

Deteksi Tepi dalam Pengolahan Citra Digital

Andi Sofyan Anas¹, Ahmad Ashril Rizal²

^{1,2}STMik Bumigora Mataram, Jalan Ismail Marzuki Mataram

¹andi_anas@ymail.com, ²ashril.rizal@gmail.com

Abstrak

Deteksi tepi (*Edge Detection*) merupakan suatu proses untuk menghasilkan tepi-tepi dari obyek-obyek citra. Tujuan deteksi tepi adalah untuk menandai bagian yang menjadi detail citra dan memperbaiki detail dari citra yang kabur karena error atau efek dari proses akuisisi citra. Suatu titik dikatakan sebagai tepi suatu citra apabila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya. Nilai perbedaan intensitas didapatkan melalui konvolusi citra dengan nilai block mask operator seperti Gradien, Kompas, Prewitt, Sobel dan lainnya. Teknik peningkatan mutu citra (*image enhancement*) bertujuan untuk melakukan pemrosesan terhadap citra supaya hasilnya mempunyai kualitas relatif lebih baik dari citra awal untuk aplikasi tertentu.

Kata kunci—Deteksi Tepi, Peningkatan Mutu Citra, Block Mask Operator

1. PENDAHULUAN

Representasi dan pemodelan citra, peningkatan kualitas citra, restorasi citra, analisis citra, rekonstruksi citra dan kompresi citra adalah beberapacara untuk melakukan pengolahan citra digital. Pada makalah ini, pengolahan citra digital difokuskan pada teknik peningkatan mutu pada domain spatial khususnya penggunaan untuk teknik deteksi tepi (*Edge Detection*).

Tepi merupakan terjadinya perubahan nilai intensitas derajat keabuan yang cepat dan besar dalam jarak singkat [1]. Deteksi tepi adalah seperangkat metode matematis yang digunakan untuk mengidentifikasi titik-titik dalam citra digital dimana warna kecerahannya mengalami perubahan secara drastis atau tingkat kecerahannya berbeda [2]. Deteksi tepi (*Edge Detection*) suatu citra adalah proses yang menghasilkan tepi-tepi dari objek-objek citra dengan tujuan untuk menandai bagian yang menjadi detail citra dan untuk memperbaiki detail dari citra yang kabur [3]. Metode yang akan digunakan dalam makalah ini adalah

metode operator gradien, kompas, prewitt, sobel, robert dan canny.

2. METODE PENELITIAN

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 Gradien

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].

$$\text{Gradien} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (1)$$

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].

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

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

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*) 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)], maka besar kekuatan tepi adalah:

Jika mask k adalah mask yang memberikan kekuatan terbesar, maka arah tepi ditentukan dari mask k tersebut. Operator kompas yang dipakai untuk pendeteksian tepi menampilkan tepi dari 8 macam arah mata angin:

Utara	Timur Laut	Timur	Tenggara
$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$
Selatan	Barat Daya	Barat	Barat Laut
$\begin{bmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{bmatrix}$

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. Operator

Prewitt menggunakan 8 buah *kernel* operator gradien :

(a)

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

(b)

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

(c)

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

(d)

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

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 adalah kemampuannya dapat mengurangi noise sebelum melakukan perhitungan deteksi tepi. Operator Sobel menggunakan kernel operator gradient 3 x 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 :

$$G_x = [f(i-1, j-1) + 2f(i-1, j) + f(i-1, j+1)] - [f(i+1, j-1) + 2f(i+1, j) + f(i+1, j+1)]$$

$$G_y = [f(i-1, j-1) + 2f(i, j-1) + f(i+1, j-1)] - [f(i-1, j+1) + 2f(i, j+1) + f(i+1, j+1)]$$

$$G[f(x, y)] = \sqrt{G_x^2 + G_y^2}$$

a. Metode Robert

Metode Robert merupakan nama lain dari teknik differensial pada arah horizontal dan differensial pada arah vertikal, dengan ditambahkan proses konversi biner setelah dilakukan differensial. Teknik konversi biner yang disarankan yaitu konversi biner dengan meratakan distribusi warna hitam dan putih.. Metode Robert juga disamakan dengan teknik DPCM (*Differential Pulse Code Modulation*). Operator gradient berukuran 2 x 2 digunakan sebagai operator Robert:

1	1
-1	-1

Gradient magnitude dari operator Robert adalah:

$$G[f(i, j)] = [f(i, j) + f(i+1, j+1)] + [f(i, j+1)]$$

Operator Robert sangat sensitif terhadap *noise* karena hanya menggunakan *convolution mask* berukuran 2x2.

b. Metode Canny

John F. Canny pada tahun 1986 mengembangkan metode canny dengan menggunakan algoritma multi-

tahap untuk mendeteksi berbagai tepi dalam gambar. Kategori algoritma yang dikembangkan adalah:

1. Deteksi: Kemungkinan mendeteksi tepi yang benar harus dimaksimalkan sedangkan kemungkinan mendeteksi tepi yang salah harus diminimalkan. Hal ini bertujuan untuk memaksimalkan rasio *signal to noise*.
2. Lokalisasi: tepi yang terdeteksi harus sedekat mungkin dengan tepi nyata.
3. Jumlah tanggapan: satu tepi nyata tidak harus menghasilkan lebih dari satu ujung yang terdeteksi.

3. HASIL DAN PEMBAHASAN

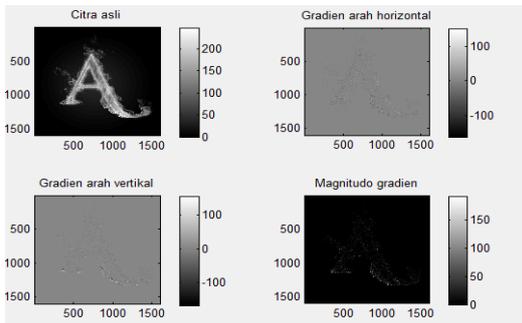
a. Metode Gradient

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

```
I = double(rgb2gray imread('R.jpg'));
% Gradient orde satu pada arah horizontal
gx = [-1 1];
I1 = conv2(I,gx,'same');
% Gradient 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,colorbar,title('Citra asli');
axes(ax2)
imagesc(I1),axis image,colorbar,title('Gradient arah horizontal');
axes(ax3)
imagesc(I2),axis image,colorbar,title('Gradient arah vertikal');
axes(ax4)
imagesc(J),axis image,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 adalah:



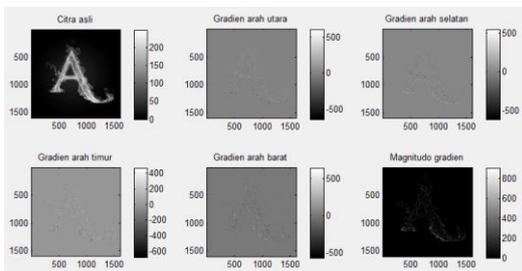
b. Metode Kompas

Listing program matlab:

```
I = double(rgb2gray(imread('A.jpg')));
%Konvolusi dengan operator compass
I1 = [1 1 1; 1 -2 1; -1 -1 -1];
I2 = [-1 -1 -1; 1 -2 1; 1 1 1];
I3 = [-1 1 1; -1 -2 1; -1 1 1];
I4 = [1 1 -1; 1 -2 -1; 1 1 -1];
I1 = conv2(I,I1,'same');
I2 = conv2(I,I2,'same');
I3 = conv2(I,I3,'same');
I4 = conv2(I,I4,'same');
J = sqrt((I1.^2)+(I2.^2)+(I3.^2)+(I4.^2));
%Gambar Hasil
axes(ax5)
imagesc(I),axis image,colormap gray,colorbar,title('Citra asli');
axes(ax6)
imagesc(I1),axis image,colormap gray,colorbar,title('Gradien arah utara');
axes(ax7)
imagesc(I2),axis image,colormap gray,colorbar,title('Gradien arah selatan');
axes(ax8)
imagesc(I3),axis image,colormap gray,colorbar,title('Gradien arah timur');
axes(ax9)
imagesc(I4),axis image,colormap gray,colorbar,title('Gradien arah barat');
axes(ax10)
imagesc(J),axis image,colormap gray,colorbar,title('Magnitudo gradien');

ax11 = axes('Parent',tab3,'Position',[.2 .53 .25 .4]);
ax12 = axes('Parent',tab3,'Position',[.55 .53 .25 .4]);
ax13 = axes('Parent',tab3,'Position',[.2 .1 .25 .4]);
ax14 = axes('Parent',tab3,'Position',[.55 .1 .25 .4]);
```

Running programnya:

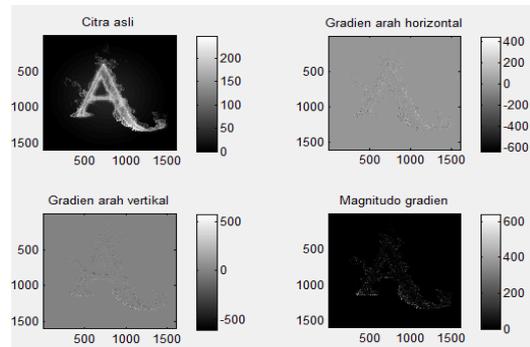


c. Metode Prewitt

Berikut ini adalah *listing* program dengan menggunakan metode prewitt:

```
I = double(rgb2gray(imread('A.jpg')));
%Konvolusi dengan operator Prewitt
prewittthor = [-1 0 1; -1 0 1; -1 0 1];
prewittver = [-1 -1 -1; 0 0 0; 1 1 1];
I1 = conv2(I,prewittthor,'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]);
```



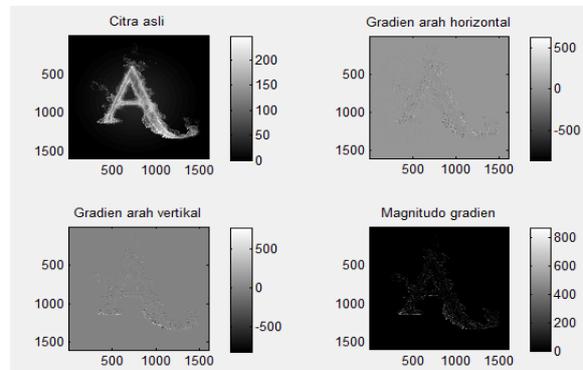
d. Metode Sobel

Listing program matlab dengan metode sobel adalah:

```
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]);
```

Hasilnya:



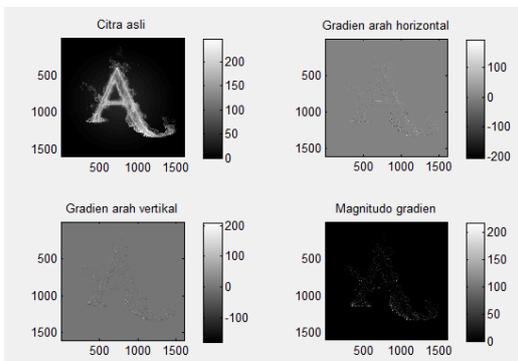
e. Metode Robert

Listing program menggunakan matlab untuk metode Robert:

```
I = double(rgb2gray(imread('A.jpg')));
% Konvolusi dengan operator Roberts
robertshor = [0 1; -1 0];
robertsver = [1 0; 0 -1];
I1 = conv2(I,robertshor,'same');
I2 = conv2(I,robertsver,'same');
J = sqrt((I1.^2)+(I2.^2));
% Gambar Hasil
axes(ax11)
imagesc(I),axis image,colormap gray,colorbar,title('Citra asli');
axes(ax12)
imagesc(I1),axis image,colormap gray,colorbar,title('Gradien arah horizontal');
axes(ax13)
imagesc(I2),axis image,colormap gray,colorbar,title('Gradien arah vertikal');
axes(ax14)
imagesc(J),axis image,colormap gray,colorbar,title('Magnitudo gradien');

ax15 = axes('Parent',tab4,'Position',[.2 .53 .25 .4]);
ax16 = axes('Parent',tab4,'Position',[.55 .53 .25 .4]);
ax17 = axes('Parent',tab4,'Position',[.2 .1 .25 .4]);
ax18 = axes('Parent',tab4,'Position',[.55 .1 .25 .4]);
```

Hasilnya:

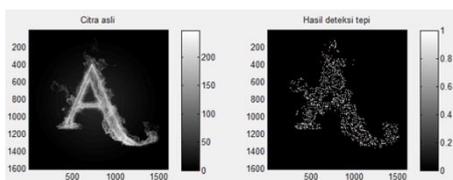


a. Metode Canny

Program Matlab menggunakan metode canny:

```
I = double(rgb2gray(imread('A.jpg')));
J = edge(I,'canny');
axes(ax23)
imagesc(I),axis image,colormap gray,colorbar,
axes(ax24)
imagesc(J),axis image,colormap gray,colorbar,
```

Hasilnya:



4. KESIMPULAN

Dari hasil dan analisis dapat diambil kesimpulan yaitu:

1. Gambar dengan tepian nyata yang baik dapat dihasilkan dengan metode operator gradient, kompas, prewitt, sobel, robert dan canny.
2. Kesalahan dalam pendeteksian noise dan variasi warna sebagai tepi semakin kecil dengan menggunakan metode yang berbeda-beda.
3. Detail dari citra yang kabur yang terjadi karena *error* atau adanya efek dari proses akuisisi citra dapat diperbaiki.

5. SARAN

Masih terdapat banyak kekurangan dalam makalah ini, diantaranya kesulitan untuk membandingkan hasil akhir dari setiap metode yang digunakan, sehingga kedepannya diharapkan dapat dilakukan perbaikan.

DAFTAR PUSTAKA

[1] Danil, Cristopher, Edge Detection dengan Algoritma Canny, STMIK IBBI, <http://dosen.publikasistmikibbi.lppm.org/document>, diakses tanggal 19 Oktober 2017.

[2] Pitas, I., 1993, Digital Image Processing Algorithms, Prentice Hall, Singapore.

[3] Kusban, Muhammad, 2013, Deteksi Tepi Bidang Kedokteran dalam Kawasan Alihragam Powerlaw, <https://publikasiilmiah.ums.ac.id/bitstream/handle/11617/3055/Makalah.pdf?sequence=1>, diakses tgl 30 Mei 2017.

[4] 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.

- [5] Madenda. S. 2015. Pengolahan Citra & Video Digital : Teori, Aplikasi dan Pemrograman Menggunakan Matlab, Erlangga, Jakarta. Digital/Bab-8_Pendeteksian%20Tepi.pdf, diakses tgl 30 Mei 2017.
- [6] Deteksi Tepi Modul 8, <http://informatika.stei.itb.ac.id/~rinaldi.munir/Buku/Pengolahan%20Citra%20>