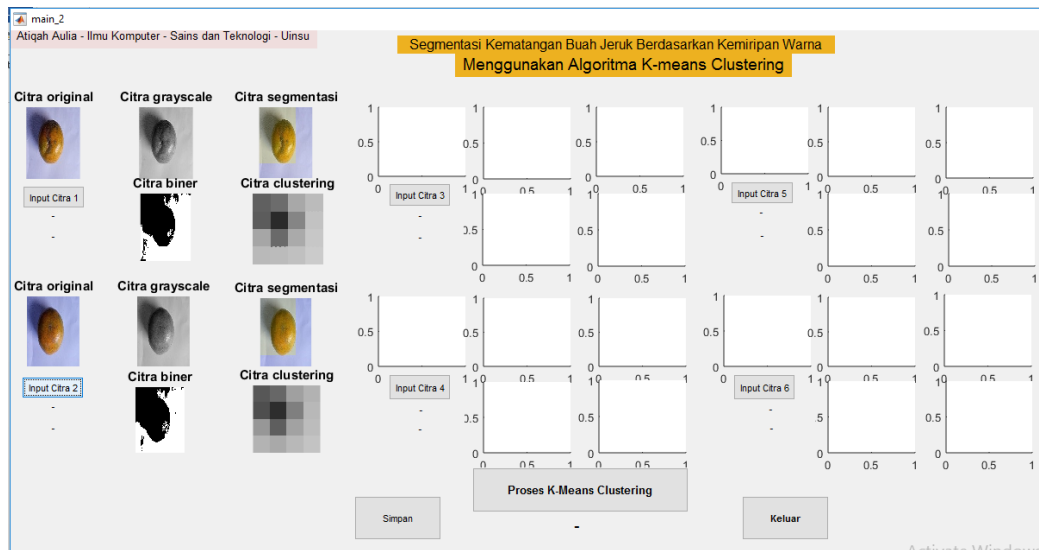
Gambar 2Tampilan Inputan Citra

Setelah itu, klik button input citra 1 ataupun button input citra 2 sampai button input citra 6 dari file dataset buah jeruk untuk mengambil gambar citra dari sudut atas, bawah, kanan, kiri, depan dan belakang, dengan nama file jeruk1 *.jpg sampai jeruk48 *.jpg dataset dengan ukuran citra 124 x 165 *pixel*.



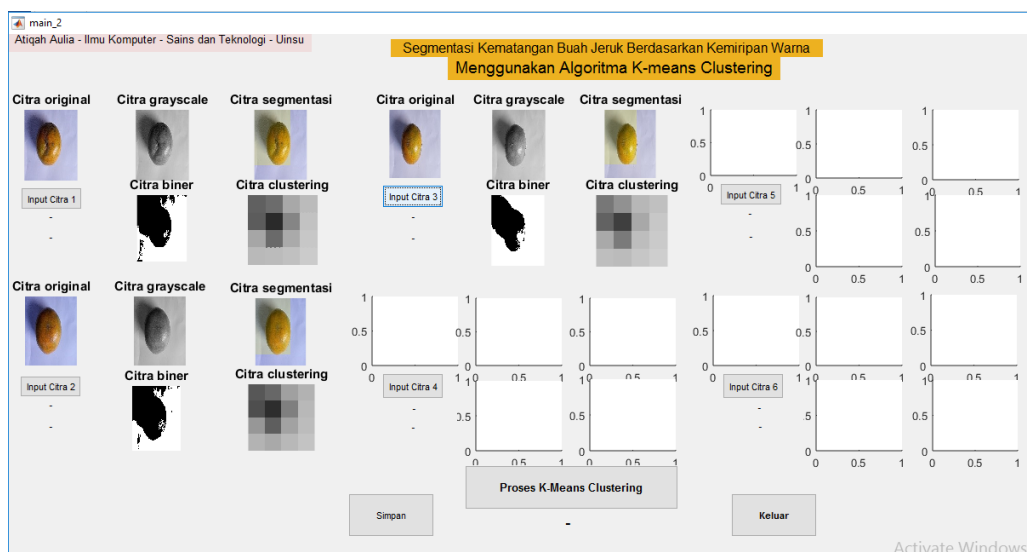
Gambar 3Tampilan input citra 1

Berikut tampilan input citra 1, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *clusterkan* berdasarkan kematangannya.



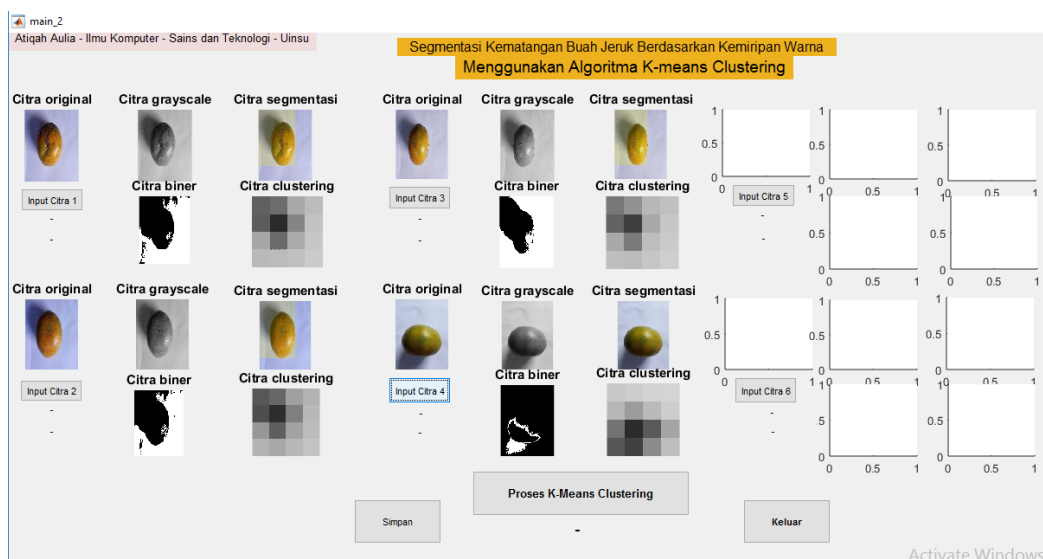
Gambar 4 Tampilan input citra 2

Selanjutnya tampilan input citra 2, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuan citra akan di *cluster* berdasarkan kematangannya.



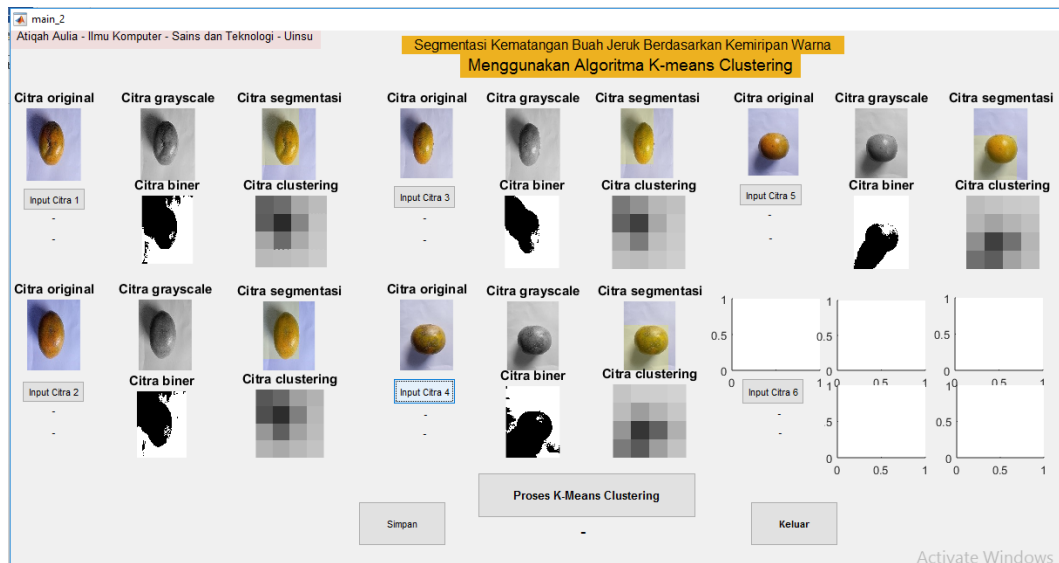
Gambar 5 Tampilan input citra 3

Selanjutnya tampilan input citra 3, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *cluster*kan berdasarkan kematangannya.



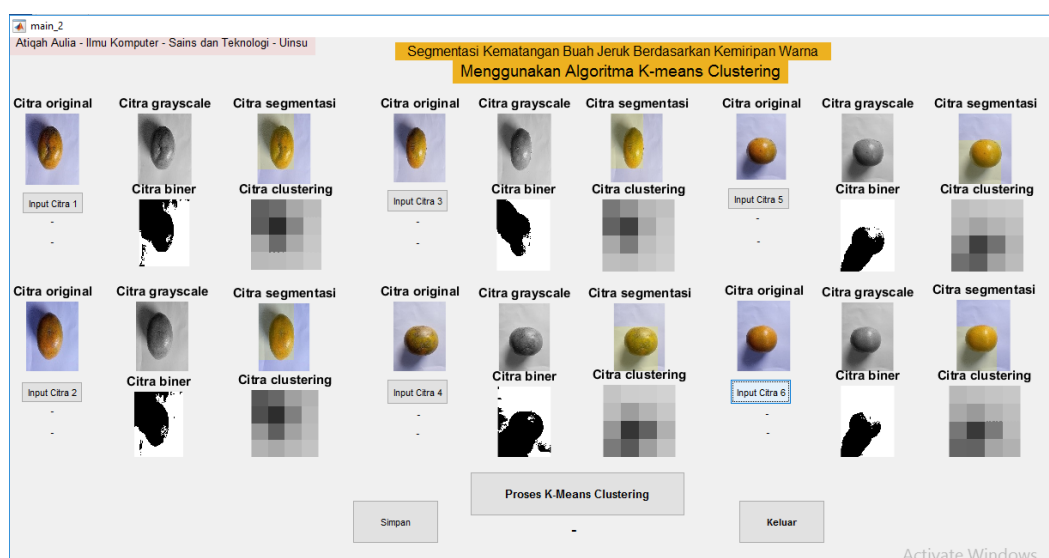
Gambar 6 Tampilan input citra 4

Selanjutnya tampilan input citra 4, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *cluster*kan berdasarkan kematangannya.



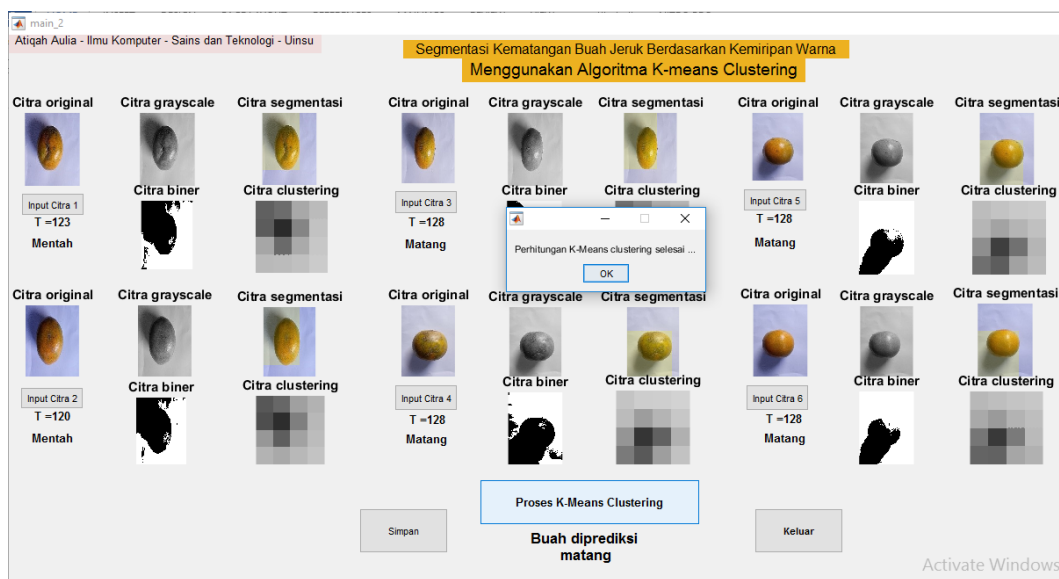
Gambar 7 Tampilan input citra 5

Selanjutnya tampilan input citra 5, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuannya citra akan di *cluster*kan berdasarkan kematangannya.



Gambar 8 Tampilan input citra 6

Selanjutnya tampilan input citra 6, dimana terdapat citra original, citra *grayscale*, citra segmentasi, citra biner, dan citra hasil *clustering*. Proses tahapan yang dilakukan yaitu yang pertama akan menampilkan inputan dari citra original selanjutnya diubah ke citra *grayscale*, selanjutnya dari citra *grayscale* akan di segmentasi menggunakan algoritma *k-means*, dari hasil segmentasi menggunakan algoritma *k-means* menghasilkan citra biner dan sebagai acuan citra akan di *cluster* berdasarkan kematangannya.



Gambar 9 Tampilan proses *k-means clustering*

Berikut tampilan proses *k-means clustering* buah jeruk yang diukur dari 6 sudut. Sudut atas, bawah, kanan, kiri, depan dan belakang, dengan nama file jeruk1 *.jpg sampai jeruk6 *.jpg dengan ukuran citra 124 x 165 *pixel*, nilai T masing-masing didapatkan secara otomatis dari sistem proses segmentasi menggunakan algoritma *k-means* yang menyatakan bahwa buah jeruk diprediksi matang ataupun mentah dari masing-masing input citra. Selanjutnya pada proses *k-means clustering* proses inputan citra 1 sampai inputan citra 6 akan disatukan, apabila nilai matang lebih besar dari 3 maka dinyatakan matang. Akan tetapi, apabila nilai matang lebih kecil atau sama dengan 3 maka dinyatakan mentah. Setelah hasil citra dari 6 sudut diproses dengan proses *k-means clustering*, klik

button simpan untuk menyimpan gambar proses *k-means clustering*, dan selanjutnya klik button keluar jika ingin keluar maka proses *k-means clustering* selesai. Seperti gambar 4.2.1.9. Tampilan proses *k-means clustering*.

DAFTAR RIWAYAT HIDUP



NAMA	: ATIQAHA AULIA
NIM	: 0701163098
TEMPAT, TANGGAL LAHIR	: HUTAGODANG, 03 MARET 1998
ALAMAT	: HUTAGODANG
KELURAHAN	: HUTAGODANG
KECAMATAN	: ULU PUNGKUT
KABUPATEN	: MANDAILING NATAL
AGAMA	: ISLAM
NO. HP	: 082241946967
NAMA ORANG TUA	
AYAH	: ARSYAD LUBIS
IBU	: MASLIYA NASUTION
ALAMAT EMAIL	: atiqahaulialbs6@gmail.com
PENDIDIKAN FORMAL	
SD	: SDN 144459 HUTAGODANG
SMP	: SMPN 1 ULU PUNGKUT
SMA	: SMAN 1 KOTANOPAN



KARTU BIMBINGAN SKRIPSI
PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
T.A 2020/2021

Nama : Atiqah Aulia	Pembimbing I : Dr. Mhd. Furqan, S.Si., M.Comp. Sc
NIM : 0701163098	Pembimbing II : Sriani, S.Kom., M.Kom
Prog. Studi : Ilmu Komputer	
Judul Skripsi : SEGMENTASI KEMATANGAN BUAH JERUK BERDASARKAN KEMIRIPAN WARNA MENGUNAKAN ALGORITMA K-MEANS	

PERT.	Tanggal	Materi Bimbingan	Tanda Tangan Pembimbing Skripsi	
			PS. I	PS.II
I	07 April 2020	Revisi BAB I (Terkait mengenai isi dari Latar Belakang, Rumusan Masalah, Batasan Masalah, dan Tujuan Penelitian)		
II	15 April 2020	Revisi BAB I (Rumusan masalah dan Tujuan Penelitian, dimana pada tujuan penelitian merupakan jawaban dari rumusan masalah).		
III	30 April 2020	ACC BAB I (Lanjut penyusunan BAB II, III, dan Daftar Pustaka)		
IV	03 Mei 2020	Revisi Judul Proposal Skripsi		
V	08 Mei 2020	Revisi BAB II (Susunan teori-teori terkait harus runut, dan simbol-simbol Flowchart harus dibuat sendiri)		

VI	09 Juni 2020	Revisi BAB III (Bentuk Jadwal dan Waktu Penelitian dibuat sederhana, dan bentuk flowchart harus sesuai dengan diagram penelitian)	\$.	87
VII	11 Juni 2020	ACC BAB I, II, III dan Daftar pustaka dari pembimbing II. Lanjut ke pembimbing I dengan melengkapi proposal skripsi	\$.	87
VIII	16 Agustus 2020	Bimbingan serta ACC Proposal Skripsi dari pembimbing I. Lanjut ke Seminar Proposal	\$.	87
IX	18 November 2020	Revisi Proposal Skripsi Lanjut ke BAB IV dan V.	\$.	87
X	3 November 2020	ACC BAB IV dan V Lanjut ke Abstrak, Daftar Pustaka, Lembar Persetujuan, Lampiran	\$.	87
XI	16 Desember 2020	ABSTRAK, DAFTAR PUSTAKA, LEMBAR PERSETUJUAN, KESIMPULAN DAN LAMPIRAN	\$.	87
XII	4 Januari 2021	ACC SIDANG MUNAQASYAH	\$.	87

Medan,2021

An. Dekan
Ketua Jurusan/Program Studi
Ilmu Komputer


Ilka Zulfa, S.Kom., M.Kom.
NIP. 198506042015031006