

SEGMENTASI PADA CITRA BUAH MANGGA DENGAN MENGUNAKAN APLIKASI MATLAB

Siswanto¹⁾, Gunawan Pria Utama²⁾

^{1,2}Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Ciledug Raya, Petukangan Utara, Jakarta Selatan, 12260

Telp: (021) 5853753, Fax : (021) 5853753

E-mail: siswantobl@gmail.com¹⁾, gputama@gmsil.com²⁾

Image segmentation on the image of the mango fruit is a process that is intended to get the objects contained in the image or split image mango mangoes into multiple regions with each object or area that has a resemblance to these attributes, such as shape, color and size. The formulation of the problem in this research is how to get the objects contained in mango and how to calculate the derivative (slope) of an image of mangoes in all directions. The purpose of image segmentation is to sort mangoes by size using image matlab. Segmentasi application is also used to do as a first step to implement image segmentation classification objek. Setelah implemented, features on the object object diambil. Fitur mangoes mangoes can be comparison width and length of the object, the average color of the object, or even the texture on objects fruit mangga. Selanjutnya through classification, object types mangoes super size can be determined. Image segmentation technique based on two basic properties of gray level values: discontinuity and similarity between pixels. In the first form, the separation of the mango fruit image based on sudden changes in levels of gray, which is the approach line detection (detection line) and edge detection (edge detection) to the image of a mango. The second way is based on the similarities between the pixels of mangoes in an area. Super mango fruit with sweet taste can be detected edge with precision precisely when using modified sobel operator algorithm (GPU & Sis) with magnitude value of $5.24833e + 06$ and speed of process time for 8 seconds and mangoes image length of 15.0 cm and thick mangoes image of 5.5 Cm.

Keyword: *image segmentation, image classification, line detection, edge detection, the similarities between pixels..*

Abstract

Segmentasi citra terhadap citra buah mangga merupakan proses yang ditujukan untuk mendapatkan objek-objek yang terkandung di dalam citra buah mangga atau membagi citra buah mangga ke dalam beberapa daerah dengan setiap objek atau daerah yang memiliki kemiripan aribut, seperti bentuk, warna dan ukuran. Perumusan masalah dalam penelitian ini adalah bagaimana mendapatkan objek-objek yang terkandung dalam buah mangga dan bagaimana mengkalkulasi derivative (slope) dari sebuah citra buah mangga dalam semua arah. Tujuan dari segmentasi citra ini untuk memilah buah mangga berdasarkan ukuran dengan menggunakan aplikasi matlab. Segmentasi citra juga biasa dilakukan sebagai langkah awal untuk melaksanakan klasifikasi objek. Setelah segmentasi citra dilaksanakan, fitur yang terdapat pada objek buah mangga diambil. Fitur objek buah mangga dapat berupa perbandingan lebar dan panjang objek, warna rata-rata objek, atau bahkan tekstur pada objek buah mangga. Selanjutnya melalui klasifikasi, jenis objek buah mangga ukuran super dapat ditentukan. Teknik segmentasi citra didasarkan pada dua property dasar nilai aras keabuan: ketidaksinambungan dan kesamaan antar piksel. Pada bentuk pertama, pemisahan citra buah mangga didasarkan pada perubahan mendadak pada aras keabuan, yaitu dengan pendekatan deteksi garis (line detection) dan deteksi tepi (edge detection) pada citra buah mangga. Cara kedua didasarkan pada kesamaan antar piksel buah mangga dalam suatu area. Buah mangga super dengan rasa manis dapat dideteksi tepinya dengan tepat presesinya bila menggunakan algoritma operator modified sobel (GPU & Sis) dengan nilai magnitude sebesar $5.24833e+06$ dan kecepatan waktu proses selama 8 detik serta panjang citra mangga sebesar: 15.0 Cm dan tebal citra mangga sebesar 5.5 Cm.

Kata Kunci: *segmentasi citra, klasifikasi citra, deteksi garis, deteksi tepi, kesamaan antar piksel.*

1. PENDAHULUAN

Pesatnya perkembangan dari ICT yang semakin canggih telah menyebabkan perubahan besar pada kebiasaan para penggunanya. Perubahan tersebut adalah pada penggunaan device yang pada awalnya hanya digunakan

untuk keperluan mengetik, buat program, transaksi elektronik dan *browsing* saja, maka saat ini menuju pada penggunaan yang mulai menunjang kegiatan sehari - hari, baik itu keperluan pekerjaan maupun keperluan hiburan. Sehingga dengan *source code* matlab dapat

digunakan untuk memilah buah mangga ukuran super dengan proses segmentasi citra buah mangga.

Di antara banyak kegiatan yang dilakukan oleh manusia, Segmentasi citra terhadap citra buah mangga merupakan proses yang ditujukan untuk mendapatkan objek-objek yang terkandung di dalam citra buah mangga atau membagi citra buah mangga ke dalam beberapa daerah dengan setiap objek atau daerah yang memiliki kemiripan aribut, seperti bentuk, warna dan ukuran.

Masalah pada penelitian ini adalah bagaimana mendapatkan objek-objek yang terkandung dalam buah mangga, bagaimana mengkalkulasi *derivative (slope)* dari sebuah citra buah mangga dalam semua arah dan bagaimana *source code* matlab untuk proses segmentasi citra buah mangga sampai diperoleh jenis buah mangga ukuran super.

Tujuan penelitian ini adalah: melakukan proses segmentasi citra untuk memilah buah mangga berdasarkan ukuran dengan menggunakan aplikasi matlab. Batasan masalah dalam penelitian ini adalah

Proses segmentasi citra buah mangga sampai diperoleh jenis buah mangga ukuran super dengan *source code* matlab dan teknik segmentasi citra didasarkan pada dua *property* dasar nilai aras keabuan: ketidaksinambungan dengan pendekatan deteksi garis dan deteksi tepi (operator Roberts, Prewitt, Sobel, Canny dan modified Sobel) serta kesamaan antar piksel dalam suatu area (pengambangan berdasarkan histogram, pengambangan iteratif, pengambangan adaptif).

Segmentasi citra terhadap citra merupakan tahapan penting dalam proses pengenalan pola. Setelah objek berhasil tersegmentasi, maka kita dapat melakukan proses ekstraksi ciri citra. Ekstraksi ciri merupakan tahapan yang bertujuan untuk mengekstrak ciri dari suatu objek di mana ciri tersebut digunakan untuk membedakan antara objek satu dengan objek lainnya. [7][10]. Pada citra yang mengandung hanya satu objek, objek dibedakan dari latar belakangnya [3]. Pada citra yang mengandung sejumlah objek, proses untuk memilah semua objek tentu saja lebih kompleks. Contoh penerapan segmentasi adalah untuk membuat fasilitas semacam "*magic wand*", yang biasa terdapat pada perangkat pengedit foto di Adobe Photoshop, seperti gambar 1 yang memperlihatkan pemilihan area ditandai oleh warna khusus

Secara prinsip, segmentasi dilakukan untuk mendapatkan objek yang menjadi perhatian. Segmentasi juga dapat dilakukan sebagai langkah awal untuk melakukan klasifikasi objek. Setelah segmentasi citra dilaksanakan, fitur yang terdapat

pada objek dapat diambil. Teknik segmentasi citra didasarkan pada dua *property* dasar nilai aras keabuan (*gray scale*): ketidaksinambungan dan kesamaan antar piksel[6]. Pada cara pertama, pemisahan citra didasarkan pada perubahan mendadak pada aras keabuan[12]. Contoh yang menggunakan pendekatan ketidaksinambungan antar piksel adalah deteksi garis (*line detector*) dan deteksi tepi (*edge detector*). Pada cara kedua didasarkan pada kesamaan antar piksel dalam suatu area, termasuk dalam cara kedua ini, yaitu:[10]

- a. Pengambangan berdasarkan histogram
- b. Pertumbuhan area
- c. Pemisahan dan penggabungan area
- d. Pengelompokan atau klasifikasi
- e. Pendekatan teori graf
- f. Pendekatan yang dipandu pengetahuan atau dengan menggunakan aplikasi aturan

Berdasarkan teknik yang digunakan segmentasi dapat dibagi menjadi empat kategori, sebagai berikut: [4]

- a. Teknik pengambangan
- b. Metode dengan menggunakan aplikasi batas
- c. Metode dengan menggunakan aplikasi area
- d. Metode hybrid yang mengkombinasikan kriteria batas dan area

Thresholding atau segmentasi *gray-level* adalah sebuah konsep penting terkait dengan pengolahan citra dan visi computer[5]. *Thresholding* adalah konversi antara gambar *gray-level* dan gambar *bilevel*. *Bilevel image* adalah gambar monokrom hanya disusun oleh piksel hitam dan putih. Ini harus berisi informasi yang paling penting dari citra(yaitu, jumlah, posisi dan bentuk objek), tetapi tidak sebanding dengan informasi yang ditawarkan oleh *citragray-level*. Sebagian besar piksel waktu dengan tingkat abu-abu mirip milik objek yang sama. Oleh karena itu, mengelompokkan citra dengan tingkat keabuan piksel dapat mengurangi dan menyederhanakan beberapa operasi pengolahan citra seperti pengenalan pola, dan operasi *thresholding* yang paling penting dalam klasifikasi pemilihan nilai ambang tunggal. Semua tingkat abu-abu di bawah nilai ini diklasifikasikan sebagai hitam(0), dan di atas putih(1).

Deteksi garis pada citra diperoleh melalui penggunaan *cadar(mask)* dengan contoh tercantum di gambar 2 [4]. *Cadar(a)* berguna untuk memperoleh garis horizontal, *cadar(b)* untuk mendapatkan garis yang berorientasi 45° , *cadar(c)* untuk memperoleh garis tegak, dan *cadar(d)* untuk mendapatkan garis yang berorientasi -45° .

Fungsi deteksi garis berguna untuk menguji *cadar-cadar* yang terdapat di gambar 1. Fungsi

deteksi garis dapat dipakai juga sekaligus untuk menguji cadar-cadar yang lain.

$$\begin{matrix}
 \begin{pmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{pmatrix} & \begin{pmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{pmatrix} & \begin{pmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{pmatrix} & \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \\
 \text{(a)} & \text{(b)} & \text{(c)} & \text{(d)}
 \end{matrix}$$

Gambar 1. Empat Pasang Cadar Untuk Mendeteksi Keberadaan Garis

Deteksi tepi (*Edge Detection*) berfungsi untuk memperoleh tepi objek[6] Deteksi tepi memanfaatkan perubahan nilai intensitas yang drastis pada batas dua area[8]. Definisi tepi di sini adalah “himpunan piksel yang terhubung yang terletak pada batas dua area”[1]. Perlu diketahui, tepi sesungguhnya mengandung informasi sangat penting. Informasi yang diperoleh dapat berupa bentuk maupun ukuran objek

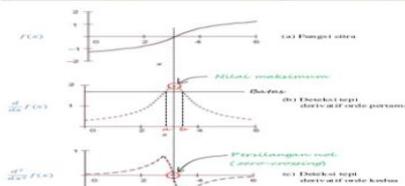
Umumnya deteksi tepi menggunakan dua macam detektor, yaitu detektor baris(Hy) dan detektor kolom(Hx). Beberapa contoh yang tergolong jenis ini adalah operator Roberts, Prewitt, Sobel dan Frei-Chen[7].

Tabel 1 memberikan definisi turunan orde pertama dan kedua baik pada bentuk kontinu maupun diskret. Bentuk diskret sangat berguna untuk melakukan deteksi tepi. Gambar 2 menunjukkan hubungan antara fungsi citra dan deteksi tepi orde pertama dan orde kedua. Perlu diketahui, terkait dengan turunan, tepi sesungguhnya terletak pada:

- Nilai absolut maksimum pada turunan pertama
- Persilangan nol(zero-crossing) pada turunan kedua

Tabel 1: Turunan Orde Pertama Dan Kedua Pada Bentuk Kontinu Dan Diskret [3]

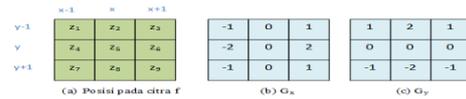
Turunan	Bentuk kontinu	Bentuk diskret
$\frac{df}{dx}$	$\lim_{\Delta x \rightarrow 0} \frac{f(y, x + \Delta x) - f(y, x)}{\Delta x}$	$f(y, x+1) - f(y, x)$
$\frac{df}{dy}$	$\lim_{\Delta y \rightarrow 0} \frac{f(y + \Delta y, x) - f(y, x)}{\Delta y}$	$f(y+1, x) - f(y, x)$
$\nabla f(y, x)$	$\left[\frac{df}{dx} \quad \frac{df}{dy} \right]$	$[f(y, x+1) - f(y, x), f(y+1, x) - f(y, x)]$
$\frac{d^2f}{dx^2}$	$\lim_{\Delta x \rightarrow 0} \frac{\left(\frac{df}{dx}\right)(y, x + \Delta x) - \left(\frac{df}{dx}\right)(y, x)}{\Delta x}$	$f(y, x+1) - 2f(y, x) + f(y, x-1)$
$\frac{d^2f}{dy^2}$	$\lim_{\Delta y \rightarrow 0} \frac{\left(\frac{df}{dy}\right)(y + \Delta y, x) - \left(\frac{df}{dy}\right)(y, x)}{\Delta y}$	$f(y+1, x) - 2f(y, x) + f(y-1, x)$
$\nabla^2 f(y, x)$	$\frac{d^2f}{dx^2} + \frac{d^2f}{dy^2}$	$f(y, x+1) + f(y, x-1) - 4f(y, x) + f(y+1, x) + f(y-1, x)$



Gambar 2: Deteksi Tepi Orde Pertama Dan Orde Kedua Pada Arah X[7][11]

Operator Sobel adalah salah satu operator yang digunakan dalam pengolahan citra dan visi komputer (*computer vision*), terutama dalam algoritma deteksi tepi, dan menciptakan sebuah gambar yang menekankan tepi dan transisi.

Formula operator Sobel dapat dirangkum dalam Gambar 3.



Gambar 3: Formula Operator Sobel

Canny edge detector dikembangkan oleh John F. Canny pada tahun 1986 dan menggunakan algoritma multi-tahap untuk mendeteksi berbagai tepi dalam gambar. Walaupun metode tersebut telah berumur cukup lama, namun metode tersebut telah menjadi metode deteksi tepi standar dan masih dipakai dalam penelitian.

Adapun kategori algoritma yang dikembangkan oleh John F. Canny adalah sebagai berikut:

- Deteksi :Kemungkinan mendeteksi titik tepi yang benar harus dimaksimalkan sementara kemungkinan salah mendeteksi titik tepi harus diminimalkan. Hal ini dimaksudkan untuk memaksimalkan rasio *signal-to-noise*.
- Lokalisasi: Tepi terdeteksi harus sedekat mungkin dengan tepi yang nyata.
- Jumlah tanggapan : Satu tepi nyata Tidak harus menghasilkan lebih dari satu ujung yang terdeteksi.

Tidak dapat dipungkiri bahwa semua gambar yang diambil dari kamera akan berisi sejumlah *noise*. *Noise* harus dikurangi untuk mencegah *noise* salah dideteksi sebagai tepian. Oleh karena itu pada langkah pertama gambar harus diperhalus dengan menggunakan Gaussian filter. Inti dari Gaussian filter adalah standar deviasi dengan $\sigma = 1.4$ ditunjukkan pada persamaan (1) di bawah ini.

$$G = \frac{1}{150} \begin{bmatrix} 2 & 4 & 6 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 6 & 12 & 16 & 12 & 6 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 6 & 4 & 2 \end{bmatrix} \dots\dots\dots(1)$$

Algoritma Canny pada dasarnya menemukan titik tepi pada gambar *grayscale* dengan perubahan nilai intensitas yang paling besar, daerah ini ditemukan dengan menentukan gradien gambar. Gradien pada setiap piksel gambar yang telah diperhalus ditentukan dengan menerapkan operator Sobel. Langkah Kedua adalah memperkirakan gradien pada arah x dan y. Hal tersebut ditunjukkan dalam Persamaan (2).

$$\begin{matrix}
 G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ 1 & 0 & -1 \end{bmatrix} \\
 G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}
 \end{matrix} \dots\dots\dots(2)$$

Magnitudo gradien (juga dikenal sebagai kekuatan tepi) dapat ditentukan sebagai jarak *Euclidean* yang diukur mengukur dengan menerapkan hukum *Pythagoras* seperti yang ditunjukkan dalam Persamaan (3). Yang terkadang disederhanakan dengan menerapkan ukuran jarak *Manhattan* seperti yang ditunjukkan dalam Persamaan (4) untuk mengurangi kompleksitas komputasi.

$$|G| = \sqrt{G_x^2 + G_y^2} \dots\dots\dots(3)$$

$$|G| = |G_x| + |G_y| \dots\dots\dots(4)$$

dimana:

Gx dan Gy adalah gradien pada masing-masing arah x dan y.

Namun, tepian biasanya luas dan dengan demikian tidak dapat menunjukkan persis di mana tepian yang sebenarnya. Arah tepian harus ditentukan dan disimpan untuk menentukan tepian yang sebenarnya, seperti ditunjukkan dalam Persamaan (5).

$$\theta = \arctan \left(\frac{|G_y|}{|G_x|} \right) \dots\dots\dots(5)$$

Pada langkah ketiga bertujuan untuk mengkonversikan tepian yang masih *blurred* pada gambar hasil *magnitude gradien* hingga menghasilkan tepian yang tajam. Pada dasarnya hal ini dilakukan dengan mempertahankan semua maxima lokal dalam gambar gradien dan menghapus segala sesuatu yang lain.

Operator Robert Cross adalah operator gradien yang sederhana berukuran 2x2. Operator ini menyediakan pendekatan paling sederhana dari *magnitude gradien*, yang dapat ditunjukkan dengan fungsi matematis (6) berikut ini, yaitu:

$$G[f(i,j)] = [f(i,j) - f(i+1,j+1)] + [f(i+1,j) - f(i,j+1)] \dots\dots\dots(6)$$

Mask konvolusi untuk Operator **ROBERTS** seperti Gambar 4.

1	1
-1	-1

Gambar 4: Mask Konvolusi Untuk Operator Roberts

Pengurangan 255 terhadap nilai hasil penjumlahan 2 konvolusi dari citra dimaksudkan untuk menegaskan, karena hasil konvolusi biasanya berada di sekitar nilai 0 (nilai kecil), sehingga jika ditampilkan, citra tepi kebanyakan berwarna gelap (hitam).

Persamaan gradien pada operator Prewitt sama dengan gradien pada operator sobel perbedaannya adalah pada prewitt menggunakan konstanta c = 1, seperti Gambar 5.

$$P_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ dan } P_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Gambar 5: Persamaan Gradien pada Operator Prewitt

Pengembangan Dwi-Aras juga dikenal dengan pengembangan berdasarkan histogram merupakan segmentasi yang paling sederhana dilaksanakan dengan menggunakan ambang intensitas. Nilai yang lebih kecil daripada nilai ambang diperlakukan sebagai area pertama dan yang lebih besar dari pada atau sama dengan nilai ambang dikelompokkan sebagai area yang kedua. Dalam hal ini, salah satu area tersebut berkedudukan sebagai latar belakang. Cara seperti itulah yang disebut pengembangan dwi-aras (*bi-level thresholding*)[5]. Secara matematis, hal itu dinyatakan dengan:

$$b(y, x) = f(s) = \begin{cases} 1, & \text{untuk } (y,x) \geq T \\ \dots\dots\dots(7) \\ 0, & \text{untuk } (y,x) < T \end{cases}$$

Pada rumus (7) di atas, T merupakan ambang intensitas. Dalam prakteknya nilai 1 atau 0 pada persamaan (7) di atas dapat dipertukarkan.

Salah satu cara untuk menentukan nilai ambang adalah dengan memperhatikan histogram citra[13]. Sebagai contoh perhatikan Gambar 6.



Gambar 6: Citra Daun Dan Histogramnya[3]

Pengembangan interaktif dikenal juga dengan sebutan pengembangan optimal (*Optimum Thresholding*)[2]. Pengembangan interaktif digunakan untuk mengatasi persoalan yang dihadapi bila menggunakan pengembangan intensitas terletak pada penentuan nilai ambang(T) secara otomatis, dengan menentukan nilai ambang iteratif secara interaktif.

Contoh hasil pengembangan secara interaktif dapat di lihat pada Gambar 7.



Gambar 7: Contoh Hasil Pengembangan Secara Interaktif[3]

Ada tiga pendekatan yang biasa digunakan untuk mewujudkan pengembangan ini. Pendekatan pertama menggunakan statistik rerata terhadap intensitas lokal, secara matematis dapat dirumuskan seperti persamaan (8), sebagai berikut:

$$T = \frac{\sum_{i,j} w \cdot f(i,j)}{N_w} - c \dots\dots\dots(8)$$

Dimana: W menyatakan jendela pada citra
 N_w menyatakan jumlah piksel dalam jendela
 C menyatakan nilai konstanta
 T menyatakan nilai ambang

Pendekatan pengembangan adaptif yang kedua menggunakan statistik median. Secara matematis dapat dirumuskan seperti persamaan (9), sebagai berikut:

$$T = \text{median}(f(y,x), (y,x) \in W) - c \dots\dots\dots(9)$$

Dimana: W menyatakan jendela pada citra
 N_w menyatakan jumlah piksel dalam jendela
 C menyatakan nilai konstanta
 T menyatakan nilai ambang

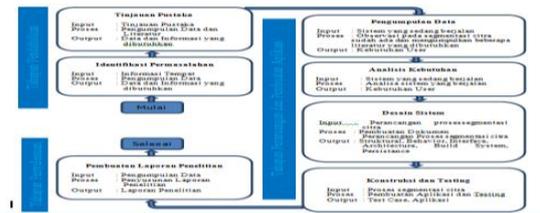
Pendekatan pengembangan adaptif yang ketiga menggunakan statistik maksimum dan minimum. Secara matematis dapat dirumuskan seperti persamaan (10), sebagai berikut:

$$T = (\text{Max}(f(y,x), (y,x) \in W) + \text{Min}(f(y,x), (y,x) \in W)) / 2 \dots\dots\dots(10)$$

Dimana: W menyatakan jendela pada citra
 N_w menyatakan jumlah piksel dalam jendela
 T menyatakan nilai ambang

2. METODE

Metodologi penelitian berisi langkah-langkah yang digunakan dalam penelitian ini agar terstruktur dengan baik. Dengan sistematika ini proses penelitian dapat dipahami dan diikuti oleh pihak lain. Penelitian yang dilakukan untuk merancang sistem diperoleh dari pengamatan data yang ada. Adapun langkah-langkah yang dilakukan untuk mencapai tujuan dari penelitian ini adalah seperti Gambar 8.



Gambar 8 : Kerangka Pikir Penelitian

Tahapan yang di jelaskan pada gambar 8 kerangka pikir penelitian tersebut adalah sebagai berikut :

2.1 Identifikasi Masalah

Tahap awal dalam penelitian ini adalah merumuskan masalah yang akan dijadikan sebagai objek penelitian. Perumusan masalah dilakukan dengan terlebih dahulu melihat kondisi aktual di lapangan. Setelah masalah dirumuskan langkah selanjutnya adalah menentukan tujuandari penelitian. Tujuan penelitian ini merupakan sasaran yang nantinya ingin diwujudkan dari penyelesaian permasalahan yang diteliti.

2.2 Tinjauan Pustaka

Tinjauan pustaka dilakukan guna mencari literatur pendukung penelitian ini. Pada tahap ini dijelaskan dengan mempelajari aplikasi yang berhubungan dengan konvensional dan digital segmentasi citra (*image processing*), teori-teori, untuk pengumpulan data dan *tools* yang digunakan oleh peneliti. Serta dijelaskan mengenai metode yang digunakan.

2.3 Metode Pengumpulan Data

Metode yang digunakan untuk mengumpulkan data pada penelitian ini ada dua cara yaitu dengan melakukan observasi terhadap proses segmentasi citra dan literatur. Observasi digunakan untuk memperoleh dan mengumpulkan data yang dibutuhkan. Pengamatan ini dilakukan pada segmentasi citra konvensional dan proses segmentasi citra yang sudah beredar.

a. Observasi

Observasi dilakukan dengan mengumpulkan data dan informasi yang dibutuhkan dalam perancangan, pengembangan sistem serta proses kegiatan dari proses segmentasi citra yang diterapkan.

b. Studi literatur

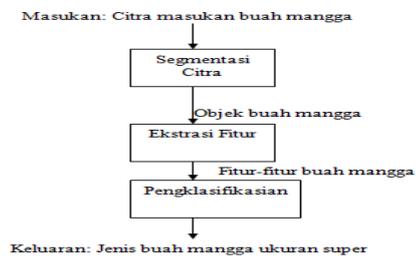
Sedangkan Studi literatur digunakan untuk mengumpulkan data dari penelitian terdahulu, pembelajaran dari berbagai macam literatur dan dokumen seperti buku, jurnal dan teori-teori yang mendukung penelitian, *tools* yang akan digunakan dan data penunjang lainnya

yang berkaitan dengan proses segmentasi citra dengan menggunakan aplikasi matlab.

2.4 Analisis Kebutuhan

Analisis kebutuhan proses segmentasi citra dilakukan untuk mengetahui kebutuhan pengguna terhadap proses segmentasi citra yang dikembangkan. Hal ini perlu dilakukan agar proses yang dikembangkan sesuai dengan kebutuhan pengguna. Dibagian ini jugadijelaskan siapa saja yang akan menggunakan proses segmentasi citra ini, dan informasi apa saja yang digunakan oleh mereka.

Proses Segmentasi Citra Dapat di lihat pada Gambar 9.



Gambar 9: Proses Segmentasi Citra

Tahapan pengembangan yang digunakan didalam penelitian ini yaitu menggunakan model SDLC (*System Development Life Cycle*) dengan menggunakan model / metode *waterfall*. [9]

2.5 Desain Aplikasi

Pada tahap ini dibuat perancangan aplikasi yang akan diusulkan. Perancangan tersebut meliputi, merancang tampilan user, kemudian yang terakhir adalah merancang coding program matlab dari algoritma yang digunakan dalam aplikasi ini.

2.6 Implementasi Aplikasi Dan Testing

Implementasi Aplikasi dan testing adalah proses untuk memastikan bahwa aplikasi yang dikembangkan bebas dari kesalahan, dilakukan testing (*uji coba*) pada aplikasi tersebut. Pada tahap ini juga akan dilakukan evaluasi terhadap hasil penelitian yang dilakukan. Evaluasi dilakukan mencakup evaluasi hasil dan manfaat cara dengan membandingkan hasil yang didapatkan dengan kebutuhan pengguna.

2.7 Pembuatan Laporan Penelitian

Langkah terakhir dari penelitian ini adalah membuat laporan penelitian. Laporan ini berisi hal-hal yang dikerjakan selama penelitian dan hasil yang didapatkan pada saat melakukan penelitian.

3. ANALISA DAN RANCANGAN

3.1. Analisa Masalah

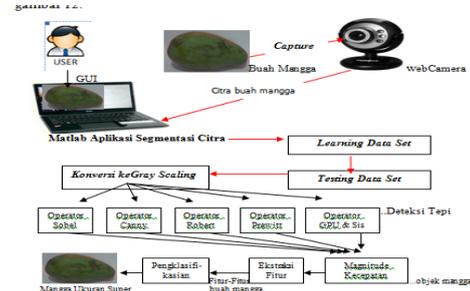
Citra buah mangga super yang ada di pasar tradisional, pasar modern dan toko buah memiliki kemiripan aribut, seperti bentuk, warna dan ukuran sehingga menyulitkan bagi penjual dan pembeli untuk mengidentifikasi apakah ini buah mangga super yang akan dipilih. Belum lagi bentuk, warna dan ukuran buah besarnya yang sama belum tentu buah mangga super, karena ternyata tingkat kematangan buahnya belum maksimal dan rasanya tidak manis. Demikian juga bentuk dan warna yang sama tetapi rasanya manis, ini juga bukan mangga super.

Perumusan masalah dalam penelitian ini adalah bagaimana mendapatkan objek-objek yang terkandung dalam buah mangga dan bagaimana mengkalkulasi *derivative (slope)* dari sebuah citra buah mangga dalam semua arah.

3.2. Strategi Pemecahan Masalah

Dari permasalahan yang telah diuraikan diatas, maka dibuatlah suatu aplikasi segmentasi citra ini untuk memilah buah mangga berdasarkan ukuran dengan menggunakan aplikasi matlab. Segmentasi citra terhadap citra buah mangga merupakan proses yang ditujukan untuk mendapatkan objek-objek yang terkandung di dalam citra buah mangga atau membagi citra buah mangga ke dalam beberapa daerah dengan setiap objek atau daerah yang memiliki kemiripan aribut, seperti bentuk, warna dan ukuran. Segmentasi citra juga biasa dilakukan sebagai langkah awal untuk melaksanakan klasifikasi objek. Setelah segmentasi citra dilaksanakan, fitur yang terdapat pada objek buah mangga diambil. Fitur objek buah mangga dapat berupa perbandingan lebar dan panjang objek, warna rata-rata objek, atau bahkan tekstur pada objek buah mangga. Selanjutnya melalui klasifikasi, jenis objek buah mangga ukuran super dapat ditentukan. Teknik segmentasi citra didasarkan pada dua *property* dasar nilai aras keabuan: ketidaksinambungan dan kesamaan antar piksel. Pada bentuk pertama, pemisahan citra buah mangga didasarkan pada perubahan mendadak pada aras keabuan, yaitu dengan pendekatan deteksi garis (*line detection*) dan deteksi tepi (*edge detection*) pada citra buah mangga. Cara kedua didasarkan pada kesamaan antar piksel buah mangga dalam suatu area.

Aplikasi segmentasi buah mangga dibuat dengan bahasa pemrograman matlab dilengkapi dengan *graphical user interfaces* yang rancangan dapat dilihat pada gambar 12 dengan memasukkan citra buah mangga yang diinput/dibaca dengan *webcamera* ukuran 506 x 351 piksel, kemudian proses segmentasi dilakukan dengan teknik pengambangan, metode dengan aplikasi batas tepi dan metode dengan aplikasi batas area. Lalu diuji dengan empat(4) algoritma operator: Sobel, Canny, Robert dan Prewitt serta Modified Sobel (Algoritma Operator Gpu & Sis). Berikut ini adalah gambaran atau *rich picture* pembuatan aplikasi *segmentasi citra buah mangga*, dapat dilihat pada gambar 10.



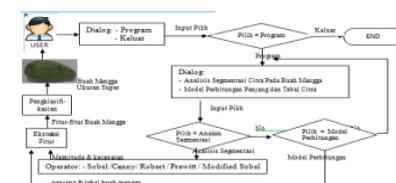
Gambar 10 : Rich Picture Aplikasi Segmentasi Citra Pada Buah Mangga

Dengan adanya aplikasi ini dan segmentasi deteksi tepi diharapkan bisa mengidentifikasi jenis buah mangga ukuran super.

3.3. Perancangan Aplikasi

Tahap perancangan aplikasi dilakukan untuk mencari bentuk yang optimal dari aplikasi yang akan dibuat dengan mempertimbangkan faktor permasalahan dan kebutuhan yang telah dijelaskan sebelumnya, sehingga diperoleh hasil yang optimal dan mudah untuk diimplementasikan.

Berdasarkan analisa yang telah dilakukan, untuk melakukan segmentasi citra buah mangga, user dapat langsung melakukan analisis segmentasi citra buah mangga dan memodelkan perhitungan panjang dan tebal citra buah mangga. Skema rancangan aplikasisegmentasi citra untuk buah mangga yang akan dibuat dapat dilihat pada Gambar 11.



Gambar 11: Alur Program Aplikasi Segmentasi Citra Pada Buah Mangga

3.4. Rancangan Basis Data

a. Model Data Relational

Tabel Buah Mangga dapat dilihat pada Tabel 2.

Tabel 2: Tabel Buah Mangga

Jenis_Mangga	Magnitude	Kecepatan	Panjang	Tebal
Suger Dengan Rasa Nenas	5.24833E6	8	1.5	5.5
Tipe A Dengan Rasa Nenas	5.64833E6	13.3	1.7	4.5
Tipe A Dengan Rasa Anan	5.64833E6	12.8	1.3	4
Tipe B Dengan Rasa Anan	5.44833E6	12.3	1.4	3.5

b. Spesifikasi Basis Data

- Nama File : Buah Mangga
- Isi : Berisi tentang data sample buah mangga
- Media : SQLite
- Primary Key: Jenis_Mangga
- Struktur : lihat Tabel 3

Tabel 3: Struktur Tabel Buah Mangga

No.	Nama Field	Type	Lebar	Keterangan
1	Jenis_Mangga	Variabel	13	Jenis Buah Mangga
2	Magnitude	Float	2.5	Magnitude
3	Kecepatan	Float	2.5	Kecepatan Proses
4	Panjang	Float	3.2	Panjang Citra Mangga
5	Tebal	Float	3.2	Tebal Citra Mangga

3.5. Rancangan Layar

Rancangan layar sangat penting dalam membuat suatu program. Oleh karena itu rancangan layar harus mudah dimengerti dan dipahami oleh user, agar dalam menggunakan aplikasi, user merasa nyaman dalam menggunakannya. Sehingga rancangan layar tidak membingungkan dan juga user tidak mengalami kesulitan saat menggunakan aplikasi ini. Rancangan layar yang akan dibangun terdiri dari beberapa menu yaitu, menu utama, menu program, menu Analisis Perbandingan Segmentasi Pada Citra Buah Mangga dan menu Model Perhitungan Panjang dan Tebal Citra.

a. Rancangan Layar Menu Utama

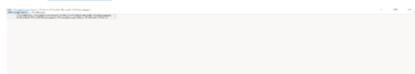
Menu Utama merupakan suatu tampilan *Screenawal* dari aplikasi segmentasi citra untuk buah mangga yang diatur agar dapat muncul secara otomatis ketika aplikasi mulai dijalankan. Pada aplikasi ini *Menu Utamahanya* mempunyai dua pilihan item Program untuk memanggil Menu Program dan item Keluar untuk keluar dari aplikasi segmentasi citra untuk buah mangga. Rancangannya terdapat pada Gambar 12.



Gambar 12 : Rancangan Layar Menu Utama

b. Rancangan Layar Menu Program

Pada menu program terdapat beberapa menu seperti menu Analisis Perbandingan Segmentasi Pada Citra Buah Mangga dan menu Model Perhitungan Panjang dan Tebal Citra. Rancangannya terdapat pada Gambar 13.



c. Rancangan Layar Menu Analisis Perbandingan Segmentasi Pada Citra Buah Mangga

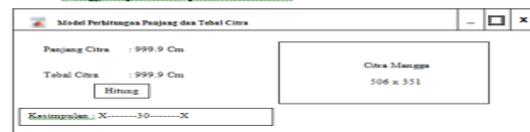
Pada menu analisis perbandingan segmentasi pada citra buah mangga terdapat beberapa *button* diantaranya yaitu operator Sobel, operator Canny, operator Roberts, operator Prewitt dan *Modified Sobel* (Operator GPU & Sis). Serta untuk menampilkan hasil proses masing-masing operator serta nilai *magnitude* dan kecepatan proses dari masing-masing operator. Rancangannya terdapat pada Gambar 14.



Gambar 14 : Rancangan Layar Analisis Perbandingan Segmentasi Pada Citra Buah Mangga

d. Rancangan Layar Menu Model Perhitungan Panjang Dan Tebal Citra

Rancangan layar menu Model Perhitungan panjang dan tebal citra mangga dapat dilihat pada Gambar 15.



Gambar 15: Rancangan Layar Menu Model Perhitungan Panjang dan Tebal Citra

4. IMPLEMENTASI HASIL RANCANGAN DAN HASIL UJICOBA

4.1. Implementasi Hasil Rancangan

a. Perangkat dan Instalasi

Implementasi Aplikasi Segmentasi Citra Pada Buah Mangga berguna untuk mengetahui apakah sistem yang telah dibuat dapat berjalan secara maksimal. Oleh karena itu program harus diuji terlebih dahulu kemampuannya agar dapat berjalan sesuai dengan apa yang diharapkan pada saat implementasi nanti.

b. Tampilan Layar

Pada bagian ini, diuraikan mengenai tampilan layar Aplikasi Segmentasi Citra Pada Buah Mangga mulai dari pertama kali dijalankan sampai selesai dijalankan. Berikut ini akan diberikan penjelasan dan gambar mengenai tampilan – tampilan yang ada pada aplikasi Segmentasi Citra Pada Buah Mangga ini.

Tampilan layar Hasil Deteksi Tepi Operator *Modified Sobel* muncul pada saat user mengklik *button Modified Sobel* pada Menu Analisis Perbandingan Segmentasi

Pada Citra Buah Mangga. Tampilan layar Hasil Deteksi Tepi Operator Modified Sobel dapat dilihat pada Gambar 16.



Gambar 16: Tampilan Layar Hasil Deteksi Tepi Operator Modified Sobel

Gambar dari hasil proses dari deteksi tepi operator Modified Sobel setelah mengklik *button* Modified Sobel dapat dilihat pada Gambar 17. Di mana hasil deteksi tepi operator Modified Sobel dapat dideteksi tepi buahnya tanpa *noise* (a) dan bagian bawahnya terdeteksi dengan tepat tepi citra buah mangganya(b).



Gambar 17: Tampilan Layar Hasil Proses Dari Deteksi Tepi Operator Modified Sobel

Setelah proses dari analisis perbandingan segmentasi citra pada buah mangga dilakukan dengan deteksi tepi lima(5) operator, yaitu: Sobel, Canny, Roberts, Prewitt dan Modified Sobel (Gpu & Sis), maka diperoleh hasil proses deteksi tepi dengan tepat yaitu dengan operator Modified Sobel (Gpu & Sis), setelah memperoleh nilai *magnitude* yang sesuai $5.24833e+06$ dan kecepatan waktu proses selama 8 detik, lalu pilih item Model Perhitungan Panjang dan Tebal Citra Pada Menu Program sehingga akan muncul tampilan layar menu Model Perhitungan Panjang dan Tebal Citra Mangga dapat dilihat pada Gambar 18.



Gambar 18 : Tampilan Layar Menu Model Perhitungan Panjang dan Tebal Citra

4.2. Hasil Uji Coba

Setelah dilakukan uji coba terhadap 6 Citra Buah Mangga dengan ukuran citra 506 x 351 piksel, maka diperoleh hasil uji coba seperti Tabel 4

Tabel 4: Hasil Uji Coba Segmentasi Citra Pada Buah Mangga

Citra Mangga	Deteksi Tepi Operator										Panjang Citra	Tebal Citra	Citra Hasil	Kesimpulan
	Sobel		Canny		Roberts		Prewitt		GPU & Sis					
	Mgn	Wak	Mgn	Wak	Mgn	Wak	Mgn	Wak	Mgn	Wak				
	5.6e6	12.6	1.4e7	12.2	5.7e6	6.65	5.5e6	7.34	5.3e6	8	15.0	5.5		Mangga Super dengan rasa manis
	5.6e6	12.8	1.4e7	12.6	5.9e6	6.63	5.7e6	7.64	5.3e6	8	15.0	5.5		Mangga Super dengan rasa manis
	5.4e6	12.4	1.2e7	11.8	5.5e6	6.51	5.4e6	7.14	5.3e6	8	15.0	5.5		Mangga Super dengan rasa manis
	5.8e6	13.2	1.8e7	12.8	6.3e6	6.68	5.8e6	7.74	5.4e6	12.2	14.0	3.8		Mangga Tipe B dengan rasa asam
	5.7e6	13.0	1.6e7	12.6	5.9e6	6.43	5.7e6	7.64	5.5e6	12.8	13.0	4.0		Mangga Tipe A dengan rasa asam
	5.9e6	13.4	1.7e7	12.6	6.6e6	6.83	5.9e6	7.66	5.6e6	13.2	17.0	4.5		Mangga Tipe A dengan rasa manis

5. SIMPULAN

Simpulan yang dapat diambil dari penelitian ini sebagai berikut :

- a. Dengan dibuatnya program aplikasi segmentasi citra pada buah mangga ini *user* lebih mudah dalam memilah/mensortir buah mangga super dengan rasa manis, buah mangga tipe A dengan rasa manis, buah mangga dengan tipe A dengan rasa asam dan buah mangga tipe B dengan rasa asamkapan saja dan dimana saja.
- b. Buah mangga super dengan rasa manis dapat dideteksi tepinya dengan tepat presesinya bila menggunakan algoritma operator modified sobel (GPU & Sis) dengan nilai *magnitude* sebesar $5.24833e+06$ dan kecepatan waktu proses selama 8 detik serta. panjang citra mangga sebesar: 15.0 Cm dan tebal citra mangga sebesar 5.5 Cm.

Beberapa saran dapat di berikan untuk penelitian lebih lanjut, sebagai berikut :

- a. *File* citra mangga yang dibaca masih banyak mengandung *noise*, dengan pengembangan lebih lanjut dapat ditambahkan *file* citra buah mangga yang lebih bagus dan tidak mengandung *noise*.
- b. Pada pengembangan selanjutnya diharapkan dilengkapi fasilitas untuk mendeteksi citra yang bernoise dan fasilitas untuk memfilter *noise* dari citra buah mangga algoritma filter *Laplacian of gaussian* sebelum proses segmentasi dilakukan.

6. PUSTAKA

[1] Farid, H., 2014. *Fundamentals of Image Processing*, Dartmouth University.
 [2] Harley E. Mayer, A.E.W., 1998. *The-pocket-handbook-of-image-processing-*

- algorithms-in-C.pdf*, New jersey: Prentice Hall, Inc.
- [3] Kadir, A. dan A.S., 2013. *Teori dan Aplikasi Pengolahan Citra* Edisi 1. D. Hardjono, ed., Yogyakarta: Penerbit Andi.
- [4] Koproński, R., 2011. *Image Processing in Optical Coherence Tomography using matlab* D. hab. inż. Porwik, ed., Poland: University of Silesia.
- [5] Martin, A., Tosunoglu, S. & Street, W.F., 2012. Image processing techniques for machine vision. , pp.1–9.
- [6] Parker, J.R., 2011. *Algorithms for Image Processing and Computer Vision* Second Edi., indiana Polis, Indiana: Wiley Publishing, Inc.
- [7] Petrou, M., 2010. *Image Processing: The Fundamentals* First Edit., United Kingdom: John Wiley & Sons, Ltd.
- [8] Phillips, D., 2000. *Image Processing in C* Second Edi., Kansas: Miller-Freeman, Inc.
- [9] Pressman, Roger S., 2015. *Software Engineering: A Practitioner's Approach* 8th Edition, Michigan : Mc.Graw-Hill, Inc.
- [10] Rao, K.M.M., 2014. OVERVIEW OF IMAGE PROCESSING. *Readings in Image Processing*, pp.1–7.
- [11] Starck, J. & Murtagh, F., 2012. *Handbook of Astronomical Data Analysis*, New York: Springer-Verlag.
- [12] Toczek, T. & Mancini, S., 2011. Efficient Memory Management for Uniform and Recursive Grid Traversal. *Springer Science & Business*, 10.1007, p.25.
- [13] Zhou, H., Wu, J. & Zhang, J., 2010. *Digital Image Processing: Part I* First Edit., Sylvania: bookboon.