

---

## Deteksi Tepi Pola Tulisan Arab Menggunakan Metode *Canny* pada Nisan Kuno di Sumatera Utara

Sriani<sup>\*1</sup>, Mhd. Furqan<sup>2</sup>, Rizky Sundari TP. Bolon<sup>3</sup>

<sup>1,2,3</sup>Program Studi Ilmu Komputer, Universitas Islam Negeri Sumatera Utara Medan

Email: <sup>1</sup>[sriani@uinsu.ac.id](mailto:sriani@uinsu.ac.id), <sup>2</sup>[mfurqan@uinsu.ac.id](mailto:mfurqan@uinsu.ac.id), <sup>3</sup>[rizkysundari@uinsu.ac.id](mailto:rizkysundari@uinsu.ac.id)

Email Penulis Korespondensi: <sup>1</sup>[sriani@uinsu.ac.id](mailto:sriani@uinsu.ac.id)

---

### Article History:

Received Dec 03<sup>th</sup>, 2022

Revised Dec 20<sup>th</sup>, 2022

Accepted Jan 03<sup>th</sup>, 2023

### Abstrak

Nisan merupakan salah satu peninggalan sejarah Islam di Sumatera Utara. Pada sisi nisan-nisan tersebut terdapat tulisan arab, tetapi ukiran dan tulisan arab pada nisan kuno tersebut mulai tidak terlihat disebabkan faktor lingkungan dan faktor usia. Untuk membantu agar pola tulisan arab dapat terlihat diterapkan proses deteksi tepi (*edge detection*) yang merupakan salah satu operasi utama pengenalan pola pada Pengolahan Citra. Dalam penelitian ini, deteksi tepi yang penulis gunakan adalah metode Canny. Metode Canny terkenal sebagai operator deteksi tepi yang optimal karena memberikan tingkat kesalahan yang rendah, melokalisasi titik-titik tepi, dan hanya memberikan satu tanggapan untuk satu tepi. Citra yang diuji melalui enam tahapan metode Canny yaitu penapisan noise (*filtering*), mendapatkan kekuatan tepi (*edge strength*), penghitungan arah tepi, menghubungkan arah tepi, *non-maximum suppression*, dan *hysteresis* untuk mendapatkan citra hasil. Berdasarkan sistem yang telah dibuat dengan menerapkan metode Canny dan pengujian terhadap beberapa data citra nisan kuno, dihasilkan citra digital yang terlihat batasan antara objek dengan latar belakang berupa tepi objek dengan jelas.

**Kata Kunci:** Deteksi Tepi, Metode Canny, Nisan Kuno, Pola Tulisan Arab, Pengolahan Citra.

---

### Abstract

*Tombstone is one of the historical heritage of Islam in North Sumatra. On the sides of the tombs there are Arabic inscriptions, but the carvings and Arabic inscriptions on the ancient tombs are starting to disappear due to environmental factors and age factors. To help the pattern of Arabic writing can be seen, an edge detection process is applied which is one of the main pattern recognition operations in Image Processing. In this study, the edge detection that the authors use is the Canny method. The Canny method is known as the optimal edge detection operator because it provides a low error rate, localizes edge points, and only gives one response for one edge. The images tested were carried out through six stages of the Canny method, namely filtering, obtaining edge strength, calculating edge directions, connecting edge directions, non-maximum suppression, and hysteresis to obtain the resulting image. Based on the system that has been created by applying the Canny method and testing some ancient gravestone image data, a digital image is produced that clearly shows the boundaries between objects with a background in the form of object edges.*

**Keywords:** *Edge Detection, Canny Method, Ancient Tombstones, Arabic Writing Patterns, Image Processing.*

---

## 1. PENDAHULUAN

Citra merupakan fungsi menerus (*continue*) atas intensitas cahaya pada bidang dua dimensi. Sebuah citra adalah suatu representasi, kemiripan atau imitasi dari suatu obyek atau benda [13]. Objek yang direpresentasikan oleh sebuah citra memiliki banyak informasi dan manfaat dari penyebaran intensitas cahaya yang ada pada bidang dua dimensi [5].

Citra merupakan salah satu komponen yang berperan penting sebagai suatu bentuk dari informasi visual [10]. Selain itu, citra merupakan salah satu bentuk informasi yang diperlukan manusia selain teks, suara dan video [11].

Dengan bantuan teknologi informasi, proses digitalisasi citra nisan kuno diharapkan mampu menjaga informasi yang terkandung di dalamnya. Masing-masing objek citra memiliki nilai perbedaan yang dapat diperhitungkan secara matematis, sehingga menunjukkan ciri yang berbeda antara objek yang satu dengan yang lain[9]. Pengolahan citra digital adalah teknik mengolah citra yang bertujuan memperbaiki kualitas citra agar mudah direpresentasikan [3].

Di Sumatera Utara, terdapat sebuah daerah bernama Barus yang kaya dengan tinggalan budaya keagamaan. Salah satu tinggalan budaya yang masih dapat disaksikan adalah makam-makam kuno yang terdapat inkripsi-inkripsi peradaban masa lalu. Tulisan/inkripsi yang ada sebagai bukti sejarah masuknya Islam[7]. Deteksi tepi (Edge Detection) merupakan suatu proses untuk menghasilkan tepi-tepi dari obyek-obyek citra tersebut. Tujuan deteksi tepi adalah untuk menandai bagian yang menjadi detail citra[2]. Dalam penelitian ini, deteksi tepi digunakan adalah metode *Canny*.

Algoritma Canny adalah suatu algoritma deteksi tepi yang dilakukan dengan pendekatan konvolusi terhadap fungsi matriks gambar dan operator Gaussian[12]. Pada algoritma Canny akan mencari titik tepi yang berada di atas ambang batas pertama dan berada di titik ambang batas pertama dan kedua[1].

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Metode Canny merupakan salah satu metode deteksi tepi yang optimal. Hal ini dikarenakan pada metode canny operator yang digunakan adalah Gaussian Derivative Kernel yang dapat menyaring kegaduhan dalam citra digital menggunakan rumus sebagai berikut [4].

$$\frac{1}{k} + \sum_{p=0}^{n-1} \left( \sum_{q=0}^{m-1} G(p, q) \cdot \text{piksel } A \left( i + p \frac{(n-1)}{2}, j + q \frac{(m-1)}{2} \right) \right) \quad (1)$$

Setelah penghalusan gambar terhadap *noise* dilakukan, dilakukan proses untuk mendapatkan kekuatan tepi (*edge strength*). Untuk mendapatkan tepian vertikal dan horizontal nilai citra uji akan dikalikan dengan operator Sobel yang memiliki operator matriks 3 x 3 sebagai berikut [7].

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

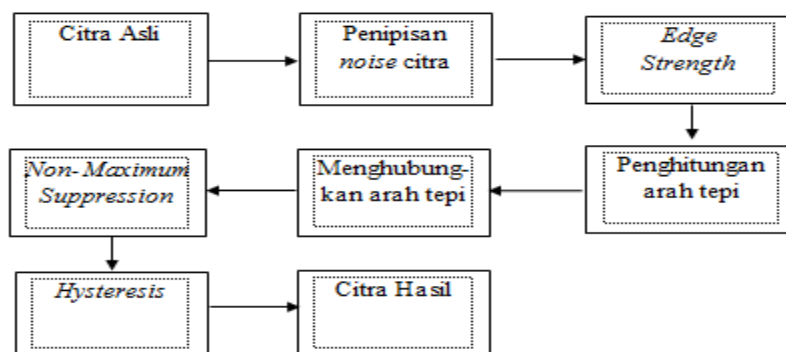
$$|G| = |G_x| + |G_y| \quad (3)$$

Langkah selanjutnya berupa penghitungan arah tepi dan perkalian menggunakan operator Sobel dengan matrik 3 x 3.. Rumus yang digunakan untuk keperluan ini adalah sebagai berikut [7].

$$\theta = \arctan \left( \frac{G_y}{G_x} \right) \quad (4)$$

Setelah arah tepi diperoleh, perlu menghubungkan antara arah tepi dengan sebuah arah yang dapat dilacak dari citra dan dilakukan penghilangan non-maximal. Non-maximal suppression bertujuan mengubah tepi yang kabur pada citra menjadi tepi yang tajam[6]. Selanjutnya dilakukan pengambangan (*thresholding*) menggunakan dua *threshold* T1 dan T2. T1 sebagai ambang bawah dan T2 sebagai ambang atas, dan nilai yang lebih besar dari T2 dianggap sebagai tepi yang kuat [8].

Penelitian ini dilakukan untuk mendigitalisasi tepi pola tulisan Arab yang terdapat di dalam nisan kuno peninggalan sejarah Sumatera Utara dengan menggunakan deteksi tepi metode *Canny*. Adapun diagram perencanaan dalam penerapan deteksi tepi metode *Canny* Utara terdapat pada Gambar 1.



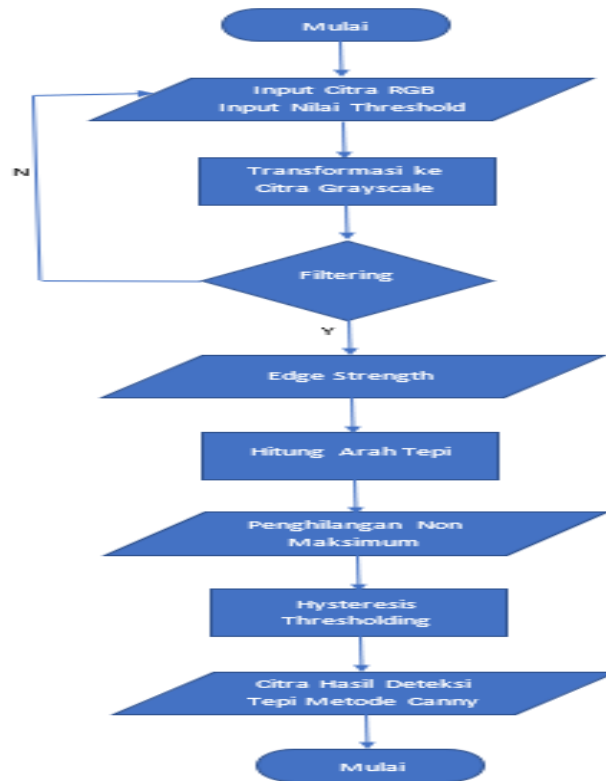
Gambar 1. Diagram Pendeteksian Tepi Metode *Canny*

### 3. HASIL DAN PEMBAHASAN

Hasil dan pembahasan deteksi tepi pola tulisan arab pada nisan kuno menggunakan metode Canny meliputi perancangan, analisis data dan implementasi sebagai berikut:

#### 3.1 Perancangan

Perancangan *flowchart* deteksi tepi menggunakan metode *Canny* seperti pada gambar 2 sebagai berikut:



Gambar 2. *Flowchart* Deteksi Tepi Metode *Canny*

#### a. Analisis Data

Penerapan metode *Canny* untuk deteksi tepi pola tulisan Arab menggunakan citra sampel RGB yang telah ditransformasi ke citra *grayscale* ukuran 8 x 8 piksel dengan bobot, sesuai pada gambar 3 sebagai berikut.

138	133	149	136	146	132	132	125
140	129	132	130	121	132	132	135
140	123	122	128	102	132	129	136
138	116	131	131	109	133	126	128
137	112	143	125	125	135	127	130
138	115	147	114	131	136	123	139
136	126	144	112	128	139	102	134
131	136	141	120	124	141	77	120

Gambar 3. Bobot Piksel Citra Sampel

Untuk memulai tahapan dari metode Canny, langkah pertama yang dilakukan dimulai dengan proses *filtering* dengan *mask Gaussian* dan diakhiri dengan tahap *hysteresis*.

1. Input citra RGB: Data yang berupa citra RGB dengan ukuran 8 x 8 kemudian ditransformasikan ke citra grayscale dengan kedalaman warna 8 bit dan rentang warna 0 – 255.
2. Filtering dengan mask Gaussian: Mask Gaussian yang berfungsi untuk penipisan terhadap citra dengan tujuan menghilangkan noise. Mask ini berukuran jauh lebih kecil daripada ukuran citra asli dan diinisialisasikan langsung ke dalam sistem. Selanjutnya citra sampel (A) dikonvolusikan dengan citra kernel Gauss (G) untuk mendapatkan citra perkalian B dengan menggunakan rumus sebagai berikut:

$$\frac{1}{k} + \sum_{p=0}^{n-1} \left( \sum_{q=0}^{m-1} G(p, q) \cdot \text{piksel A} \left( i + p \frac{(n-1)}{2}, j + q \frac{(m-1)}{2} \right) \right) \quad (1)$$

3. Menghitung kekuatan tepi: Untuk mendapatkan kekuatan tepi dapat menggunakan perkalian matriks citra hasil dengan konvolusi operator Sobel. Selanjutnya, gradien citra dapat dihitung melalui rumus berikut,

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

$$|G| = |G_x| + |G_y| \quad (3)$$

4. Menghitung arah tepi: Langkah selanjutnya berupa penghitungan arah tepi dan menyesuaikan arah tepi sesuai aturan konversi. Rumus yang digunakan untuk keperluan ini adalah,

$$\theta = \arctan \left( \frac{G_y}{G_x} \right) \quad (4)$$

5. Non-Maximum Suppression: Setelah arah tepi diperoleh, perlu menghubungkan antara arah tepi dengan sebuah arah yang dapat dilacak dari citra dan dilakukan penghilangan non-maximal. Non-maximal suppression bertujuan mengubah tepi yang kabur pada citra menjadi tepi yang tajam.
6. *Hysteresis*: *Hysteresis* Proses menghilangkan garis-garis yang seperti terputus-putus pada tepi objek yang dilakukan dengan memasukkan nilai ambang bawah dan ambang atas sehingga nantinya didapatkan citra hasil K.

### 3.2 Implementasi

Pada tahap menguji data sampel menggunakan citra sampel A dengan jenis RGB berukuran 8 x 8 piksel yang sebelumnya ditransformasikan ke dalam citra *grayscale* dengan bobot, sesuai pada gambar 4 sebagai berikut.

138	133	149	136	146	132	132	125
140	129	132	130	121	132	132	135
140	123	122	128	102	132	129	136
138	116	131	131	109	133	126	128
137	112	143	125	125	135	127	130
138	115	147	114	131	136	123	139
136	126	144	112	128	139	102	134
131	136	141	120	124	141	77	120

Gambar 4. Bobot Piksel Citra Sampel A

Selanjutnya citra sampel A dikenakan langkah metode Canny sebagai berikut,

- a. Filtering dengan Mask Gaussian: *Mask Gaussian* yang diinisialisasikan dalam sistem, seperti pada gambar 5 sebagai berikut.

1	2	1
2	3	2
1	2	1

Gambar 5. Matriks *Mask Gaussian* G dengan Kernel 3 x 3

Selanjutnya citra sampel (A) dikonvolusikan dengan citra kernel *Gauss* (G) tersebut untuk mendapatkan citra perkalian (B). Untuk mendapatkan setiap piksel citra perkalian B digunakan rumus sebagai berikut,

$$\text{Piksel B (i, j)} = \frac{1}{k} + \sum_{p=0}^{n-1} \left( \sum_{q=0}^{m-1} G(p, q) \cdot \text{piksel A} \left( i + p \frac{(n-1)}{2}, j + q \frac{(m-1)}{2} \right) \right) \quad (5)$$

Keterangan :

- Piksel A = Matriks gambar sampel
- Piksel B (i,j) = Bobot hasil perkalian pada posisi (i,j)
- n = Jumlah kolom matriks kernel
- m = Jumlah baris matriks kernel
- G(p,q) = Elemen matriks kernel *Gauss* pada posisi p,q
- K = Jumlah matriks kernel *Gauss*

Langkah perkalian citra sampel A dengan matriks kernel *Gauss* G jika diketahui bahwa bobot citra sampel A pada gambar 4 dan matriks kernel *Gauss* G pada gambar 5. Perkalian matriks citra sampel A dan matriks kernel *Gauss* G dilakukan untuk mendapatkan nilai citra perkalian (B) dengan cara mengalikan setiap 3 x 3 piksel citra sampel (A) dan mengganti nilai tengahnya dengan koordinat (2,2), (3,2) dan seterusnya. Hal ini dilakukan secara berulang-ulang melalui pergeseran satu kolom ke samping kanan, jika telah sampai di piksel yang paling kanan maka perkalian bergeser satu kolom ke bawah. Proses tersebut dapat dijabarkan pada gambar 6.

138	133	149	136	146	132	132	125
140	129	132	130	121	132	132	135
140	123	122	128	102	132	129	136
138	116	131	131	109	133	126	128
137	112	143	125	125	135	127	130
138	115	147	114	131	136	123	139
136	126	144	112	128	139	102	134
131	136	141	120	124	141	77	120

Gambar 6. Perkalian Citra Sampel A dengan Matriks *Gauss* G

Perhitungan untuk mendapatkan nilai piksel perkalian koordinat (2,2) adalah sebagai berikut :

$$(1 \times 138) + (2 \times 133) + (1 \times 149) + (2 \times 140) + (3 \times 129) + (2 \times 132) + (1 \times 140) + (2 \times 123) + (1 \times 122) = 1992$$

Sehingga hasil citra perkalian (B), seperti pada gambar 7 sebagai berikut.

138	133	149
140	1992	132
140	123	122

Gambar 7. Hasil Perkalian Citra Sampel A dengan Matriks *Gauss* G pada Koordinat (2,2)

Pada penelitian ini perkalian matriks citra sampel A dan matriks kernel Gauss G untuk mendapatkan citra perkalian B dapat dijabarkan dengan perhitungan tiga bobot pertama citra sampel A pada Tabel 1.

Tabel 1. Perkalian Citra Sampel A dan G

Koordinat Piksel dalam A	Perkalian Matriks A dan G	Nilai Titik Tengah Baru
(2,2)	$(1 \times 138) + (2 \times 133) + (1 \times 149) + (2 \times 140) + (3 \times 129)$ $+ (2 \times 132) + (1 \times 140) + (2 \times 123) + (1 \times 122)$	1992
(3,2)	$(1 \times 133) + (2 \times 149) + (1 \times 136) + (2 \times 129) + (3 \times 132)$ $+ (2 \times 130) + (1 \times 123) + (2 \times 122) + (1 \times 128)$	1976
(4,2)	$(1 \times 149) + (2 \times 136) + (1 \times 146) + (2 \times 132) + (3 \times 130)$ $+ (2 \times 121) + (1 \times 122) + (2 \times 128) + (1 \times 102)$	1943

Sehingga didapatkan citra hasil perkalian B yang memiliki nilai tengah baru. Bobot matriks perkalian B, dapat dilihat pada gambar 8.

138	133	149	136	146	132	132	125
140	1992	1976	1943	1911	1939	1977	135
140	1924	1900	1847	1812	1876	1967	136
138	1898	1905	1871	1829	1886	1945	128
137	1912	2035	1919	1889	1936	1945	130
138	1951	1948	1912	1910	1946	1915	139
136	1997	1968	1891	1907	1886	1788	134
131	136	141	120	124	141	77	120

Gambar 8. Matriks Citra Perkalian B

Untuk mendapatkan matriks gambar hasil maka bobot tengah matriks perkalian B dikenakan persamaan (5), dapat dilihat pada gambar 9.

$$\frac{1}{15} \times$$

138	133	149	136	146	132	132	125
140	1992	1976	1943	1911	1939	1977	135
140	1924	1900	1847	1812	1876	1967	136
138	1898	1905	1871	1829	1886	1945	128
137	1912	2035	1919	1889	1936	1945	130
138	1951	1948	1912	1910	1946	1915	139
136	1997	1968	1891	1907	1886	1788	134
131	136	141	120	124	141	77	120

Gambar 9. Matriks Citra Perkalian B yang Dikenakan Persamaan

Contoh perhitungan dengan contoh tiga bobot awal citra perkalian B untuk mendapatkan matriks gambar hasil dijabarkan pada Tabel 2.

Tabel 2. Perkalian Citra untuk Mendapatkan Citra Hasil K

$\frac{1}{K}$	Koordinat	Bobot Piksel	Hasil
1/15	(2,2)	1992	$132,8 \approx 133$
1/15	(3,2)	1976	$131,7 \approx 132$
1/15	(4,2)	1943	$129,5 \approx 130$

Hasil matriks B yang telah dikenakan persamaan (5) untuk mendapatkan matriks hasil dengan bobot baru, dapat dilihat pada gambar 10.

138	133	149	136	146	132	132	125
140	133	132	130	127	129	132	135
140	128	127	123	121	125	131	136
138	127	127	125	122	126	130	128
137	127	136	128	126	129	130	130
138	130	130	127	127	130	128	139
136	133	131	126	127	126	119	134
131	136	141	120	124	141	77	120

Gambar 10. Matriks Citra Perkalian B yang Telah Dikenakan Persamaan (5)

b. Menghitung Kekuatan Tepi (Edge Strength) dan Gradien: Mendapatkan kekuatan tepi dapat menggunakan perkalian matriks citra B yang telah dikenakan persamaan (5) dengan konvolusi operator Sobel. Konvolusi operator Sobel yang digunakan untuk mencari  $g_x$  sesuai gambar 11 dan  $g_y$  sesuai pada gambar 12 sebagai berikut:

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

Gambar 11. Matriks  $g_x$  Konvolusi Operator Sobe

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

Gambar 12. Matriks  $g_y$  Konvolusi Operator Sobel

Kemudian dilakukan proses menghitung gradien citra menggunakan rumus sebagai berikut,

$$G = \sqrt{G_x^2 + G_y^2} \tag{6}$$

$$|G| = |G_x| + |G_y| \tag{7}$$

Perkalian matriks citra perkalian B dengan konvolusi operator Sobel untuk mendapatkan nilai kekuatan tepi dan besar gradien citra tersebut dijabarkan dengan menggunakan contoh tiga bobot awal citra perkalian B. Perkalian matriks citra perkalian B dengan konvolusi operator Sobel untuk mendapatkan nilai kekuatan tepi dan besar gradien citra tersebut dijabarkan dengan menggunakan contoh tiga bobot awal citra perkalian B terdapat pada Tabel 3.

Tabel 3. Perkalian Citra B dan Konvolusi Operator Sobel

Koordinat Piksel dalam A	Perkalian Matriks A dan G	$g_x$	$g_y$	G
(2,2)	$g_x = (138 \times (-1)) + (133 \times 0) + (149 \times 1) + (140 \times (-2)) + (133 \times 0) + (132 \times 2) + (140 \times (-1)) + (128 \times 0) + (127 \times 1)$	-18	30	48
(3,2)	$g_y = (138 \times 1) + (133 \times 2) + (149 \times 1) + (140 \times 0) + (133 \times 0) + (132 \times 0) + (140 \times (-1)) + (128 \times (-2)) + (127 \times (-1))$	-8	62	70
	$g_x = (133 \times (-1)) + (149 \times 0) + (136 \times 1) + (133 \times (-2)) + (132 \times 0) + (130 \times 2) + (128 \times (-1)) + (127 \times 0) + (123 \times 1)$			



$$\begin{aligned}
 &gy = (133 \times 1) + (149 \times 2) + (136 \times 1) + (133 \times 0) + (132 \times 0) + \\
 &(130 \times 0) + (128 \times -1) + (127 \times -2) + (123 \times -1) \\
 (4,2) \quad &gx = (149 \times (-1)) + (136 \times 0) + (146 \times 1) + (132 \times (-2)) + (130 \times 0) \quad -19 \quad 73 \quad 92 \\
 &+ (127 \times 2) + (127 \times (-1)) + (123 \times 0) + (121 \times 1) \\
 &gy = (149 \times 1) + (136 \times 2) + (146 \times 1) + (132 \times 0) + (130 \times 0) + \\
 &(127 \times 0) + (127 \times (-1)) + (123 \times (-2)) + (121 \times (-1))
 \end{aligned}$$

Kekuatan tepi citra hasil B setelah dikonvulasikan dengan operator Sobel, dapat dilihat pada gambar 13.

138	133	149	136	146	132	132	125
140	48	70	92	74	46	131	135
140	64	36	42	22	46	44	136
138	40	30	50	24	42	20	128
137	30	14	40	20	28	24	130
138	28	18	26	10	8	48	139
136	24	54	34	26	94	114	134
131	136	141	120	124	141	77	120

Gambar 13. Matriks Citra B yang telah Dicari Kekuatan Tepi

c. Menghitung Arah Tepi: Langkah selanjutnya berupa penghitungan arah tepi menggunakan rumus berikut,

$$\theta = \arctan \left( \frac{Gy}{Gx} \right) \tag{8}$$

Setelah memperoleh arah tepi, maka arah tepi tersebut perlu dihubungkan dengan arah yang dapat dikenali oleh citra. Berikut adalah aturan konversi yang berlaku, jika diperoleh arah tepi antara 0 dan 22,5 serta 157,5 dan 180 derajat, arah tersebut diubah menjadi 0 derajat. Jika diperoleh arah tepi antara 22,5 dan 67,5 derajat, arah tersebut diubah menjadi 45 derajat. Jika diperoleh arah tepi antara 67,5 dan 112,5 derajat, arah tersebut diubah 90 derajat. Jika diperoleh arah tepi antara 112,5 dan 157,5 derajat, arah tersebut diubah menjadi 135 derajat. Arah tepi 3 bobot awal piksel citra B dapat dilihat pada Tabel 4.

Tabel 4. Arah Tepi Citra B dan Hasil Konversi

Koordinat Piksel B	gx	gy	gy/gx	Arah Tepi (θ)	Konversi
(1,1)	138	138		89, 584°	90°
(2,1)	133	133		89, 562°	90°
(3,1)	149	149		89, 615°	90°

d. *Non-Maximum Suppression*: Penghilangan non-maksimum adalah menghilangkan garis yang tidak diperlukan dalam sebuah citra dan dilakukan di sepanjang tepi. Penghilangan non-maksimum akan membandingkan gradien titik perhatian dengan gradien dua titik tetangga. Apabila gradien titik perhatian lebih besar dari gradien tetangga maka akan dianggap sebagai tepi, apabila gradien titik perhatian lebih kecil dari titik tetangga maka akan dihilangkan. Perbandingan nilai gradien titik perhatian dengan titik tetangga untuk tiga bobot awal citra B akan dijabarkan pada Tabel 5.

Tabel 5. Perbandingan Nilai Gradien Titik Perhatian dengan Titik Tetangga

Titik Perhatian (r,c)	Gradien	Arah Tepi (θ)	Titik Tetangga (r, c-1), (r, c+1)	Nilai
(1,1)	138	90°	(140)	0
(2,1)	133	90°	(48)	255
(3,1)	149	90°	(70)	255

Sehingga nilai non maksimum gradien titik perhatian setelah dibandingkan dengan kedua titik tetangga dapat dilihat pada gambar 14.



0	255	255	255	255	255	255	0
255	0	0	0	0	0	0	0
255	0	0	0	0	255	0	255
0	0	0	255	255	255	0	0
0	0	0	255	0	255	0	0
255	0	0	255	0	0	0	255
0	0	0	0	0	0	255	0
0	255	255	255	255	255	0	0

Gambar 14. Nilai Non Maksimum Gradien Titik Perhatian

Piksel yang telah melalui tahap penghilangan non-maksimum biasanya merupakan sebuah tepi dan ditandai sebagai piksel yang kuat. Tetapi ada kemungkinan bahwa piksel tersebut merupakan derai atau objek lain seperti *background* kasar dari sebuah objek. Untuk membedakan apakah piksel tersebut adalah tepi atau objek lain di dalam citra diperlukan proses *hysteresis*.

d. *Hysteresis*: *Hysteresis* adalah proses menghilangkan garis-garis yang seperti terputus-putus pada tepi objek yang dilakukan dengan memasukkan nilai ambang bawah dan ambang atas sehingga nantinya didapatkan citra hasil K. Dalam Sistem Penerapan Metode *Canny* untuk Deteksi Tepi Pola Tulisan Arab pada Nisan Kuno Peninggalan Sejarah Sumatera Utara ini, nilai ambang bawah dan ambang atas ditentukan oleh pengguna. Sementara dalam perhitungan data *sample A*, T1 (ambang bawah) yang digunakan adalah 0.05 dan T2 (ambang atas) yang digunakan adalah 0.15 dan akan dijabarkan di Tabel 6.

Tabel 6. Perbandingan Gradien dan Nilai Ambang

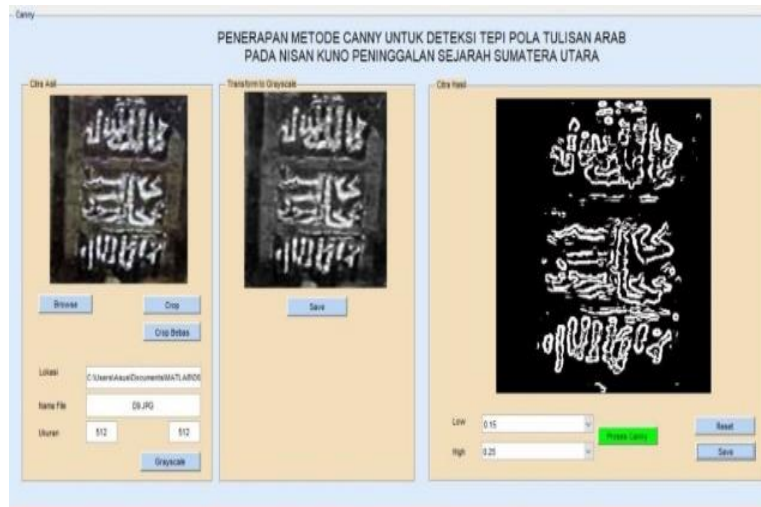
Titik (r,c)	Besaran Gradien	T1	T2	Nilai Akhir
(1,1)	138	0.05	0.15	255
(2,1)	133	0.05	0.15	255
(3,1)	149	0.05	0.15	255

Sehingga nilai akhir gradien titik perhatian setelah dibandingkan akan menghasilkan bobot citra hasil K, dapat dilihat pada gambar 15.

255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255

Gambar 15. Nilai Akhir Gradien

Sistem penerapan metode *Canny* untuk deteksi tepi pola tulisan Arab pada nisan kuno peninggalan sejarah Sumatera Utara dirancang menggunakan Matlab. Matlab merupakan *software* dengan bahasa pemrograman tingkat tinggi, memiliki kemampuan grafis yang baik, dan dipakai untuk menganalisis data-data, memvisualisasikan *tool* dan *prototyping* matematis dengan dukungan operasi matrik. Gambar 16 merupakan *form* uji dari sistem yang dibuat, pengujian sampel lainnya akan dijabarkan di Tabel 7.



Gambar 16. Sistem Deteksi Tepi Metode *Canny* Citra Nisan Kuno







Selanjutnya terdapat beberapa sampel yang digunakan untuk sistem deteksi tepi metode Canny, pengujian sampel-sampel tersebut akan dijabarkan pada Tabel 7.

Tabel 7. Pengujian Citra Nisan Kuno menggunakan Metode Canny

Nama File dan Ukuran	T1	T2	Citra Awal	Citra Hasil
<b>D1.jpg</b> 512 x 512	0,15	0,25		
<b>D2.jpg</b> 512 x 512	0,15	0,25		
<b>D3.jpg</b> 512 x 512	0,30	1		
<b>D4.jpg</b> 512 x 512	0,15	0,30		
<b>D5.jpg</b> 512 x 512	0,15	0,30		

---

---

<b>D8.jpg</b> 512 x 512	0,15	0,35		
<b>D9.jpg</b> 512 x 512	0,15	0,25		
<b>D15.jpg</b> 512 x 512	0,20	0,25		

---

Berdasarkan beberapa citra nisan kuno dengan pola tulisan arab yang telah diuji menggunakan salah satu operator deteksi tepi yaitu Metode *Canny*, terbukti bahwa deteksi tepi metode *Canny* dapat mendeteksi tepi objek dan membatasi objek tulisan arab dari *background* nisan kuno dengan baik.

#### 4. KESIMPULAN

Dari penelitian yang telah dilakukan maka diperoleh kesimpulan bahwa metode *Canny* merupakan salah satu operator deteksi tepi yang dapat digunakan untuk mendeteksi tepi pola tulisan arab pada nisan kuno dengan baik. Untuk melihat kejelasan dari tepi pola tulisan arab yang terdapat di nisan kuno, pengguna dapat memvariasikan dan menyesuaikan nilai ambang bawah (*low threshold*) dan nilai ambang atas (*high threshold*). Nilai *threshold* yang terlalu kecil menampilkan tepian-tepian yang terlalu banyak dan tidak dibutuhkan, sementara nilai *threshold* yang terlalu besar akan menyebabkan tepian-tepian tidak terdeteksi dan tidak dianggap sebagai tepi. Pada penelitian ini metode yang digunakan adalah untuk mendeteksi tepi pola tulisan Arab yang terdapat di dalam nisan kuno.

Saran untuk pengembangan penelitian selanjutnya agar sistem dapat menggunakan pengenalan pola (*pattern recognition*). Tujuan dari tahap ini untuk mengklasifikasikan dan mendeskripsikan pola huruf yang terkandung di dalam objek. Saran lainnya, untuk pengembangan penelitian selanjutnya diharapkan bisa mengkombinasikan metode *Canny* dengan menggunakan jaringan syaraf tiruan sehingga dapat mengenali jenis huruf yang terkandung sehingga fungsi dari sistem dapat lebih optimal.

#### UCAPAN TERIMA KASIH

*Bismillahirrahmanirrahim*, Puji dan syukur penulis ucapkan kepada Allah *Subhanahuwata'ala* yang telah memberikan kesehatan dan kesempatan kepada penulis sehingga penulis mampu menyelesaikan penelitian ini. Penulis juga mengucapkan terimakasih kepada keluarga, rekan satu profesi, dan institusi pendidikan penulis.

#### DAFTAR PUSTAKA

- [1] A. Kurniasari, Jalinus, "Pendeteksian Tingkat Kepadatan Jalan Menggunakan Metode Canny Edge Detection," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 25, no.3, hal. 239-248, 2020.
- [2] A.S. Anas, A.A. Rizal, "Deteksi Tepi dalam Pengolahan Citra Digital," *Proceeding Seminar Nasional TIK dan Ilmu Sosial (Sociotech)*, vol. 1, no. 1, hal. 1-6, 2017.
- [3] J. Jumadi, Yupianti, D. Sartika, "Pengolahan Citra Digital untuk Identifikasi Objek Menggunakan Metode Hierarchical Agglomerative Clustering", *Jurnal Sains dan Teknologi*, vol. 10, no. 2, hal. 148-156, 2021.
- [4] M. C. Kirana, Sartikha, E. Erminawati, "Penerapan Metode Canny dalam Koreksi Lembar Jawaban Komputer untuk Try Out," *Prosiding SENTIA 2017*, vol. 9, no. 1, hal. 9-14, 2017.
- [5] M. Furqan, Sriani & I. E. Y. Sari, "Penerapan Metode Otsu dalam Melakukan Segmentasi Citra pada Citra Naskah Arab," *Jurnal Universitas Bumigora*, vol. 20, no. 1, hal. 59-72, 2020.
- [6] M. Masril, A. Yuhandri, J. Na'am, "Analisis Perbandingan Perbaikan Kualitas Citra pada Motif Batik dengan Konsep Deteksi Tepi Robert, Sobel dan Canny menggunakan Metode Morfologi," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 3, no. 1, hal. 36-41, 2019.

- [7] M. Pinem, "Inkripsi Islam pada Makam-Makam Kuno Baru," *Jurnal Lektur Keagamaan*, vol. 16, no. 1, hal. 101-126, 2018.
- [8] M. Yulianti, C. Suhery, I. Ruslianto, " Pendeteksi Tempat Parkir Mobil Kosong Menggunakan Metode Canny," *Jurnal Coding*, vol. 5, no. 3, hal. 48-56, 2017.
- [9] M. Widyarningsih, "Identifikasi Kematangan Buah Apel dengan Gray Level Co-Occurrence Matrix (GLCM)," *Jurnal SAINTEKOM*, vol. 6, no. 1, hal. 71-88, 2017.
- [10] N. Fadillah, C. R. Gunawan, "Mendeteksi Keakuratan Metode Noise Salt and Pepper dengan Median Filter," *Jurnal Informatika*, vol. 6, no. 1, hal. 91-95, 2019.
- [11] Yuli K.S., "Analisis Perbandingan Algoritma Contraharmonic Mean Filter dan Arithmetic Mean Filter untuk Reduksi Exponential Noise pada Citra Digital", hal 5, 2019.
- [12] S. Ratna, "Pengolahan Citra Digital dan Histogram dengan Phyton dan Text Editor Phycharm", *Jurnal Ilmiah Technologia*, vol. 11, no. 3, hal. 181-186, 2020.
- [13] Sriani, A. Ikhwan, Yusnidah, "Aplikasi Metode Transformasi Wavelet Diskrit untuk Kompresi Citra pada Pengolahan Citra Digital", *Jurnal Ilmiah Saintikom*, vol. 16, no. 1, hal. 31-40, 2017.