

APLIKASI PENGOLAHAN CITRA DIGITAL UNTUK IDENTIFIKASI UMUR POHON

Nadzir Zaid Munantri⁽¹⁾, Herry Sofyan⁽²⁾, Mangaras Yanu F⁽³⁾

Jurusan Teknik Informatika, Universitas Pembangunan Nasional "Veteran" Yogyakarta
Jl. Bababsari 2, Tambakbayan Caturtunggal, Depok, DIY, 55281, INDONESIA
e-mail : nadzirmunantri94@gmail.com⁽¹⁾

Abstract

The good quality of wood can be determined from several factors. One factor is the knowledge of the age of trees. Age of trees can be known through the circle of years on wood or also called growth. From the considerations above the formula issued is how to make a growth calculation system that is able to calculate by computer calculation, so we need an analysis to determine the age of the tree with growth. To support growth in wood one method that can be used is edge detection. The first stage in the process is grayscale which is done changing the RGB color image to gray, then the sobel edge detection process works to display the outline / edge of the image, then the calculation process is carried out with the formula.

Keywords : image processing, annual ring, growthring, edge detection, sobel

Kualitas kayu yang baik dapat ditentukan dari beberapa faktor. Salah satu faktor itu adalah dengan mengetahui usia pohon. Usia pohon dapat diketahui berdasarkan lingkaran tahun pada kayu atau disebut juga dengan growthring. Dari permasalahan diatas rumusan masalahnya adalah bagaimana membuat sistem penghitungan growthring yang mampu menghitung dengan perhitungan komputer, maka diperlukan suatu analisis untuk mengetahui umur pohon dengan growthring. Untuk mendeteksi growthring pada kayu salah satu metode yang bisa digunakan adalah menggunakan edge detection. Tahapan yang pertama dilakukan proses dilakukan adalah proses grayscale yang dimana berfungsi mengubah gambar warna RGB menjadi abu abu, kemudian proses edge detection sobel yang berfungsi menampilkan garis tepi/tepi gambar, kemudian dilakukan proses perhitungan dengan rumus.

Kata Kunci : pengolahan citra, lingkaran tahun, growthring, edge detection, sobel

1. PENDAHULUAN

1.1 Latar Belakang

Dalam kehidupan kita sehari-hari, kayu merupakan bahan yang sangat sering dipergunakan untuk tujuan penggunaan tertentu. Salah satu contohnya seperti meja, kursi, almari dan lain-lain. Oleh karena itu kayu banyak diperdagangkan belikan. Banyak masyarakat awam kurang wawasan/ pengetahuan dalam membeli kayu yang berkualitas bagus. Salah satu indikator kualitas kayu bagus adalah dengan mengetahui umur pohon tersebut. Masyarakat hanya beramsumsi bahwa kayu yang berukuran besar menganggap umurnya sudah tua. Pendapat ini belum tentu sepenuhnya benar maka akan berakibatnya dijumpai permasalahan dalam proses transaksi jual beli kayu, dimana seseorang calon pembeli yang hendak membeli kayu yang sudah tua dengan harga yang mahal ternyata mendapatkan kayu yang masih muda. Umur kayu dapat diketahui berdasarkan lingkaran tahun pada kayu atau disebut juga dengan *growthring*.

Maka dari itu berdasarkan latar belakang diatas, sehingga diperlukanlah sebuah analisis mengenai cara bagaimana mengetahui umur dari sebuah kayu tersebut secara digital dengan menggunakan komputer sehingga dapat mempermudah dan membantu permasalahan yang ada di atas. Langkah pertama yang dilalukan adalah dengan mengambil dengan gambar penampang batang pohon/kayu dengan kamera. Selanjutnya gambar yang telah diperoleh akan diproses menggunakan teknik pemrosesan gambar (*image processing*) pada komputer. Selanjutnya semua informasi yang dikumpulkan akan diuji dengan menggunakan metode *grayscale, edge detection, sobel*.

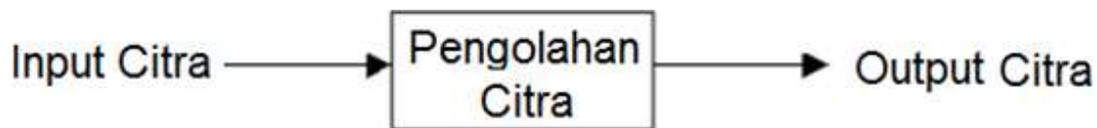
1.2 Landasan Teori

a. Citra Digital

Citra digital adalah gambar dua dimensi yang dihasilkan dari analog dua dimensi yang kontinu menjadi gambar melalui proses sampling. Gambar analog dibagi menjadi N baris dan M kolom sehingga menjadi gambar diskrit. Citra digital merupakan citra yang dapat diolah komputer. Yang disimpan dalam komputer hanyalah angka-angka yang menunjukkan besar intensitas pada masing-masing piksel. Karena berbentuk data numerik, maka citra digital dapat diolah dengan komputer.

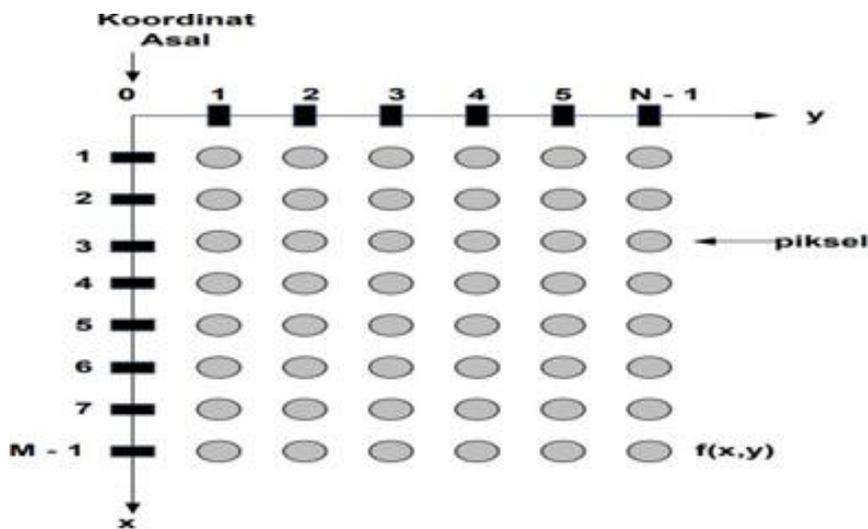
b. Pengolahan Citra Digital

Pengolahan citra digital adalah ilmu yang mempelajari hal-hal berkaitan dengan perbaikan kualitas terhadap suatu gambar (meningkatkan kontras, perubahan warna, restorasi citra), transformasi gambar (translasi, rotasi transformasi, skala, geometrik), melakukan pemilihan citra ciri (*feature images*) yang optimal untuk tujuan analisis, melakukan penyimpanan data yang sebelumnya dilakukan reduksi dan kompresi, transmisi data, dan waktu proses data. Adapun diagram sederhana dari proses pengolahan citra dapat dilihat pada gambar di bawah ini.



Gambar 1. Proses sederhana pengolahan citra

Secara umum pengolahan citra digital dapat diartikan sebagai pemrosesan gambar dua dimensi dengan menggunakan komputer. Citra digital merupakan sebuah *array* (larik) yang berisikan nilai-nilai *real* maupun kompleks yang dapat direpresentasikan dengan deretan *bit* tertentu.



Gambar 2. Koordinat dalam citra digital

Sebuah citra digital dapat diwakili oleh sebuah matriks yang terdiri dari M kolom dan N baris, dimana perpotongan antara kolom dan baris tersebut disebut dengan *pixel* (*picture element*) yang merupakan elemen terkecil dari sebuah citra. *Pixel* mempunyai dua parameter, yaitu koordinat dan intensitas atau warna. Nilai yang terdapat pada koordinat (x,y) adalah $f(x,y)$ yang merupakan besar intensitas atau warna dari *pixel* di titik itu.

$$f = \begin{bmatrix} f(1,1) & f(1,2) & \dots & f(1,N) \\ f(2,1) & f(2,2) & \dots & f(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ f(M,1) & f(M,2) & \dots & f(M,N) \end{bmatrix} \quad (1)$$

Berdasarkan gambaran diatas, maka secara matematis citra digital dapat dituliskan sebagai intensitas $f(x,y)$ dimana nilai x (baris) dan y (kolom) adalah koordinat posisi dan $f(x,y)$ adalah nilai fungsi pada setiap titik (x,y) yang menyatakan besar intensitas yang diterima oleh citra atau tingkat keabuan atau warna dari piksel di titik tersebut. Besarnya tergantung pada intensitas yang dipantulkan oleh objek. Ini berarti $f(x,y)$ sebanding dengan energi yang dipancarkan oleh sumber cahaya, sehingga besar intensitas $f(x,y)$.

c. Lingkaran Tahun

Lingkaran tahun adalah aktivitas dari pembelahan sel-sel yang ada di kambium. Maka dari itu, yang dapat membentuk lingkaran tahun hanya batang tumbuhan yang memiliki jaringan kambium. Sebagai contoh tumbuhan mahoni yang berasal dari kelompok tumbuhan dikotil. Kambium pada batang tumbuhan ini dapat membentuk lingkaran tahun. Lingkaran tahun adalah fenomena alam yang dialami suatu tumbuhan yang memiliki kambium. Dan dengan lingkaran tahun ini dapat menghitung perkiraan usia pohon itu sendiri.



Gambar 3. Penampang pohon

d. Grayscale

Proses yang banyak dilakukan dalam *image processing* adalah mengubah citra berwarna menjadi citra *grayscale*. Hal ini dilakukan dengan menyederhanakan model citra berwarna (RGB) yang terdiri dari 3 layer matrik yaitu R-layer, G-layer dan B-layer. Setiap proses perhitungan dilakukan menggunakan tiga layer. Artinya dilakukan tiga perhitungan yang sama pada ketiga layer tersebut. Untuk mendapatkan citra dalam bentuk *grayscale*, salah satu rumus yang dapat digunakan adalah sebagai berikut :

$$\text{Grayscale} = 0.3\text{RED} + 0.59\text{GREEN} + 0.11\text{BLUE} \quad (2)$$

e. Edge Detection

Edge Detection (deteksi tepi) adalah proses untuk menemukan perubahan intensitas yang berbeda nyata dalam sebuah bidang citra. Sebuah operator *edge detection* merupakan operasi ketetanggaan, yaitu sebuah operasi yang memodifikasi nilai keabuan sebuah titik berdasarkan nilai-nilai keabuan dari titik yang ada disekitarnya yang masing-masing mempunyai bobot tersendiri. Bobot tersebut nilainya tergantung pada operasi yang akan dilakukan. Sedangkan banyaknya titik disekitarnya yang terlibat biasanya adalah 2×2 , 3×3 , 3×4 , 7×7 dan sebagainya. Suatu *edge* adalah batas antara dua *region* yang memiliki *graylevel* yang relatif berbeda. Pada dasarnya ide yang ada di balik sebagian besar teknik *edge detection* adalah menggunakan perhitungan *local derivative* operator. Gradien dari suatu citra $f(x,y)$ pada lokasi (x,y) adalah vektor.

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (3)$$

Dalam *edge detection* nilai yang penting di sini adalah *magnitude* dari vektor, yang biasanya hanya disebut dengan gradien dan dituliskan dengan ∇f , dimana:

$$\nabla f \approx |G_x| + |G_y| \quad (4)$$

Rumus tersebut lebih mudah diimplementasikan, khususnya jika menggunakan *hardware* untuk pemrosesan. Arah dari vektor gradien juga merupakan kuantitas yang penting. Jika $\alpha(x,y)$ menunjukkan arah sudut vektor. ∇f pada (x,y) , maka dari analisa vektor:

$$\alpha(x, y) = \tan^{-1} \left(\frac{G_x}{G_y} \right) \quad (5)$$

f. Operator Sobel

Operator *Sobel* diperkenalkan oleh Irwin Sobel pada tahun 1970. Sobel merupakan metode *edge detection* yang termasuk dalam *gradient edge detector*. Operator ini identik dengan bentuk matriks 3x3 atau jendela ukuran 3x3 piksel. Piksel dari sebuah gambar yang akan dilakukan pendeteksian batas (*edge*) akan menjadi sebuah *edge* jika piksel tersebut melewati batas (*threshold*) tertentu. Sehingga apabila *threshold* telah ditetapkan maka nilai dari setiap piksel dapat kita tentukan apakah piksel tersebut merupakan batas atau bukan. Operator *Sobel* melakukan deteksi tepi dengan memperhatikan tepi *vertical* dan *horizontal*. Kelebihan dari metode *Sobel* ini adalah kemampuan untuk mengurangi *noise* sebelum melakukan perhitungan deteksi tepi.

Operator *Sobel* terdiri dari matriks 3x3 masing-masing adalah G_x dan G_y . Matriks *mask* tersebut dirancang untuk memberikan respon secara maksimal terhadap tepi objek baik horizontal maupun vertikal. *Mask* dapat diaplikasikan secara terpisah terhadap *input* citra. Operator *Sobel* menggunakan kernel operator gradien 3 x 3, dengan koefisien yang telah ditentukan dan dapat dinyatakan sebagai berikut.

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \text{ dan } G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (6)$$

Kernel di atas dirancang untuk menyelesaikan permasalahan deteksi tepi baik secara *vertical* maupun *horizontal*. Penggunaan kernel-kernel ini dapat digunakan bersamaan ataupun secara terpisah. Untuk mendapatkan nilai maksimum dari operator *Sobel*, proses selanjutnya adalah dengan menghitung kekuatan tepi citra terhadap warna kecerahannya dengan cara mencari nilai *magnitude* yang dapat dihitung dengan persamaan sebagai berikut :

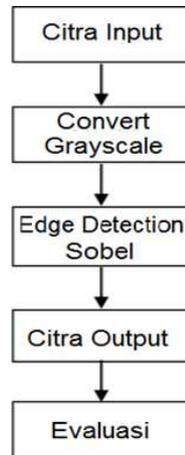
$$G = \sqrt{S_x^2 + S_y^2} \quad (7)$$

Dengan, G = Besar gradien operator *Sobel* S_x = gradien *sobel* arah horizontal S_y = gradien *Sobel* arah Vertikal Di mana G adalah besar gradien di titik tengah. Karena menghitung akar adalah persoalan rumit dan menghasilkan nilai *real*, maka dalam mencari kekuatan tepi (*magnitude*) dapat disederhanakan perhitungannya. Besarnya *magnitude* gradien dapat dihitung lebih cepat lagi dengan menggunakan persamaan sebagai berikut :

$$M = |G_x| + |G_y| \quad (8)$$

2. METODOLOGI PENELITIAN

Langkah-langkah yang ditempuh dalam penelitian ini dapat dilihat pada blog diagram berikut :



Gambar 4. Metodologi Penelitian

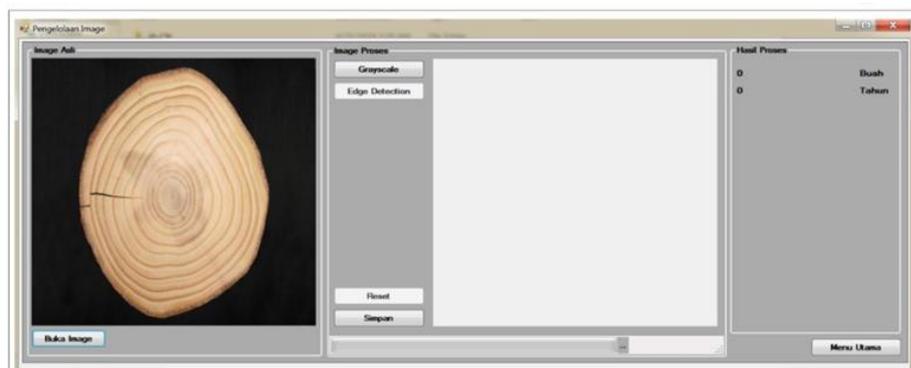
Terdapat lima langkah yang akan dilaksanakan dalam penelitian ini yaitu :

- 1) Langkah pertama adalah menentukan citra masukkan (*input*) yang akan dijadikan bahan uji coba.
- 2) Kemudian langkah berikutnya adalah melakukan konversi *colour* dari RGB ke *grayscale*. Maksud dari langkah ini adalah untuk memperoleh kualitas garis pemisah objek yang lebih baik.
- 3) Lalu berikutnya adalah melakukan *edge detection* dengan operator *sobel*. Tujuan dari tahapan ini adalah untuk mendapatkan batas antara satu obyek.
- 4) Kemudian adalah mendapatkan citra *output* dari setiap kernel yang digunakan. Tujuan dari tahapan ini adalah untuk mendapatkan citra keluaran (*output*) yang berupa objek yang akan dihitung pada tahap berikutnya.
- 5) Dan tahap akhir adalah melakukan evaluasi citra keluaran (*output*) yang telah didapatkan. Tujuan dari tahapan ini adalah untuk meneliti apakah ada perbedaan dari perhitungan program dan perhitungan manual.

3. HASIL DAN PEMBAHASAN

3.1 Input Citra

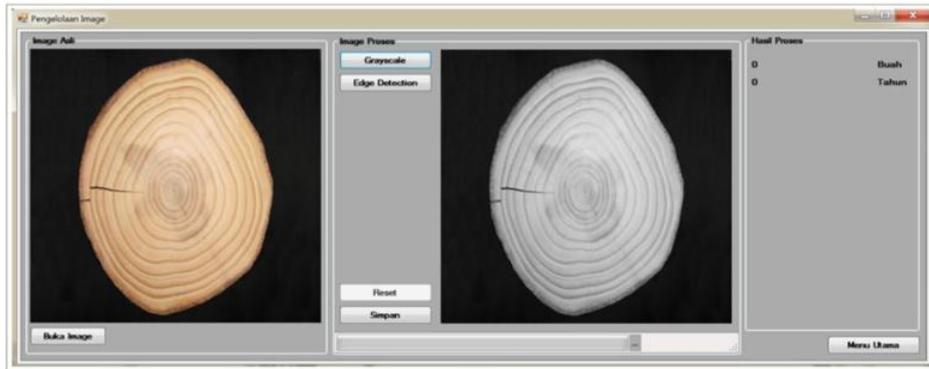
Sebelum melakukan proses segmentasi dilakukan, hal yang dilakukan terlebih dahulu adalah akuisisi citra, yaitu proses pengambilan *file* citra digital dari *drive* komputer yang akan disegmentasi.



Gambar 5. Proses input citra

3.2 Proses Grayscale

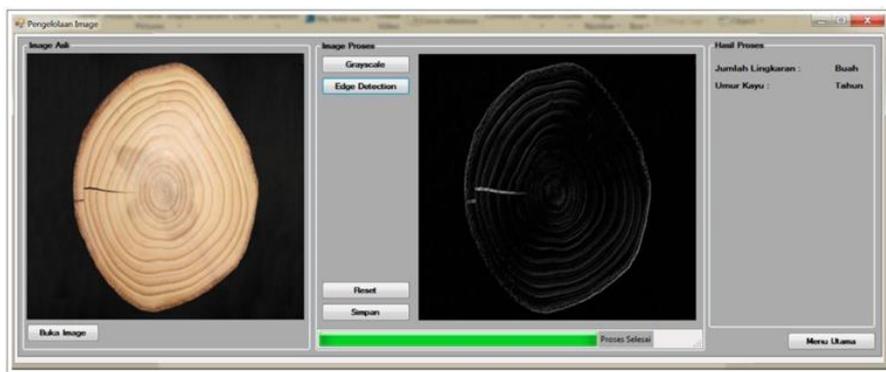
Dalam penelitian ini, pra-pemrosesan terdiri dari konversi. Citra penampang kayu berwarna (RGB) ke citra beraras keabuan (*grayscale*). Citra yang dihasilkan pada tahap masukan adalah citra berwarna sehingga perlu dikonversi ke citra *grayscale*. Berikut ini adalah hasil konversi citra RGB menjadi citra *grayscale*.



Gambar 6. Proses *Grayscale*

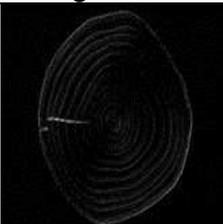
3.3 Proses Edge Detection

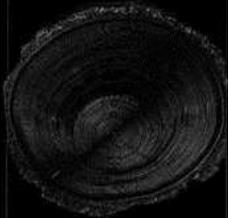
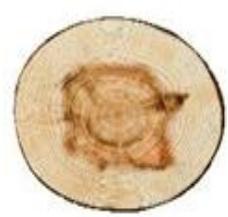
Proses deteksi tepi dengan metode sobel dilakukan pada semua citra penampang kayu berwarna *grayscale*. *Edge detection* berfungsi mendeteksi garis/tepi pada semua obyek yang ada di dalam gambar.



Gambar 7. Metodologi Proses Deteksi Tepi

Tabel 1. Uji Coba Hasil

No	Gambar Asli	Perhitungan Manual	Perhitungan Aplikasi	Error	Perkiraan Umur	Hasil Gambar Segmentasi
1		11	13	18,18 %	6 tahun	

No	Gambar Asli	Perhitungan Manual	Perhitungan Aplikasi	Error	Perkiraan Umur	Hasil Gambar Segmentasi
2		22	25	13,6%	13 tahun	
3		57	67	17,54%	34 tahun	
4		56	60	7,1%	30 tahun	
5		54	56	3,7%	32 tahun	
6		21	23	9,5%	11 tahun	
7		66	64	3%	32 tahun	
8		37	47	27.0%	23 tahun	

No	Gambar Asli	Perhitungan Manual	Perhitungan Aplikasi	Error	Perkiraan Umur	Hasil Gambar Segmentasi
9		23	22	4,34%	11 tahun	

Tingkat keberhasilan dari metode yang diterapkan pada pengolahan citra ini dilakukan pengujian dengan mencari perbandingan antara hasil program dengan hasil perhitungan secara manual.

$$\text{Keakuratan} = \frac{S}{PM} \times 100 \% \quad (9)$$

Keterangan : S = Selisih antara perhitungan manual dan sistem.
PM = Perhitungan manual.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, maka diambil kesimpulan berikut :

1. Aplikasi pengolahan citra ini memiliki pengaturan proses yang sederhana, sehingga *user* dapat dengan mudah mempelajari teknik dari pengolahan citra.
2. Kekurangan dari program adalah tidak dapat melakukan pengeditan gambar.
3. Metode *Sobel* adalah satu dari beberapa metode *edge detection* untuk mendeteksi tepi dari sebuah *image*.
4. Diperlukan waktu yang diperlukan pada proses pengolahan citra berbeda, semakin besar ukuran matriks dan ukuran file, maka waktu yang diperlukan akan semakin lama.
5. Implementasi dari aplikasi ini sudah bekerja dengan baik sesuai dengan perancangan.

DAFTAR PUSTAKA

- Dolly Indra. 2016. Pendeteksian Tepi Objek Menggunakan Metode Gradien, Jurnal Jurnal Ilmiah Edisi I, Volume 8 Nomor 2, Agustus 2016, ISSN: 2087-1716.
- Munawaroh, Siti, "Deteksi Growthring Pada Kayu Dengan Metode Edge Linking", 2012,
- Munir, R., Pengolahan Citra Digital dengan Pendekatan Alogaritmik, Informatika, Bandung, 2004
- Panji Stephanus, Shinta Estri Wahyuningrum, "Perhitungan Lingkaran Tahun Dengan Edge Linking Dan Chain Code" 2017.
- Tri Septia Prihartini, Pulung Nurtantio Andono, "Deteksi Tepi Dengan Metode Laplacian Of Gaussian Pada Citra Daun Tanaman Kopi".