

PERBANDINGAN *EDGE DETECTION* METODE LAPLACIAN OF GAUSSIAN DAN CANNY PADA CITRA CT SCAN MENGGUNAKAN SOFTWARE MATLAB

Rafi Fadhlurrahman, Sevada Afta Wijaksana, Farhan Ibnu Prabowo, Reni Rahmadewi

Teknik Elektro, Universitas Singaperbangsa Karawang

Jl. HS. Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat, Indonesia

2110631160056@student.unsika.ac.id

ABSTRAK

Edge detection merupakan sebuah proses identifikasi titik-titik perubahan intensitas yang signifikan dalam sebuah citra. Dalam bidang medis, deteksi tepi sangat penting untuk memudahkan proses segmentasi serta analisis dan diagnosis, salah satunya pada citra *Computed Tomography (CT) scan* yang seringkali menghadapi masalah *noise*. Oleh karena itu, diperlukan analisis perbandingan untuk menentukan metode deteksi tepi yang tepat guna meningkatkan efektivitas dalam identifikasi citra hasil *scan*. Penelitian ini bertujuan untuk membandingkan dua metode deteksi tepi, yaitu Laplacian of Gaussian (LoG) dan Canny, dengan citra *CT scan* paru-paru sebagai sampel. Tahapan penelitian dimulai dengan pemilihan dataset citra *CT scan* paru-paru, dilanjutkan dengan pra-pemrosesan citra yang mencakup *grayscale*, *resizing*, dan *smoothing* menggunakan filter Gaussian. Selanjutnya, dilakukan implementasi kedua metode deteksi tepi menggunakan software MATLAB. Evaluasi dilakukan berdasarkan nilai *Mean Squared Error (MSE)* dan *Peak Signal-to-Noise Ratio (PSNR)* serta melalui visualisasi hasil. Hasil penelitian menunjukkan bahwa metode Canny memberikan deteksi tepi yang lebih halus dengan *noise* yang lebih rendah dengan rata-rata pengukuran MSE sebesar 2129.08, dan rata-rata PSNR sebesar 15.30 dB. Sementara metode LoG menghasilkan lebih banyak detail tepi, tetapi dengan *noise* yang lebih tinggi dengan rata-rata pengukuran MSE sebesar 4517.60 dan rata-rata PSNR sebesar 11.28 dB.

Kata kunci : Deteksi Tepi, Laplacian of Gaussian, Canny, CT Scan, MATLAB

1. PENDAHULUAN

Pengolahan citra medis telah berkembang pesat seiring dengan kemajuan teknologi dan semakin kompleksnya kebutuhan dalam diagnosis serta perencanaan terapi. Di antara berbagai teknik pengolahan citra, deteksi tepi berguna dalam mengekstrak informasi kritis dari citra medis, terutama untuk mengidentifikasi batas antara struktur anatomi dan area secara patologis. Informasi tepi ini sangat penting untuk mendukung proses segmentasi, pengenalan pola, dan ekstraksi fitur yang nantinya dapat membantu dokter dalam membuat keputusan klinis yang tepat [1].

CT scan merupakan salah satu metode pencitraan yang banyak digunakan dalam dunia medis karena kemampuannya dalam memberikan gambaran detail struktur tubuh dengan resolusi tinggi. Meskipun demikian, citra *CT scan* tidak lepas dari tantangan, seperti munculnya *noise*, variasi intensitas, dan artefak yang diakibatkan oleh kondisi pengambilan gambar serta heterogenitas jaringan tubuh. Tantangan-tantangan ini mempersulit proses identifikasi tepi yang akurat dan konsisten. Oleh karena itu, penerapan algoritma deteksi tepi yang efektif menjadi krusial agar informasi struktural pada citra *CT scan* dapat diungkap dengan optimal [2].

Dalam literatur pengolahan citra, banyak metode deteksi tepi yang telah diusulkan. Dua di antaranya yang cukup populer dan sering diterapkan dalam analisis citra medis adalah metode Laplacian of Gaussian (LoG) dan Canny. Metode LoG menggabungkan proses *Gaussian smoothing* untuk mereduksi *noise* dengan penerapan operator Laplacian

yang sensitif terhadap perubahan intensitas. Pendekatan ini memungkinkan deteksi tepi secara detail, namun kinerjanya sangat bergantung pada pemilihan parameter, seperti nilai sigma, yang dapat mempengaruhi sensitivitas dan keakuratan deteksi [3]. Metode Canny mengusung pendekatan multi-tahap, yaitu reduksi *noise* melalui filter, perhitungan gradien intensitas, *non-maximum*, dan *hysteresis thresholding*. Proses ini menghasilkan tepi yang lebih halus dan kontinu serta lebih rendah *noise*, meskipun terkadang bisa mengorbankan detail struktural yang sangat halus [3] [4].

Kebutuhan untuk analisis yang lebih objektif ini mendorong dilakukannya penelitian yang membandingkan kedua metode tersebut pada dataset citra *CT scan* paru-paru. Penggunaan software MATLAB sebagai platform implementasi memungkinkan pemrosesan dan analisis yang terintegrasi dengan dukungan *Image Processing*, sehingga hasil yang diperoleh dapat diverifikasi secara sistematis. Dengan evaluasi berbasis *Mean Squared Error (MSE)* dan *Peak Signal-to-Noise Ratio (PSNR)*, penelitian ini tidak hanya fokus pada kualitas visual hasil deteksi tepi, tetapi juga memberikan pengukuran terkait efektivitas pengurangan *noise* dan keakuratan representasi tepi [4].

2. TINJAUAN PUSTAKA

2.1. Deteksi Tepi Dalam Citra Medis

Deteksi tepi merupakan proses identifikasi titik-titik perubahan intensitas yang signifikan dalam sebuah citra, yang biasanya merepresentasikan batas antar objek atau struktur dalam citra. Dalam konteks

citra medis seperti *CT scan*, deteksi memiliki beberapa peranan seperti:

- Segmentasi anatomi: Memisahkan struktur jaringan, dan pembuluh darah, guna mendukung analisis dan diagnosis.
- Pengurangan *noise*: Citra hasil *CT scan* sering mengandung *noise* atau artefak yang dapat mengganggu proses analisis, sehingga diperlukan algoritma yang tepat untuk membedakan antara tepi dan *noise*.
- Peningkatan fitur diagnostik: Tepi yang terdeteksi dengan baik dapat membantu dalam penentuan ukuran, bentuk, dan lokasi struktur yang tepat [1] [2].

Beberapa metode deteksi tepi lainnya, seperti operator Sobel, Prewitt, dan Roberts, telah diterapkan dalam berbagai aplikasi. Namun, untuk citra yang memiliki tantangan *noise* dan kontras rendah, metode seperti LoG dan Canny lebih banyak dipilih karena mengintegrasikan proses penghalusan (*smoothing*) dan pengolahan gradien secara optimal. Terdapat pada penelitian sebelumnya, menunjukkan implementasi deteksi tepi dalam objek bergerak, penelitian tersebut menyoroti penggunaan metode Canny dan LoG dalam mendeteksi tepi halus pada data non-medis [5]. Perbedaan konteks aplikasi tersebut membuka peluang untuk mengadaptasi dan mengoptimalkan metode ini pada citra *CT scan*, di mana penekanan pada detail struktural dan reduksi *noise* sangat penting.

2.2. Metode Laplacian of Gaussian (LoG)

Metode Laplacian of Gaussian (LoG) pertama kali diperkenalkan oleh Marr dan Hildreth [6]. Prinsip kerja dari metode ini melibatkan dua tahap utama:

- Gaussian Smoothing*: Tahap awal ini bertujuan untuk mereduksi *noise* dengan menerapkan filter Gaussian pada citra. Proses ini membantu menghaluskan perbedaan intensitas yang tidak signifikan yang disebabkan oleh *noise*. *smoothing* dilakukan dengan menerapkan persamaan berikut,

$$G(i, j) = -\frac{1}{2\pi\sigma^2} \times e^{-\frac{(i-u)^2 + (j-v)^2}{2\sigma^2}} \quad (3)$$

Dimana:

$e = 2,71$ (konstanta euler)

σ = simpangan baku (sigma)

$\pi = 3,14$ (pi)

- Penerapan Operator Laplacian*: Setelah citra mengalami *smoothing*, metode Laplacian digunakan untuk mendeteksi area dengan perubahan intensitas yang tajam. Titik-titik yang menunjukkan nol pergeseran pada respons Laplacian dianggap sebagai tepi, itulah yang disebut dengan *zero-crossing* [6].

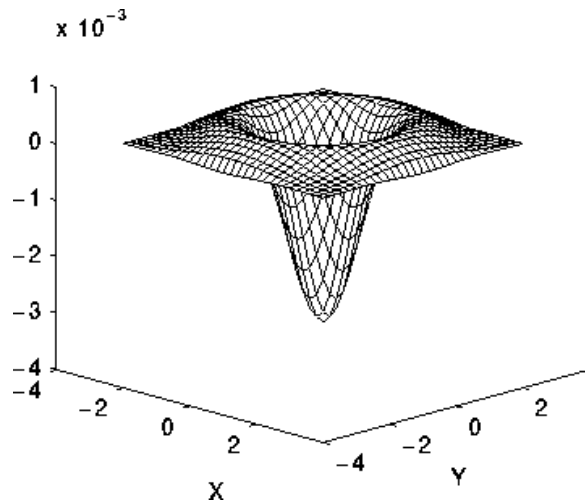
Fungsi dari Laplacian of Gaussian sebagai berikut,

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (1)$$

Contoh operator LoG dalam 1 buah kernel 3×3 gradientnya didefinisikan sebagai berikut,

$$\text{dan } G = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (2)$$

Fungsi operator Laplacian of Gaussian yang diperoleh melalui konvolusi disebut juga filter topi Meksiko (*the Mexican Hat Filter*) karena bentuk kurvanya mirip seperti topi yang dikenakan orang-orang Meksiko [7].



Gambar 1. Fungsi 2-D LoG.

2.3. Metode Canny

Metode Canny telah menjadi salah satu algoritma deteksi tepi yang paling banyak digunakan karena keunggulannya dalam hal deteksi, lokalisasi, dan minimalisasi *noise* [8].

Metode Canny dalam pendeteksian tepi bekerja dengan memproses setiap piksel citra secara berurutan, dimulai dari piksel di pojok kiri atas (timur utara) hingga ke piksel di pojok kanan bawah (barat selatan). Operator Canny dirancang untuk menghasilkan citra tepi yang optimal dengan tingkat kesalahan yang sangat rendah [8].

Algoritma Canny melibatkan empat tahap, yaitu *smoothing* melalui filter, penghitungan gradien, *non-maximum* untuk menentukan lokasi tepi, dan *hysteresis thresholding* untuk memastikan kontinuitas tepi [8].

Tahapan-tahapan ini diterapkan untuk menjalankan proses pendeteksian tepi yang kompleks. Berikut adalah langkah-langkahnya:

- Smoothing*: merupakan proses pengaburan gambar untuk mengurangi *noise*. Teknik pengaburan ini dilaksanakan dengan memanfaatkan filter Gaussian.
- Finding Gradient*: Tepi yang signifikan diharapkan memiliki nilai gradien yang tinggi, sehingga operator gradien diaplikasikan untuk melakukan pencarian dalam arah horizontal dan vertikal. Hasil

pencarian gradien didefinisikan sebagai berikut [9],

$$Gx = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ dan } Gy = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & 1 \end{bmatrix} \quad (4)$$

Hasil dari kedua operator digabungkan untuk memberikan hasil gabungan dari sisi vertikal dan horizontal menggunakan persamaan berikut,

$$G = \sqrt{Gx^2 + Gy^2} \quad (5)$$

Kemudian menentukan arah tepian yang ditemukan dengan menggunakan persamaan,

$$\theta = \arctan\left(\frac{Gy}{Gx}\right) \quad (6)$$

- c. *Non-maximum*: Proses penghapusan non-maksimum dilakukan sepanjang tepi citra untuk mengeliminasi piksel yang tidak menunjukkan nilai maksimum, sehingga hanya piksel dengan intensitas tertinggi yang dipertahankan sebagai tepi. Hal ini menghasilkan tepi yang lebih tipis [9].
- d. *Hysteresis thresholding*: Setelah *non-maximum suppression*, citra tepi biasanya masih mengandung piksel yang lemah atau *noise*. *Hysteresis thresholding* (atau *edge tracking by hysteresis*) digunakan untuk memisahkan tepi kuat dan lemah yakni *high threshold* dan *low threshold*. Langkah ini memastikan bahwa hanya tepi yang kontinu dan signifikan yang dihasilkan sebagai output akhir [9].

2.4. MSE (Mean Square Error)

Mean Square Error (MSE) merupakan metrik yang digunakan untuk mengukur perbedaan kuadrat rata-rata antara dua citra. Prinsip dasar perhitungan MSE adalah mengukur selisih intensitas piksel pada posisi yang sama antara kedua citra, kemudian mengkuadratkan selisih tersebut [10]. Hasil akhir adalah nilai rata-rata dari semua perbedaan kuadrat tersebut. Secara matematis, MSE dapat dituliskan sebagai,

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N [I(i,j) - K(i,j)]^2 \quad (7)$$

dimana:

- $I(i,j)$ adalah intensitas piksel pada posisi (i,j) pada citra asli.
- $K(i,j)$ adalah intensitas piksel pada posisi (i,j) pada citra hasil.
- M dan N merupakan dimensi citra (jumlah baris dan kolom).

2.5. PSNR (Peak Signal-to-Noise Ratio)

PSNR (*Peak Signal-to-Noise Ratio*) adalah metrik yang digunakan untuk mengukur kualitas citra hasil pemrosesan atau kompresi dengan membandingkannya dengan citra asli. Metrik ini mengindikasikan seberapa besar perbedaan antara citra asli dan citra yang telah diproses, dengan menggunakan skala logaritmik yang dinyatakan dalam satuan desibel (dB). Semakin tinggi nilai PSNR,

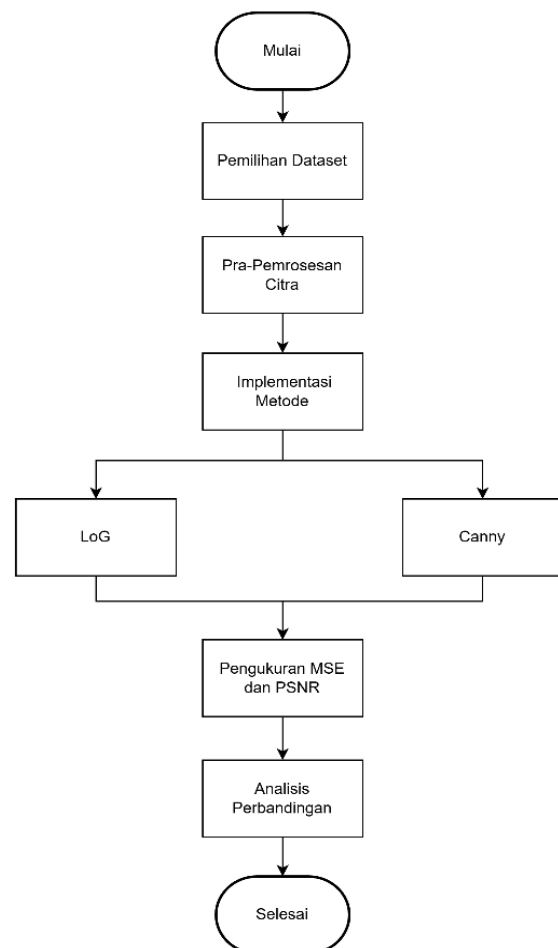
semakin baik kualitas citra hasil, karena menunjukkan tingkat kesalahan (*noise* atau distorsi) yang rendah relatif terhadap sinyal aslinya [10]. Secara matematis, didefinisikan sebagai berikut,

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right) \quad (7)$$

Dimana Max adalah nilai piksel terbesar pada keseluruhan citra, MSE adalah hasil dari *Mean Square Error*.

3. METODE PENELITIAN

Metode penelitian ini terdiri dari serangkaian langkah kerja yang terstruktur dengan baik. Untuk mencapai tujuan penelitian, langkah-langkah yang dilakukan meliputi beberapa tahapan. Diagram alir dari tahapan-tahapan penelitian ini dapat dilihat pada Gambar 2.



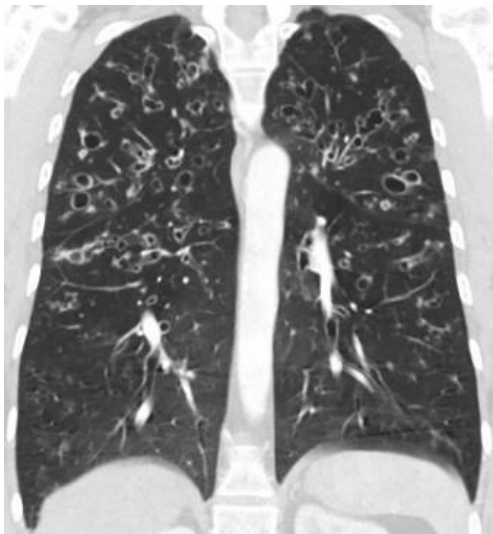
Gambar 2. Tahapan penelitian

3.1. Pemilihan Dataset

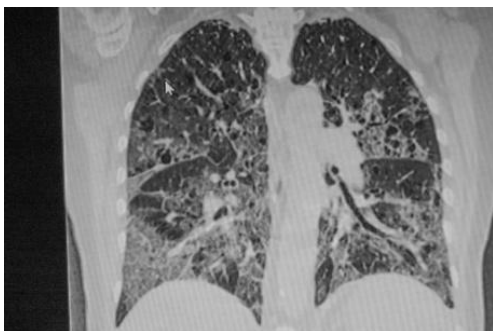
Penelitian ini menggunakan citra *CT scan* paru-paru yang diambil dari database Kaggle. Dataset tersebut dapat diakses melalui situs web dan disimpan dalam format gambar standar (JPG/PNG) sehingga mudah diolah dengan MATLAB. Untuk citra yang digunakan dapat dilihat pada gambar 3, 4, dan 5.



Gambar 3. Citra CT scan 1



Gambar 4. Citra CT scan 2



Gambar 5. Citra CT scan 3

3.2. Pra-Pemrosesan Citra

Pra-pemrosesan citra merupakan tahap awal yang krusial untuk menyiapkan data agar siap diproses oleh algoritma deteksi tepi. Tahapan ini mencakup beberapa proses, yaitu *grayscale*, *resizing*, dan *smoothing*. *Grayscale* mengubah citra berwarna menjadi citra hitam-putih sehingga menghilangkan informasi warna guna menyederhanakan perhitungan intensitas. *Resizing* memastikan bahwa semua citra memiliki dimensi yang seragam, sehingga perbandingan hasil deteksi tepi antar citra dapat dilakukan secara konsisten. *Smoothing*, umumnya menggunakan filter Gaussian, berperan penting dalam

mengurangi *noise* dan artefak yang sering muncul pada citra. Dengan mengurangi *noise*, *smoothing* membantu meningkatkan kualitas citra sehingga algoritma deteksi tepi sehingga metode yang akan diterapkan selanjutnya dapat bekerja lebih optimal dalam mengidentifikasi batas struktural secara akurat [7].

3.3. Implementasi Metode

Pada bagian implementasi metode, kedua algoritma deteksi tepi, yaitu LoG dan Canny, diimplementasikan menggunakan MATLAB. Untuk metode LoG, citra awal yang telah dihaluskan dengan filter Gaussian, kemudian operator Laplacian diterapkan untuk mendeteksi perubahan intensitas, dengan hasil deteksi tepi diperoleh melalui analisis *zero-crossing*. Sementara itu untuk metode Canny, setelah reduksi *noise* menggunakan filter Gaussian, dilanjutkan dengan perhitungan gradien intensitas menggunakan fungsi MATLAB, kemudian dilakukan proses *non-maximum suppression* untuk menipiskan tepi hasil, dan diakhiri dengan penerapan *hysteresis thresholding* guna menghasilkan tepi yang halus dan kontinu. Seluruh proses ini diintegrasikan dalam skrip MATLAB yang tidak hanya untuk visualisasi hasil deteksi tepi, tetapi juga perhitungan otomatis parameter evaluasi seperti MSE dan PSNR.

3.4. Pengukuran MSE dan PSNR

Mengukur efektivitas kedua metode deteksi tepi, yaitu LoG dan Canny, dengan menggunakan pendekatan kuantitatif dan kualitatif. Secara kuantitatif, perbandingan dilakukan dengan menghitung nilai MSE dan PSNR untuk setiap citra yang telah diproses; MSE mengukur perbedaan kuadrat rata-rata antara citra asli dan citra hasil, sedangkan PSNR mengindikasikan rasio antara sinyal maksimum dengan *noise* berdasarkan nilai MSE.

3.5. Analisis Perbandingan

Analisis dilakukan pada 3 sampel citra *CT scan* paru-paru, dengan perhitungan MSE dan PSNR dari kedua metode tersebut sebagai acuan untuk menilai kualitas deteksi tepi. Selain itu, analisis kualitatif melalui visualisasi hasil juga dilakukan, sehingga memungkinkan penilaian menyeluruh mengenai keunggulan masing-masing metode dalam mengurangi *noise* dan mempertahankan detail struktural.

4. HASIL DAN PEMBAHASAN

4.1. Grayscale

MATLAB menyediakan fungsi bawaan yaitu `'rgb2gray'` yang secara otomatis melakukan konversi menggunakan formula tersebut. Dengan cara ini, citra yang dihasilkan hanya berisi nilai intensitas yang berkisar antara 0 (hitam) hingga 255 (putih) untuk citra 8-bit.

Name ▲	Value
ct_scan1	575x675x3 uint8
ct_scan2	729x675 uint8
ct_scan3	800x1200x3 uint8

Gambar 6. Value citra sebelum *grayscale*ing

Pada gambar 6, terdapat informasi mengenai citra yang diinputkan, dapat dilihat pada *value* citra yang digunakan. Dimana 3 menginformasikan banyaknya komponen matriks yaitu *red*, *green* dan *blue* (RGB). *Uint8* menyatakan kelas data citra 8-bit. Ketika program dijalankan maka hasilnya dapat dilihat pada gambar 7.

Name ▲	Value
ct_scan1	575x675 uint8
ct_scan2	729x675 uint8
ct_scan3	800x1200 uint8

Gambar 7. Value citra hasil *grayscale*ing

4.2. Resizing

Fungsi yang digunakan untuk mengubah ukuran gambar di MATLAB adalah fungsi '*imresize*'. Fungsi ini memungkinkan untuk mengubah dimensi gambar ke ukuran yang diinginkan. Pada penelitian ini menggunakan ukuran 512x512 piksel. Hal ini penting untuk memastikan bahwa algoritma pengolahan citra, bekerja pada gambar dengan ukuran yang seragam, sehingga perbandingan antar gambar menjadi lebih mudah dan konsisten. Hasilnya dapat dilihat pada gambar 8.

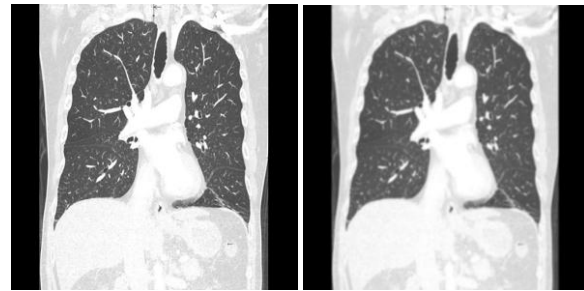
Name ▲	Value
ct_scan1	575x675 uint8
ct_scan1_resized	512x512 uint8
ct_scan2	729x675 uint8
ct_scan2_resized	512x512 uint8
ct_scan3	800x1200 uint8
ct_scan3_resized	512x512 uint8

Gambar 8. Value hasil *resizing* citra

4.3. Smoothing

Pengaplikasian *smoothing* bertujuan mengurangi noise yang ada dalam citra. Penggunaan parameter *smoothing* dengan nilai sigma (σ) yang disesuaikan untuk mencapai hasil optimal. LoG sensitif terhadap noise, dengan *smoothing* yang lebih kuat, noise pada citra dapat diminimalisasi, sehingga operator Laplacian dapat mendeteksi perubahan intensitas secara lebih akurat.

Metode Canny sudah memiliki rangkaian proses (*non-maximum suppression* dan *hysteresis thresholding*) yang dirancang untuk menjaga kehalusan dan kontinuitas tepi. melakukan *pre-smoothing* tambahan bisa jadi redundan dan berpotensi menghilangkan detail tepi yang penting. Hasil *smoothing* dapat dilihat pada Gambar 9.

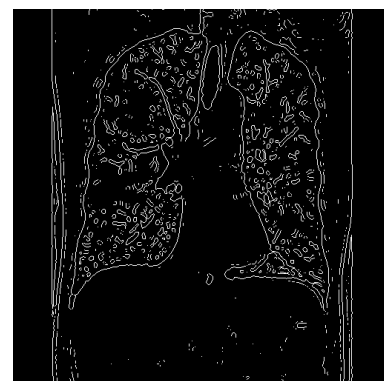


(a) (b)
Gambar 9. (a). Citra sebelum *smoothing*,
(b). Citra setelah *smoothing* ($\sigma = 1.5$)

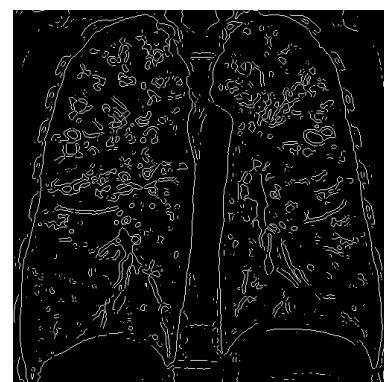
4.4. Laplacian of Gaussian (LoG)

Setelah citra melalui proses *smoothing* dengan filter Gaussian, operator Laplacian diterapkan untuk menghitung perubahan intensitas. Operator Laplacian menekankan perubahan intensitas yang tajam pada citra. Dengan menggunakan fungsi '*log*' pada MATLAB, proses ini juga menyertakan analisis *zero-crossing* secara otomatis.

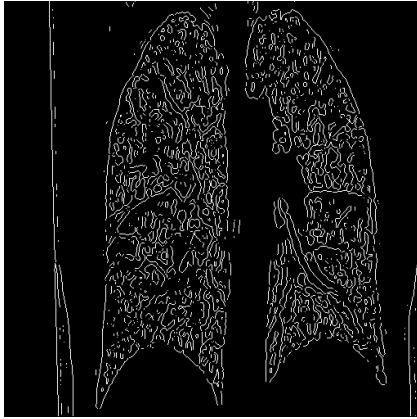
Analisis *zero-crossing* dilakukan dengan cara mencari titik-titik pada citra di mana terjadi peralihan nilai dari positif ke negatif atau sebaliknya. Dalam implementasinya secara langsung, setiap piksel diperiksa bersama dengan piksel-piksel disebelahnya. Jika terdapat perbedaan tanda antara suatu piksel dengan setidaknya satu disebelahnya, maka titik tersebut dianggap sebagai tepi. Adapun hasil pengujian program deteksi tepi terhadap citra dapat dilihat pada gambar 10, 11, dan 12.



Gambar 10. Hasil citra CT scan 1 metode LoG



Gambar 11. Hasil citra CT scan 2 metode LoG



Gambar 12. Hasil citra CT scan 3 metode LoG

Dapat dilihat pada gambar diatas, penerapan metode ini menghasilkan deteksi tepi yang cukup sensitif terhadap perubahan intensitas. *Zero-crossing* berhasil mendeteksi batas-batas utama pada struktur paru-paru, namun terlihat pula adanya *noise* di area yang memiliki transisi intensitas yang tidak terlalu signifikan. Akibatnya, garis tepi yang terdeteksi cenderung terputus-putus.

4.5. Canny

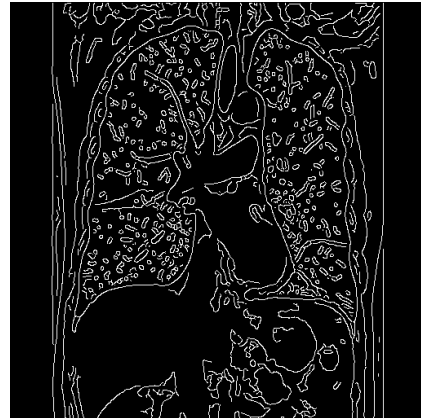
Berbeda dengan operator Laplacian, Fungsi 'Canny' di MATLAB sudah menyertakan *smoothing* internal dengan filter Gaussian, sehingga dalam kebanyakan kasus tidak perlu dilakukan *pre-smoothing* tambahan.

Setelah *smoothing*, citra dianalisis untuk menghitung gradien intensitas. Proses ini melibatkan perhitungan perubahan nilai intensitas secara horizontal dan vertikal. MATLAB kemudian menggabungkan gradien tersebut untuk menghasilkan besaran gradien dan arah tepi secara otomatis.

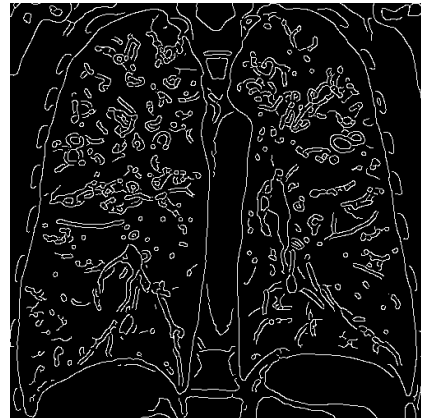
Selanjutnya pada tahap *non-maximum* dilakukan eliminasi piksel yang bukan merupakan nilai maksimum lokal sepanjang arah gradien. Dengan demikian, hanya piksel-piksel yang benar-benar mewakili tepi tajam yang dipertahankan, sehingga menghasilkan tepi yang lebih tipis. Selanjutnya, diterapkan *thresholding* untuk membedakan antara tepi kuat dan tepi lemah. Piksel dengan nilai gradien di atas *threshold* tinggi langsung dikategorikan sebagai tepi, sedangkan untuk piksel rendah melalui proses pelacakan (*edge tracking*) dengan *hysteresis*, piksel rendah dihubungkan dengan piksel tepi tinggi jika keduanya saling terhubung, sehingga menghasilkan deteksi tepi yang kontinu. Adapun hasil pengujian deteksi tepi metode Canny terhadap citra dapat dilihat pada gambar 13, 14, dan 15.

Dapat dilihat pada gambar berikut, hasil dari implementasi metode ini adalah citra yang menunjukkan posisi tepi secara jelas. Dengan kombinasi proses *smoothing*, perhitungan gradien, *non-maximum*, dan *hysteresis thresholding*, metode Canny mampu menghasilkan deteksi tepi yang halus

dan stabil, bahkan pada citra dengan *noise* yang cukup banyak.



Gambar 13. Hasil citra CT scan 1 metode Canny



Gambar 14. Hasil citra CT scan 2 metode Canny



Gambar 15. Hasil citra CT scan 3 metode Canny

4.6. Pengukuran MSE dan PSNR

pengukuran MSE dan PSNR dilakukan untuk setiap citra dari kedua metode. Pada MATLAB, perhitungan MSE dan PSNR dilakukan secara otomatis melalui fungsi bawaan. MATLAB menyediakan fungsi '*immse*' yang secara langsung menghitung nilai MSE. Setelah MSE didapatkan, nilai PSNR dapat dihitung menggunakan fungsi '*psnr*'. Berikut hasilnya dapat dilihat pada tabel 1.

Tabel 1. Hasil MSE dan PSNR pada citra

Citra	MSE (LoG)	PSNR (LoG) (dB)	MSE (Canny)	PSNR (Canny) (dB)
CT scan 1	3921.34	12.18	1188.87	17.37
CT scan 2	5021.09	10.10	3010.93	12.68
CT scan 3	4610.37	11.56	2187.44	15.86
Rata-rata	4517.60	11.28	2129.08	15.30

Dari Tabel 1, dapat dilihat bahwa untuk ketiga gambar, secara keseluruhan metode Canny menghasilkan nilai MSE yang lebih rendah dan nilai PSNR yang lebih tinggi dibandingkan metode LoG.

Pada citra pertama, Metode Canny memiliki nilai PSNR sebesar 17.37 dB, lebih tinggi dari LoG, yang menunjukkan bahwa deteksi tepi lebih halus dan memiliki tingkat *noise* yang lebih rendah. Pada citra kedua, baik metode LoG maupun Canny menunjukkan nilai MSE yang lebih tinggi dibandingkan CT scan 1, yang berarti bahwa deteksi tepi pada citra ini memiliki *noise* lebih banyak. Meskipun metode Canny mengalami peningkatan MSE, hasilnya tetap lebih baik dibandingkan LoG, dengan selisih PSNR sebesar 2.58 dB. Citra ketiga menunjukkan hasil yang konsisten dengan dua citra sebelumnya, di mana metode LoG memiliki MSE lebih tinggi dan PSNR lebih rendah dibandingkan metode Canny. Nilai PSNR 15.86 dB pada metode Canny menunjukkan bahwa citra hasil deteksi memiliki kualitas lebih baik dibandingkan metode LoG.

4.7. Analisis Perbandingan

Berdasarkan hasil evaluasi tersebut, terlihat bahwa metode Canny secara konsisten menunjukkan kinerja yang lebih baik dibandingkan dengan metode LoG. Metode LoG menghasilkan tepi yang lebih kasar dan terfragmentasi, dengan banyak area kecil yang tampak seperti *noise*. Terlihat bahwa metode ini sangat responsif terhadap perubahan intensitas, yang menyebabkan munculnya banyak detail kecil, termasuk deteksi tepi palsu akibat gangguan *noise*. Struktur utama paru-paru terlihat, tetapi sulit membedakan mana yang merupakan batasan nyata dan mana yang hanya efek dari sensitivitas tinggi terhadap intensitas. Terutama pada CT scan 2, deteksi tepi menjadi tidak stabil, dengan banyak area yang tampak terlalu terang atau berisi garis tepi berlebihan yang tidak merepresentasikan batas struktur sebenarnya.

Pada metode Canny, tepi yang dihasilkan lebih halus dan lebih jelas, dengan batasan objek yang lebih kontinu dibandingkan metode LoG. *Noise* yang muncul lebih sedikit, sehingga batas paru-paru terlihat lebih jelas dan terdefinisi dengan baik. Pada semua citra CT scan, metode ini menunjukkan hasil yang lebih stabil dan tidak mudah menangkap *noise* yang tidak perlu. Terutama pada CT scan 1, garis tepi paru-paru tampak lebih terhubung dan tidak terputus, yang

memudahkan dalam proses segmentasi lanjutan untuk analisis medis.

Metode LoG memiliki rata-rata MSE yang jauh lebih tinggi sebesar 4517.60 dibandingkan metode Canny sebesar 2129.08, yang menunjukkan bahwa metode LoG menghasilkan lebih banyak *noise* dalam deteksi tepi.

Rata-rata PSNR metode LoG hanya 11.28 dB, menunjukkan kualitas deteksi yang lebih rendah karena banyaknya *noise*. Sebaliknya, metode Canny memiliki rata-rata PSNR sebesar 15.30 dB, lebih tinggi dibandingkan LoG, sehingga lebih unggul dalam menghasilkan deteksi tepi yang lebih jelas dan minim *noise*.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian ini, metode Canny terbukti lebih unggul dibandingkan metode Laplacian of Gaussian (LoG) dalam mendeteksi tepi citra CT scan paru-paru, baik secara kuantitatif menggunakan MSE dan PSNR maupun secara visual melalui analisis pengamatan hasil deteksi tepi. Metode Canny memiliki rata-rata MSE sebesar 2129.08 dan PSNR sebesar 15.30 dB, sedangkan metode LoG memiliki rata-rata MSE lebih tinggi sebesar 4517.60 dan PSNR lebih rendah sebesar 11.28 dB, menunjukkan bahwa metode LoG lebih rentan terhadap *noise*. Dari hasil pengamatan visual, metode LoG menghasilkan deteksi tepi yang lebih kasar, dengan *noise* yang cukup banyak dan tepi terfragmentasi, sementara metode Canny menghasilkan tepi yang lebih halus, kontinu, dan lebih jelas dalam membatasi struktur paru-paru.

Oleh karena itu, metode Canny lebih direkomendasikan untuk analisis citra medis, terutama untuk CT scan paru-paru, karena memberikan hasil deteksi yang lebih akurat dan minim *noise*. Mengingat keterbatasan metode yang digunakan dalam menangani *noise* dan deteksi tepi yang terfragmentasi, penelitian selanjutnya dapat menambah pendekatan berbasis deep learning. Metode ini berpotensi memberikan deteksi yang lebih adaptif terhadap variasi intensitas serta artefak pada citra medis. Selain itu, dapat dilakukan kombinasi metode LoG dan Canny untuk mendapatkan hasil yang lebih optimal, seperti menggunakan LoG untuk menangkap detail kecil dan Canny untuk meningkatkan kehalusan tepi.

DAFTAR PUSTAKA

- [1] M. K. Pradani Ayu Widya Purnama and M. S. Dr. Ir. Sumijan, *Teori dan Aplikasi Pengolahan Citra Digital Penerapan dalam Bidang Citra Medis*. 2021.
- [2] S. R. Kondamuri *et al.*, "Chest CT Image based Lung Disease Classification – A Review," *Curr. Med. Imaging Rev.*, vol. 20, no. 1, pp. 1–14, 2023, doi: 10.2174/0115734056248176230923143105.
- [3] D. M. Silalahi, D. P. Sari, R. Prahasti, and D. D. Putra, "Analisis Performa Algoritma Pendeteksian Tepi pada Citra Multispektral," vol.

- 2, no. 3, 2024.
- [4] I. K. Seneng, I. M. B. Adnyana, I. M. A. W. Putra, and N. L. G. P. Suwirmayanti, "Studi Pembanding Edge Detection Metode Sobel dan Prewitt pada Citra Rontgen Menggunakan Software Matlab," *J. Eksplora Inform.*, vol. 13, no. 2, pp. 175–187, 2024, doi: 10.30864/eksplora.v13i2.1033.
- [5] M. S. Moelya, P. S. Ramadhan, and M. G. Suryanata, "Perbandingan Metode Canny, Sobel, Dan Laplacian of Gaussian Dalam Mendeteksi Tepi Citra Objek Bergerak," vol. 3, pp. 450–460, 2024.
- [6] O. Andika and S. Teknik, "Penerapan Matriks dalam Implementasi Gaussian Filter," pp. 1–6, 2025.
- [7] J. Nababan, "Pendeteksian Tepi Citra untuk Menentukan Kualitas Beras Berdasarkan Jenis Menggunakan Metode Laplacian of Gaussian," *Ilm. Inform. Komput.*, vol. 1, no. 1, pp. 13–19, 2020.
- [8] Prima Selvia Megawati, M. Choirel Anwar, Fatimah, Sudiyono, Darmini, and Saifudin, "Effectiveness Edge Detection Operator Canny To Improve Image Quality Thorax Ct Scan in Cases Covid-19," *J. Vocat. Heal. Stud.*, vol. 7, no. 2, pp. 77–82, 2023, doi: 10.20473/jvhs.v7.i2.2023.77-82.
- [9] M. A. Taqy, C. A. Yahya, M. Al Hafizh, and S. Nurakmalia, "Implementasi dan Analisis Metode Deteksi Tepi Canny Menggunakan OpenCV," vol. 1, no. 4, pp. 303–306, 2024.
- [10] L. Maximillian, Y. F. Riti, M. A. Agung, and Y. J. Palis, "Perbandingan Algoritma Sobel dan Canny untuk Deteksi Tepi Citra Daun Lidah Buaya," *Komputa J. Ilm. Komput. dan Inform.*, vol. 12, no. 2, pp. 69–79, 2023, doi: 10.34010/komputa.v12i2.10997.
- ..