

Deteksi Tepi pada Pengolahan Citra Digital Menggunakan Matlab

Amri¹, Egi Fransisco Saputra², M. Farel Antonio³ & Sinyo Kasanova⁴

^{1 2 3 4}Informatika, Fakultas Ilmu computer & rekayasa, Universitas Multi Data Palembang
¹amri.0205@mhs.mdp.ac.id, ²siegik@mhs.mdp.ac.id, ³muhammadfarel42@mhs.mdp.ac.id, ⁴sinyocasanova@mhs.mdp.ac.id

Kata Kunci:

Matlab ; Deteksi Tepi, Metode Pengolahan Citra

Abstrak: Deteksi tepi pada Citra dengan Menggunakan Matlab adalah suatu proses menghasilkan tepi objek gambar. Deteksi tepi bertujuan untuk menandai detail bagian dari citra dan memperbaiki detil citra yang kabur akibat kesalahan atau pengaruh dari proses akuisisi citra. Nilai perbedaan intensitas diperoleh dengan mengkonvolusikan citra dengan nilai *operator block mask* seperti *Gradient, Prewitt, Compass, dan Sobel*. Teknologi *image enhancement* dirancang untuk memproses gambar sehingga hasil aplikasi tertentu memiliki kualitas yang lebih baik daripada gambar aslinya.

Amri,Saputra,Antonio,Kasanova(2022). *Deteksi Tepi pada Pengolahan Citra Digital Menggunakan Matlab*. MDP Student Conference 2022

PENDAHULUAN

Representasi dan pemodelan gambar, peningkatan kualitas gambar, pemulih gambar, analisis citra, rekonstruksi citra, dan beberapa metode kompresi gambar melakukan pengolahan citra digital. Dalam makalah ini, pemrosesan gambar digital focus pada teknologi peningkatan kualitas pada bidang ruang, khususnya yang digunakan dalam teknologi deteksi tepi.

Tepi akan terjadi perubahan nilai intensitas abu-abu yang cepat dan besar dalam waktu singkat. [1] Deteksi tepi adalah serangkaian metode matematika yang digunakan untuk identifikasi titik-titik pada gambar, sehingga warna kecerahannya akan mengalami perubahan dengan tingkat kecerahan yang berbeda.

[2] Deteksi tepi gambar adalah suatu prosen menciptakan tepi yang menjadi tujuan pada objek gambar yang ditandai dari bagian detail citra dan memperbaiki detail dari gambar citra yang kabur. [3]. Yang mejadi metode dalam makalah ini adalah metode operator gradien, kompas, prewitt dan sobel.

METODE

Proses deteksi tepi (*edge detection*) dapat dikelompokkan berdasarkan operator atau metode yang diterapkan dalam proses pendeteksian tepi suatu citra untuk memperoleh citra hasil. Metode-metode atau operator yang digunakan adalah:

A. Metode Gradient

Gradien adalah turunan pertama yang dihitung sebagai akar pangkat dari jumlah kuadrat dari dua derivatif (x dan y) dan dinotasikan secara matematis sebagai persamaan 1. [4]

$$) ^2 \quad \text{Gradien} = \sqrt{\left(\frac{\partial}{\partial x}\right)^2 + \left(\frac{\partial}{\partial y}\right)^2}$$

Turunan parsial pertama fungsi citra $I(x,y)$ terhadap sumbu x (horizontal) dan sumbu y (vertikal) diperoleh persamaan 2 gradien citra arah x , persamaan 3 gradien citra arah y dan total gradien citra persamaan 4 [5].

$$\nabla I(x, y) = \frac{\partial I(x, y)}{\partial x} = I(x+1, y) - I(x, y) \quad (2)$$

$$\nabla I(x, y) = \frac{\partial I(x, y)}{\partial y} = I(x, y+1) - I(x, y) \quad (3)$$

$$\nabla I(x, y)$$

0	1	1
-1	0	-1
-1	-1	0

$$\nabla I(x, y) = \sqrt{(\nabla I(x, y))_x^2 + (\nabla I(x, y))_y^2} \quad (4)$$

B. Metode Kompas

Operator kompas (*compass operator*) biasa digunakan untuk mendeteksi semua tepi dari berbagai arah di dalam citra. Operator kompas yang digunakan menampilkan tepi dari 8 macam arah mata angin: Utara, Timur Laut, Timur, Tenggara, Selatan, Barat Daya, dan Barat Laut. Pendeteksian tepi dilakukan dengan mengkonvolusikan citra dengan berbagai mask kompas, kemudian dicari nilai kekuatan tepi (*magnitude*) yang terbesar dan arahnya. Jika misalnya digunakan sebanyak p buah mask kompas dan nilai kekuatan tepi pada pixel (x, y) untuk semua mask adalah $G1[f(x,y)]$, $G2[f(x,y)]$, ..., $Gp[f(x,y)]$.

C. Metode Prewitt

Metode Prewitt adalah pengembangan dari metode robert dengan menggunakan filter HPF yang diberi satu angka nol penyangga. Metode prewitt mengambil prinsip dari fungsi *laplacian* yang dikenal sebagai fungsi untuk membangkitkan HPF.

D. Metode Sobel

Metode Sobel merupakan pengembangan metode robert dengan menggunakan filter HPF yang diberi satu angka nol penyangga. Metode sobel menggunakan prinsip dari fungsi laplacian dan gaussian yang dikenal sebagai fungsi untuk membangkitkan HPF. Kelebihan dari metode sobel ini adalah kemampuannya yang dapat mengurangi noise sebelum melakukan perhitungan deteksi tepi. Operator Sobel menggunakan kernel operator gradient 3×3 :

(a)

1	2	1
0	0	0
-1	-2	-1

(b)

1	0	-1
2	0	-2
1	0	-1

Operator Sobel melakukan deteksi tepi dengan memperhatikan tepi vertical dan horizontal. Gradient Magnitude dari operator Sobel adalah sebagai berikut :

$$\begin{aligned}
 I_{xx} &= [I(x-1, y-1) + 2I(x-1, y) \\
 &\quad + I(x-1, y+1) \\
 &\quad - [I(x+1, y-1) \\
 &\quad + 2I(x+1, y) + I(x+1, y+1)]] \\
 I_{yy} &= [I(x-1, y-1) + 2I(x, y-1) + I(x+1, y-1)] - [I(x-1, y+1) \\
 &\quad + 2I(x, y+1) \\
 &\quad + I(x+1, y+1)]
 \end{aligned}$$

HASIL DAN PEMBAHASAN

A Metode Gradient

Dengan menggunakan bantuan program matlab maka deteksi tepidengan operator Gradient dapat dilihat implementasinya. Berikut ini adalah listing program yang akan digunakan untuk mengimplementasikan deteksitepi dengan operator Gradient:

Listing programnya :

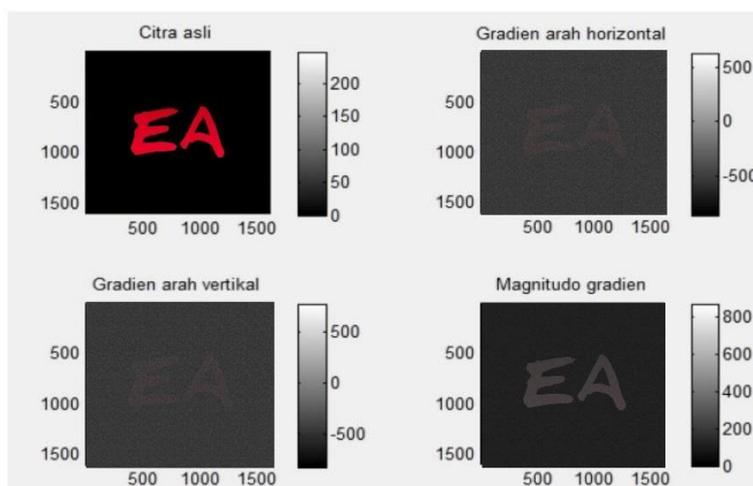
```

I = double(rgb2gray(imread('A.jpg')));
% Gradien orde satu pada arah horizontal
gx = [-1 1];
I1 = conv2(I, gx, 'same');
% Gradien orde satu pada arah vertikal
gy = [-1;1];
I2 = conv2(I, gy, 'same');
% Magnitudo gradien
J = sqrt((I1.^2)+(I2.^2));
% Gambar hasil
axes(ax1)
imagesc(I), axis image, colormap gray, colorbar, title('Citra asli');
axes(ax2)
imagesc(I1), axis image, colormap gray, colorbar, title('Gradien arah horizontal');
axes(ax3)
imagesc(I2), axis image, colormap gray, colorbar, title('Gradien arah vertikal');
axes(ax4)
imagesc(J), axis image, colormap gray, colorbar, title('Magnitudo gradien');

ax5 = axes('Parent', tab2, 'Position', [.08 .53 .25 .4]);
ax6 = axes('Parent', tab2, 'Position', [.4 .53 .25 .4]);
ax7 = axes('Parent', tab2, 'Position', [.72 .53 .25 .4]);
ax8 = axes('Parent', tab2, 'Position', [.08 .1 .25 .4]);
ax9 = axes('Parent', tab2, 'Position', [.4 .1 .25 .4]);
ax10 = axes('Parent', tab2, 'Position', [.72 .1 .25 .4]);

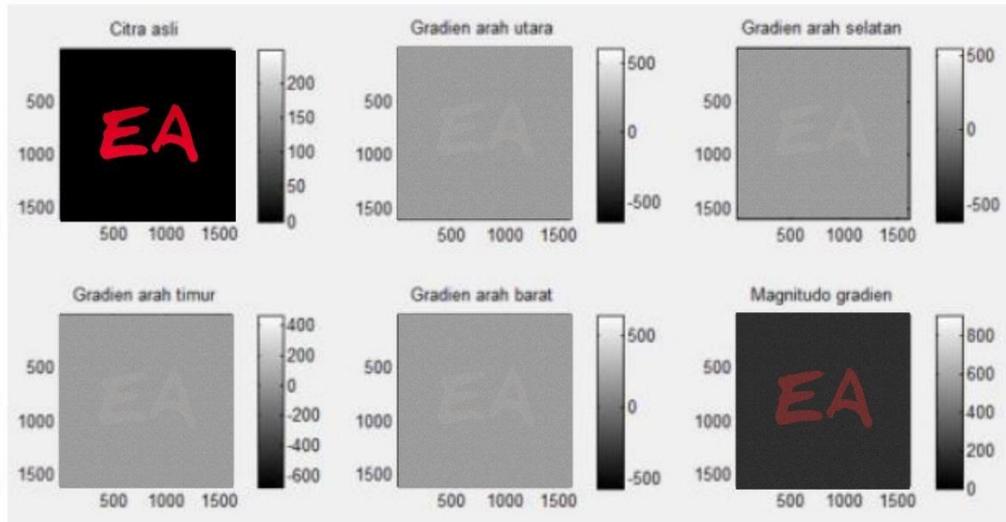
```

Hasil running program Matlabnya :



Gambar 1. Hasil Metode Gradien

B. *Metode Kompas*
 Hasil programnya :



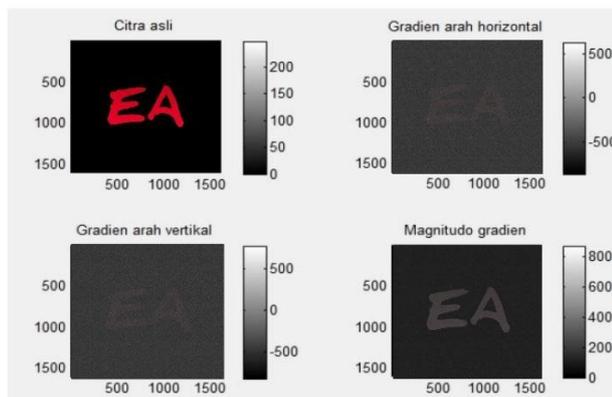
Gambar 2. Hasil Metode Kompas

C. *Metode Prewitt*
 Listing programnya :

```
I = double(rgb2gray(imread('A.jpg')));
%Konvolusi dengan operator Prewitt
prewittHor = [-1 0 1; -1 0 1; -1 0 1];
prewittVer = [-1 -1 -1; 0 0 0; 1 1 1];
I1 = conv2(I,prewittHor,'same');
I2 = conv2(I,prewittVer,'same');
J = sqrt((I1.^2)+(I2.^2));
%Gambar Hasil
axes(ax15)
imagesc(I),axis image,colormap gray,colorbar,title('Citra asli');
axes(ax16)
imagesc(I1),axis image,colormap gray,colorbar,title('Gradien arah horizontal');
axes(ax17)
imagesc(I2),axis image,colormap gray,colorbar,title('Gradien arah vertikal');
axes(ax18)
imagesc(J),axis image,colormap gray,colorbar,title('Magnitudo gradien');

ax19 = axes('Parent',tab5,'Position',[.2 .53 .25 .4]);
ax20 = axes('Parent',tab5,'Position',[.55 .53 .25 .4]);
ax21 = axes('Parent',tab5,'Position',[.2 .1 .25 .4]);
ax22 = axes('Parent',tab5,'Position',[.55 .1 .25 .4]);
```

Hasil running program Matlabnya :



Gambar 3. Hasil Metode Prewitt

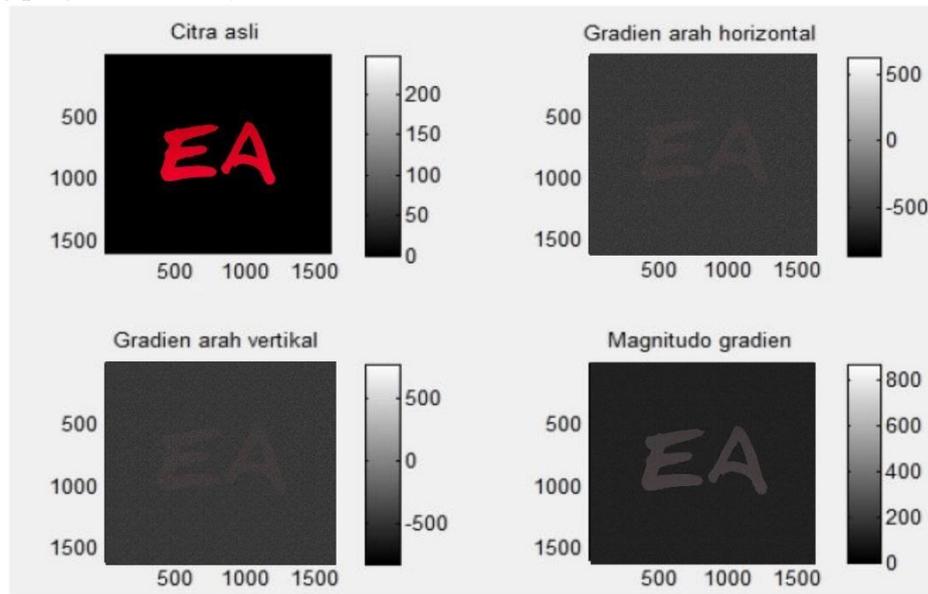
D. Metode Sobel

Listing Programnya :

```
I = double(rgb2gray(imread('A.jpg')));
%Konvolusi dengan operator Sobel
sobelhor = [-1 0 1; -2 0 2; -1 0 1];
sobelver = [-1 -2 -1; 0 0 0; 1 2 1];
I1 = conv2(I,sobelhor,'same');
I2 = conv2(I,sobelver,'same');
J = sqrt((I1.^2)+(I2.^2));
%Gambar Hasil
axes(ax19)
imagesc(I),axis image,colormap gray,colorbar,title('Citra asli');
axes(ax20)
imagesc(I1),axis image,colormap gray,colorbar,title('Gradien arah horizontal');
axes(ax21)
imagesc(I2),axis image,colormap gray,colorbar,title('Gradien arah vertikal');
axes(ax22)
imagesc(J),axis image,colormap gray,colorbar,title('Magnitudo gradien');

ax23 = axes('Parent',tab6,'Position',[.1 .23 .35 .6]);
ax24 = axes('Parent',tab6,'Position',[.55 .23 .35 .6]);
```

Hasil running program Matlabnya :



Gambar 4. Hasil Metode Sobel

SIMPULAN

Deteksi tepi bertujuan untuk menandai detail bagian dari citra dan memperbaiki detail citra yang kabur akibat kesalahan atau pengaruh dari proses akuisisi citra. Jika suatu titik sangat berbeda dengan tetangganya, maka titik ini disebut tepi bayangan. Nilai perbedaan intensitas diperoleh dengan mengkonvolusikan citra dengan nilai *operator block mask* seperti *Gradient*, *Compass*, *Prewitt*, dan *Sobel*. Berbagai operator deteksi memberikan hasil deteksi tepi yang berbeda, sehingga operator apa yang akan digunakan tentunya bergantung pada kebutuhan pengguna. Teknologi *image enhancement* dirancang untuk memproses gambar sehingga hasil aplikasi tertentu memiliki kualitas yang lebih baik daripada gambar aslinya.

DAFTAR PUSTAKA

- [1] Pitas, I., 1993, Digital Image Processing Algorithms, Prentice Hall, Singapore.
- [2] Fauzi, Yulian. 2005. Implementasi Algoritma Filtering Derivatif Dalam Mengolah Citra Satelit Pada Software Envi, Jurnal Gradien Vol.1 No.2, Maret 2005, pp:81-86.
- [3] Madenda. S. 2015. Pengolahan Citra & Video Digital: Teori, Aplikasi dan Pemrograman Menggunakan Matlab, Erlangga, Jakarta.
- [4] Deteksi Tepi Modul 8, http://informatika.stei.itb.ac.id/~rinaldi.munir/Buku/Pengolahan%20Citra%20Digital/Bab-8_Pendeteksian%20Tepi.pdf, Diakses pada 6 desember 2021.
- [5] Danil, Cristopher, Edge Detection dengan Algoritma Canny, STMIK IBBI, <http://dosen.publikasistmikibbi.lppm.org/document>, diakses tanggal 6 desember 2021.
- [6] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Englewood Cliffs, NJ, USA: Prentice Hall,