

Pendeteksian Goresan pada Citra Menggunakan Segmentasi dengan Edge Detection

Vincent Christian Siregar and 13520136
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung

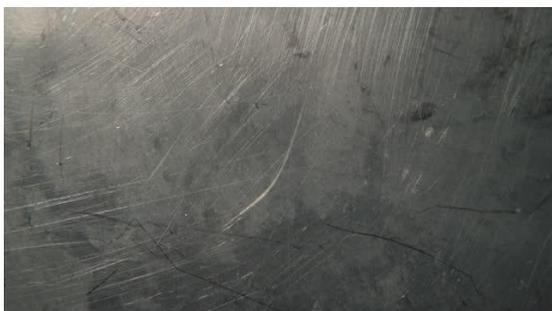
Abstract—This electronic document is a “live” template and already defines the components of your paper [title, text, heads, etc.] in its style sheet. ***CRITICAL: Do Not Use Symbols, Special Characters, or Math in Paper Title or Abstract.** (Abstract)

Keywords—component; formatting; style; styling; insert (key words)

I. PENDAHULUAN

Pengembangan dan penggunaan pemrosesan citra semakin banyak digunakan di kehidupan sehari-hari. Beberapa teknologi aplikasi pemrosesan citra seperti pendeteksian wajah, pendeteksian huruf pada plat nomor, sudah umum digunakan sehari-hari.

Seiring dengan kemajuan teknologi di bidang citra digital, pendeteksian goresan menjadi salah satu peluang pemanfaatan pemrosesan citra. Pada tahap awal perkembangan teknologi citra, fokus utama terletak pada upaya untuk meningkatkan resolusi dan reproduksi gambar. Namun, seiring berjalannya waktu, kebutuhan akan analisis yang lebih mendalam terhadap informasi visual mengarah pada pengembangan metode deteksi yang lebih canggih.



Gambar I. *Scratch* pada permukaan besi

Dalam perindustrian, penjaminan kualitas dari produk merupakan tahap yang penting. Salah satu parameter yang mempengaruhi kualitas dari suatu produk terutama produk yang berbentuk fisik adalah kecacatan pada permukaan produk. Kecacatan pada permukaan seperti goresan (*scratch*), keretakan (*ding*), dan penyok (*dent*) dapat mengurangi kepuasan pelanggan dalam menggunakan produk.

Dengan demikian, makalah ini akan mengeksplorasi dan membandingkan berbagai algoritma pendeteksian tepi, termasuk pendekatan seperti Canny dan Sobel. Pendeteksian tepi

II. LANDASAN TEORI

A. Citra Digital

Citra adalah sinyal dwimatra yang bersifat *continuous* yang dapat diamati oleh manusia. Citra digital direpresentasikan secara matematis dalam bentuk fungsi dwimatra yang merepresentasikan intensitas cahaya pada bidang dwimatra. Fungsi membentuk sebuah *array* 2 dimensi seperti pada gambar berikut.

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, N-1) \\ f(1,0) & f(1,1) & \dots & f(1, N-1) \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1, N-1) \end{bmatrix}$$

Gambar II.1 Ilustrasi Larik Dwimatra

Citra digital membentuk sebuah gambar yang dapat diinterpretasikan oleh mata manusia. Interpretasi citra adalah kegiatan menganalisis gambar sebelum atau sesudah diolah untuk tujuan tertentu seperti mengidentifikasi objek atau peran dalam gambar.

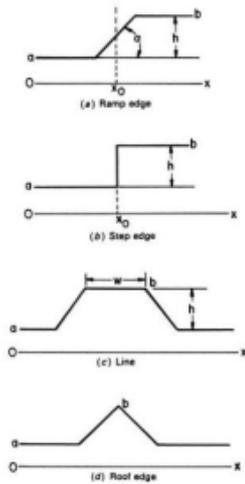
B. Pengolahan Citra

Pengolahan citra atau *image processing* adalah proses memodifikasi citra setelah diinterpretasi oleh manusia. Pengolahan citra adalah proses membuat citra baru dari citra lama untuk mendapatkan kualitas yang lebih baik. Pengolahan citra digital adalah operasi pemrosesan sinyal menggunakan operasi-operasi Komputer

C. Edge Detection

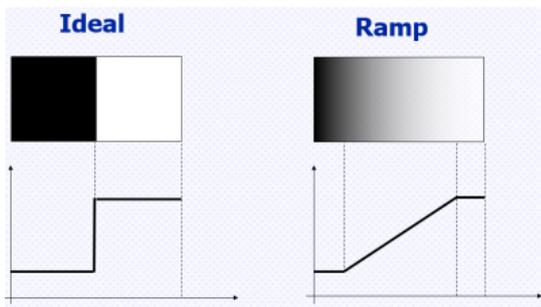
Edge atau tepi adalah perubahan nilai intensitas nilai keabuan yang mendadak dalam jarak waktu yang singkat.

Perubahan tepi memiliki arah yang berbeda-beda bergantung pada perubahan intensitas.



Gambar II.2 Ilustrasi Tepi

Tepi yang ideal memiliki perubahan intensitas yang tajam, tetapi tepi pada umumnya memiliki perubahan intensitas yang landai.



Gambar II.3 Perubahan intensitas citra pada tepi

Pendeteksian tepi ditujukan untuk menampakkan garis yang berada pada perubahan intensitas. Pendeteksian tepi mengekstraksi gambar menjadi garis-garis dalam citra. Terdapat beberapa metode deteksi tepi yang dapat diterapkan, antara lain,

1. Operator Gradien

Pendeteksian tepi dapat didekatkan dengan pendekatan kalkulus diffensial. Perubahan tajam pada citra digambarkan oleh kemiringan yang tajam. Formula turunan pertama dari citra dapat menggunakan turunan parsial untuk sumbu x dan sumbu y seperti yang tertera pada gambar II.4.

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right] = [G_x, G_y]$$

$$G_x = \frac{\partial f(x, y)}{\partial x} = \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x}$$

$$G_y = \frac{\partial f(x, y)}{\partial y} = \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y}$$

Gambar II.4 Formula pendeteksian tepi turunan pertama pada citra

Penerapan turunan pertama pada citra dapat di terapkan menggunakan konvolusi menggunakan dua buah mask seperti contoh berikut.

2. Operator Laplace

Operator Laplace menggunakan turunan kedua atau turunan dari turunan pertama. Operator ini menggunakan turunan mundur atau *backward differential*. Berikut adalah formula dari operator laplace.

$$G_3(x) = \frac{\partial^2 f(x, y)}{\partial x^2} = \frac{f(x, y) - f(x - \Delta x, y)}{\Delta x^2}$$

$$G_3(y) = \frac{\partial^2 f(x, y)}{\partial y^2} = \frac{f(x, y) - f(x, y - \Delta y)}{\Delta y^2}$$

$$\Delta x = \Delta y = 1$$

Gambar II.5 Formula pendeteksian tepi operator laplace pada citra

3. Operator Sobel

Operator Sobel adalah operasi gradien yang juga menggunakan turunan parsial. Turunan parsial dirumuskan dengan S_x dan S_y yang diformulakan seperti Formula pada gambar II.6. Nilai c pada formula merupakan konstanta.

$$s_x = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6)$$

$$s_y = (a_0 + ca_1 + a_{22}) - (a_6 + ca_5 + a_4)$$

Gambar II.6 Formula turunan parsial operasi sobel

Nilai a pada formula bersesuaian dengan pengaturan pixel pada gambar II.7. Nilai (x,y) merupakan titik tengah dari konvolusi elemen konvolusi.

$$\begin{bmatrix} a_0 & a_1 & a_2 \\ a_7 & (x, y) & a_3 \\ a_6 & a_5 & a_4 \end{bmatrix}$$

Gambar II.7 Pengaturan pixel untuk operasi sobel

4. Operator Canny

Operator Canny merupakan operasi gradien yang dapat menghasilkan ketebalan tepi berukuran 1 pixel.

Langkah-langkah operator Canny adalah sebagai berikut:

- Citra dihaluskan dengan menggunakan penapis Gaussian dengan standar deviasi yang dispesifikasikan.
- Gradien dan arah gradien setiap pixel dihitung dengan salah satu operator gradien.
- Jika nilai mutlak (magnitudo) gradien suatu pixel melebihi nilai ambang T, maka pixel tersebut termasuk pixel tepi.

Operator Canny menggunakan dua nilai ambang T yaitu T1 dan T2 ($T1 < T2$). Tepi yang dihasilkan ada dua jenis tepi yaitu tepi kuat (*strong edges*) dan tepi lemah (*weak edges*).

D. Citra Biner

Citra Biner merupakan citra yang hanya memiliki dua buah nilai keabuan yaitu 0 dan 1. Nilai graylevel 0 merupakan warna hitam dan graylevel 1 adalah putih. Citra biner digunakan untuk melakukan operasi logika (AND, OR, dan NOT). Citra biner akan digunakan dalam segmentasi citra.

E. Segmentasi citra

Segmentasi citra adalah pemisahan objek dari gambar latar belakang. Segmentasi citra banyak digunakan dalam pendeteksian objek seperti pada pendeteksian objek. Metode yang digunakan dalam melakukan segmentasi pada makalah ini adalah menggunakan menggunakan pendeteksian contour tertutup yang dihasil



Gambar II.8 Contoh segmentasi citra untuk memisahkan objek dari latar

F. OpenCV dan Python3

Dalam mengimplementasikan algoritma pemrosesan citra, pengembang menggunakan kaskas OpenCV dengan Bahasa pemrograman Python3. OpenCV merupakan kaskas *open-source* yang digunakan untuk membantu pengembang dalam mengembangkan aplikasi pendeteksi goresan. Library OpenCV digunakan untuk mengimplementasikan implementasi operasi

A. Kontour

III. TAHAPAN IMPLEMENTASI

Beberapa tahapan implementasi perlu dilakukan dalam pendeteksian goresan pada permukaan. Tahapan implementasi terdiri dari:

- Mengubah citra input dalam bentuk *grayscale*.
- Menerapkan operasi pendeteksian tepi.
- Mengubah hasil pendeteksian tepi menjadi citra biner.
- Melakukan perubahan morfologi dengan algoritma *closing* pada citra biner.
- Menerapkan algoritma *floodfill* dan menggunakan inversi *floodfill*.
- Melakukan segmentasi dengan mencari beberapa *countour* pada citra biner.

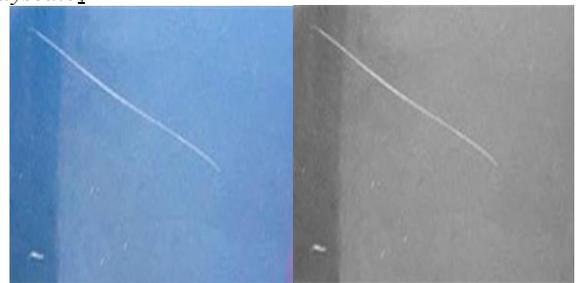
Langkah 4,5, dan 6 digunakan untuk mengisi ruang kosong yang berada di dalam garis yang membentuk tepi.

IV. EKSPERIMEN

A. Mengubah input gambar menjadi grayscale

Pada tahap ini, gambar ditransformasi menjadi citra *grayscale*.

Berikut adalah Beberapa contoh hasil pendeteksian citra *grayscale*.



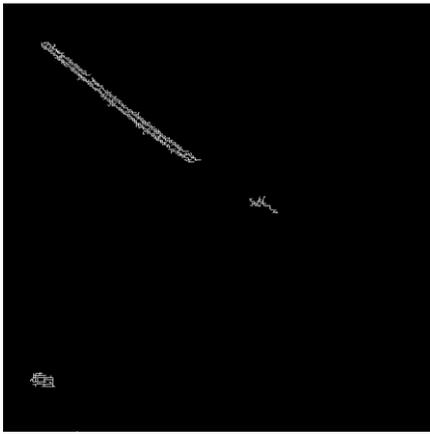
Gambar V.1 Contoh citra 1 hasil perubahan *grayscale*.
Gambar input (kiri) gambar *grayscale* (kanan)



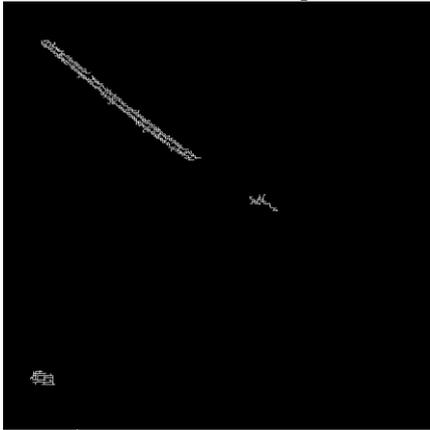
Gambar V.2 Contoh citra 2 hasil perubahan *grayscale*.
Gambar input (kiri) gambar *grayscale* (kanan)

B. Menerapkan operasi pendeteksian tepi.

Pada tahap ini, digunakan operasi *canny* dalam melakukan pendeteksian tepi,



Gambar V.3 Contoh citra 1 hasil pendeteksian tepi.

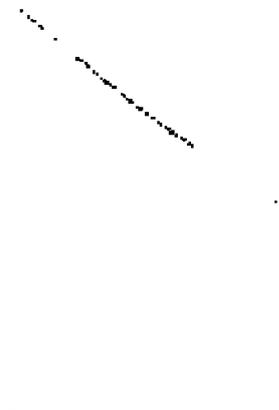


Gambar V.4 Contoh citra 2 hasil pendeteksian tepi.

C. Melakukan perubahan morfologi dengan algoritma closing pada citra biner.

Gambar V.5 Contoh citra 2 hasil pendeteksian tepi.

D. Menerapkan algoritma floodfilling dan menerapkan inversi floodfill

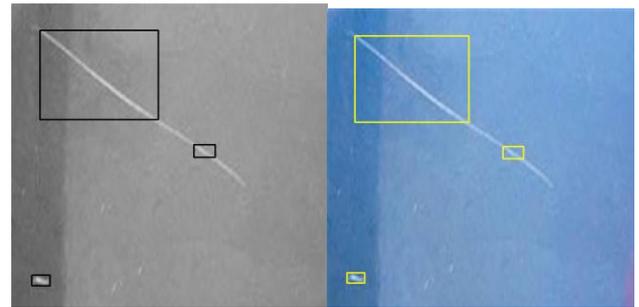


Gambar V.6 Contoh citra 2 hasil floodfill



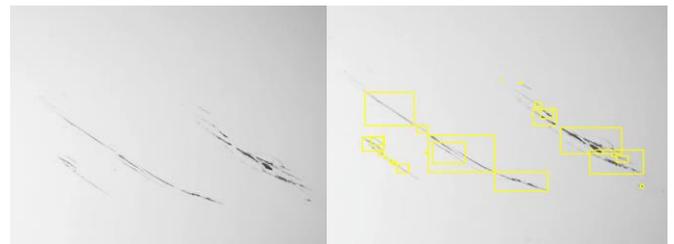
Gambar V.7 Contoh citra 2 hasil inversi floodfill

E. Melakukan segmentasi dengan mencari beberapa countour pada citra biner.



Gambar V.8 Contoh Pendeteksian Countour

F. Hasil pendeteksian contour lainnya



Gambar V.8 Contoh Pendeteksian Countou

V. PEMBAHASAN

Dari hasil eksperimen, penggunaan *edge detection* atau pendeteksian tepi dapat digunakan untuk pendeteksian goresan pada permukaan benda. Namun ketepatan pendeteksian segmentasi belum baik untuk kasus goresan yang Panjang atau banyak goresan yang berhimpit yang menyebabkan deteksi box kurang tepat menyelimuti goresan secara utuh.

VI. KESIMPULAN

Edge detection atau pendeteksian tepi merupakan salah satu hasil perkembangan pengolahan citra yang dapat digunakan untuk mendeteksi keberadaan objek pada gambar. *Edge detection* dengan segmentasi menggunakan kontur dapat digunakan untuk kasus pendeteksian goresan. Namun

beberapa hasil pendeteksian goresan tidak menghasilkan pendeteksian

Bandung, 19 November 2023

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.



Vincent Christian 13520136