**Deteksi Tepi pada Citra Digital Menggunakan Metode Sobel dan Canny Berbasis Streamlit**

**William Hans Natanael1, Geri Elway2, Ali Sufajar3**

Sains Data

Universitas Nusa Mandiri

15220006@nusamandiri.ac.id

Sains Data

Universitas Nusa Mandiri

15220004@nusamandiri.ac.id

Sains Data

Universitas Nusa Mandiri

15220011@nusamandiri.ac.id

**Abstrak**

Menemukan garis tepi yang memisahkan dua area gambar merupakan tujuan dari prosedur deteksi tepi. Teknik Sobel dan Canny untuk deteksi tepi citra digital akan dibahas dalam penelitian ini. Deteksi tepi dilakukan dengan mengidentifikasi 2(dua) citra digital, yaitu kentang.jpg, dan kolam renang.jpg. Banyaknya piksel putih yang ditampilkan dari proses pendeteksian tepi menunjukkan hasil pendeteksian tepi. Temuan ini menunjukkan bahwa pendekatan digital Canny mengungguli metode Sobel dalam identifikasi tepi gambar. Selain itu, aplikasi deteksi tepi ini juga telah diimplementasikan dalam antarmuka pengguna menggunakan Streamlit untuk visualisasi interaktif.

Kata Kunci: Deteksi Tepi, Citra Digital, Canny, Sobel,Streamlit

***Abstrak***

*Finding the edge line that separates two image areas is the goal of the edge detection procedure. The Sobel and Canny technique for digital image edge detection will be discussed in this research. Edge detection is carried out by identifying 2 (two) digital images, namely kentang.jpg, and kolam renang.jpg. The number of white pixels displayed from the edge detection process indicates the edge detection results. These findings show that Canny's digital approach outperforms Sobel's method in image edge identification. In addition, this edge detection application has also been implemented in a user interface using Streamlit for interactive visualization.*

Keywords: Edge Detection, Digital Image, Canny, Sobel, Streamlit

**PENDAhULUAN**

Pengenalan pola, penginderaan jarak jauh menggunakan satelit atau pesawat terbang, dan machine vision adalah beberapa aplikasi praktis yang sangat bergantung pada pengolahan citra. Pengolahan Citra melibatkan manipulasi gambar untuk meningkatkan kualitasnya dan membuatnya lebih mudah dipahami oleh komputer atau manusia.

Segmentasi citra merupakan salah satu teknik pengolahan citra yang sering digunakan. Tujuan dari segmentasi citra adalah untuk mengekstrak objek. disertakan di dalam gambar atau mengelompokkan gambar ke dalam area dengan masing-masing objek atau area memiliki karakteristik yang sama. Langkah awal dalam melakukan klasifikasi objek biasanya adalah segmentasi. Salah satu metode segmentasi gambar dalam pengolahan citra saat ini adalah deteksi tepi.

Penulisan ini akan menemukan deteksi tepi citra digital dengan menggunakan metode Sobel dan Canny.

**STUDI Literatur**

*A. Citra*

Citra merupakan gambaran, tiruan, atau representasi dari sebuah objek. Sebagai hasil dari sistem perekaman data, citra dapat muncul dalam berbagai bentuk, seperti foto (optik), sinyal video (analog) seperti tampilan di layar televisi, maupun dalam format digital yang bisa langsung disimpan di media penyimpanan.

*B. Pengolahan Citra*

Pengolahan citra merupakan proses memanipulasi gambar agar menghasilkan kualitas yang lebih baik sehingga dapat lebih mudah dipahami oleh manusia maupun komputer. Salah satu teknik umum yang digunakan dalam pengolahan citra adalah segmentasi citra. Segmentasi ini bertujuan untuk mengidentifikasi objek-objek dalam gambar atau membaginya ke dalam wilayah-wilayah yang memiliki kesamaan atribut. Proses segmentasi biasanya dilakukan pada tahap awal sebagai dasar dalam melakukan klasifikasi objek. Salah satu metode dalam segmentasi citra adalah pendeteksian tepi, yang digunakan untuk menandai batas antar objek dalam gambar.

*C. Deteksi Tepi*

Deteksi tepi (edge detection) adalah proses dalam pengolahan citra yang bertujuan untuk menemukan garis batas dari objek yang ada dalam gambar. Suatu titik pada koordinat (x, y) dianggap sebagai tepi jika memiliki perbedaan intensitas yang signifikan dengan titik-titik di sekitarnya. Teknik ini dimanfaatkan untuk mengenali batas objek dengan cara mendeteksi perubahan intensitas yang tajam antara dua area. Jika gambar memiliki kualitas yang tajam dan jelas, maka penentuan tepinya akan lebih mudah. Sebaliknya, jika gambar mengalami gangguan seperti noise, maka proses penentuan tepi menjadi lebih kompleks.

*D. Metode Canny*

Metode Canny merupakan pendekatan yang dirancang dengan menerapkan sejumlah tahapan dalam proses deteksi tepi menggunakan algoritma Canny, yang menghasilkan tampilan citra berbeda dengan efek visual menyerupai relief. Efek ini menciptakan kesan seperti permukaan batu yang dipahat, di mana garis-garis kasar membentuk representasi objek. Relief tersebut muncul akibat perbedaan intensitas cahaya dan bayangan, yang terbentuk karena arah datangnya cahaya terhadap gambar. Keunggulan metode Canny terletak pada kemampuannya dalam mereduksi noise sebelum proses deteksi tepi dilakukan, sehingga menghasilkan lebih banyak garis tepi yang akurat. Operator Canny juga dirancang untuk mendeteksi tepian sejati dengan tingkat kesalahan yang sangat rendah, menjadikannya metode yang optimal dalam menghasilkan citra tepi.

*D. Metode Sobel*

Metode Sobel merupakan pengembangan dari metode Robert dengan menerapkan filter High Pass Filter (HPF) yang dilengkapi dengan satu elemen nol sebagai penyangga. Metode ini didasarkan pada prinsip kerja fungsi Laplacian dan Gaussian, yang dikenal efektif untuk menghasilkan HPF. Salah satu keunggulan utama dari metode Sobel adalah kemampuannya dalam mereduksi noise sebelum melakukan proses deteksi tepi. Operator Sobel umumnya memberikan penekanan atau bobot lebih besar pada piksel-piksel yang berada dekat dengan pusat jendela, sehingga piksel-piksel tetangga memiliki pengaruh yang lebih kecil dalam perhitungan.

**METODE PENELITIAN**

Penelitian ini menggunakan metode studi pustaka, yakni dengan menelaah berbagai konsep dan teori terkait pengolahan citra digital yang relevan untuk proses deteksi tepi, seperti metode Prewitt, Sobel, dan Canny. Dilakukan implementasi program deteksi tepi citra digital menggunakan perangkat lunak **Python** dan **Streamlit**. Library yang digunakan antara lain: scikit-image untuk proses deteksi tepi dan Streamlit untuk membangun antarmuka interaktif berbasis web.

**HASIL DAN PEMBAHASAN**

Metode yang digunakan dalam penelitian ini adalah studi literatur, yaitu dengan mengkaji berbagai teori dan konsep mengenai pengolahan citra digital yang berkaitan dengan teknik deteksi tepi, seperti metode Sobel, dan Canny. Setelah itu, dibuat sebuah program pendeteksi tepi citra digital menggunakan perangkat lunak Matlab.

*A. Deteksi Tepi Gambar kentang.jpg*

def deteksi\_tepi(gambar\_path):

# Baca dan konversi ke grayscale

gambar = io.imread(gambar\_path)

gray = color.rgb2gray(gambar)

# Deteksi tepi dengan metode Canny

canny = feature.canny(gray)

# Deteksi tepi dengan metode Sobel

sobel = filters.sobel(gray)

# Konversi ke citra biner

sobel\_bin = sobel > 0.05

# Hitung jumlah piksel putih

canny\_count = np.sum(canny)

sobel\_count = np.sum(sobel\_bin)

# Tampilkan hasil

fig, ax = plt.subplots(1, 4, figsize=(18, 6))

ax[0].imshow(gambar)

ax[0].set\_title('Original')

ax[1].imshow(canny, cmap='gray')

ax[1].set\_title(f'Canny: {int(canny\_count)}')

ax[2].imshow(sobel\_bin, cmap='gray')

ax[2].set\_title(f'Sobel: {int(sobel\_count)}')

for a in ax:

a.axis('off')

plt.tight\_layout()

plt.show()

return {

'Sobel': int(sobel\_count),

'Canny': int(canny\_count)

}

A collage of images of a person's face

AI-generated content may be incorrect.

Gambar 1. Deteksi Tepi Gambar kentang.jpg

*B. Deteksi Tepi Gambar kolam renang.jpg*

def deteksi\_tepi(gambar\_path):

# Baca dan konversi ke grayscale

gambar = io.imread(gambar\_path)

gray = color.rgb2gray(gambar)

# Deteksi tepi dengan metode Canny

canny = feature.canny(gray)

# Deteksi tepi dengan metode Sobel

sobel = filters.sobel(gray)

# Konversi ke citra biner

sobel\_bin = sobel > 0.05

# Hitung jumlah piksel putih

canny\_count = np.sum(canny)

sobel\_count = np.sum(sobel\_bin)

# Tampilkan hasil

fig, ax = plt.subplots(1, 4, figsize=(18, 6))

ax[0].imshow(gambar)

ax[0].set\_title('Original')

ax[1].imshow(canny, cmap='gray')

ax[1].set\_title(f'Canny: {int(canny\_count)}')

ax[2].imshow(sobel\_bin, cmap='gray')

ax[2].set\_title(f'Sobel: {int(sobel\_count)}')

for a in ax:

a.axis('off')

plt.tight\_layout()

plt.show()

return {

'Sobel': int(sobel\_count),

'Canny': int(canny\_count)

}

A collage of images of a person's feet

AI-generated content may be incorrect.

Gambar 2. Deteksi Tepi Gambar kolam renang.jpg

Tabel 1.

Daftar Jumlah Piksel Warna Putih Pada Citra

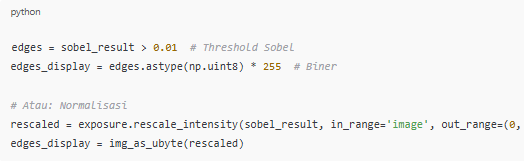
|  |  |  |  |
| --- | --- | --- | --- |
| Nama Citra | Jumlah Piksel Warna Putih | | Kesimpulan |
| Sobel | Canny |
| Kentang.jpg | 4435 | 2757 | Sobel lebih besar dari Canny |
| Kolam renang.jpg | 48327 | 14877 | Sobel lebih besar dari Canny |

Jumlah piksel berwarna putih merupakan hasil dari proses deteksi tepi pada citra. Semakin tinggi jumlah piksel putih, maka semakin banyak tepi yang berhasil terdeteksi. Oleh karena itu, setiap metode deteksi tepi akan menghasilkan jumlah piksel putih yang berbeda untuk setiap citra yang diuji. Penghitungan jumlah piksel putih dilakukan menggunakan fungsi nnz (number of nonzero entries). Tabel 1 menunjukkan jumlah piksel putih yang dihasilkan oleh masing-masing metode deteksi tepi untuk seluruh citra.

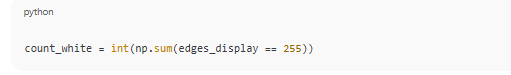
Berdasarkan data pada Tabel 1, dapat disimpulkan bahwa metode Sobel menghasilkan jumlah piksel putih yang lebih banyak dibandingkan dengan metode Canny.

C**. Implementasi Aplikasi Streamlit**

Aplikasi berbasis web dibuat menggunakan framework **Streamlit** yang memungkinkan pengguna mengunggah gambar dan memilih metode deteksi tepi secara langsung. Gambar yang diunggah kemudian diproses dengan metode **Sobel** dan hasilnya ditampilkan secara visual, baik dalam bentuk biner (0/255) maupun dalam skala abu-abu (grayscale).



Hasil deteksi ditampilkan di Streamlit dengan caption hasil dan jumlah piksel putih yang dihitung menggunakan:

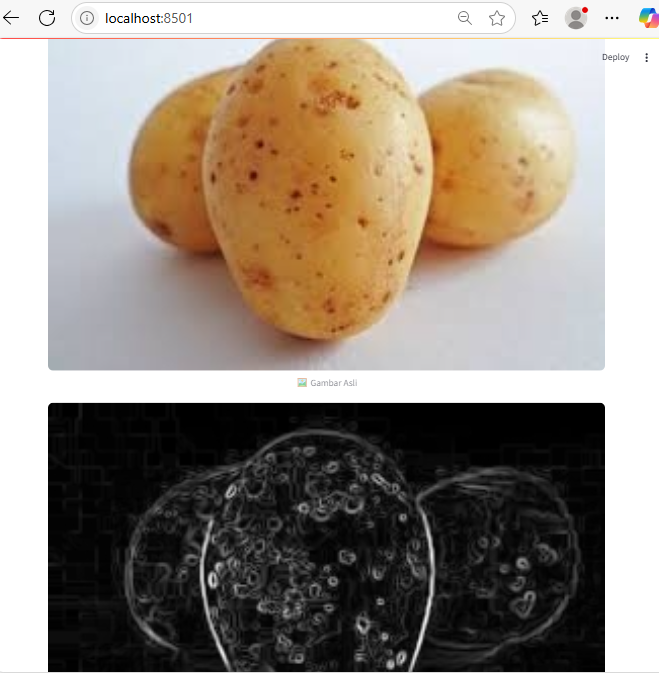


Tampilan antarmuka pengguna:

* Menu pilihan metode deteksi tepi
* Menu pilihan tampilan hasil (biner atau grayscale)
* Hasil gambar asli dan hasil deteksi tepi
* Jumlah piksel putih yang ditampilkan langsung di bawah gambar hasil deteksi



Gambar 3. Tampilan antarmuka Streamlit pada saat pemilihan gambar dan metode deteksi.



Gambar 4. Hasil deteksi tepi ditampilkan dalam bentuk grayscale beserta jumlah piksel putih.

**KESIMPULAN**

Berdasarkan hasil deteksi tepi terhadap 2 (dua) buah citra digital dapat disimpulkan bahwa deteksi tepi dengan menggunakan metode Sobel lebih baik dari metode Canny. Selain itu, implementasi dalam platform Streamlit memungkinkan pengguna menguji citra mereka sendiri secara interaktif, memperluas aksesibilitas dan kemudahan penggunaan dari metode yang dikembangkan.

**DAFTAR PUSTAKA**

Gonzalez, Rafael C, and Richard E Woods. 2018. *Digital Image Processing*. 4th ed. Pearson Education.

Herlambang, Andriana, and Alam Rahmatulloh. 2025. “Peningkatan Akurasi Deteksi Tepi Pada Citra Inversi Menggunakan Sobel‑Canny‑Prewitt.” *Jurnal Informatika dan Multimedia* 17(1). https://jurnal.polinema.ac.id/index.php/jtim/article/view/6590.

Inc., Streamlit. 2023. “Streamlit Documentation.” https://docs.streamlit.io.

McKinney, Wes. 2022. *Python for Data Analysis*. 3rd ed. O’Reilly Media.

Qisthiano, M Riski, and Ayu Okta Pratiwi. 2025. “MENGGUNAKAN ALGORITMA SOBEL DAN PREWITT.” 13(2): 1115–22.

Rosebrock, Adrian. 2017. *Practical Python and OpenCV: An Introductory, Example-Driven Guide to Image Processing*. 3rd ed. PyImageSearch.

Sukatmi, Sukatmi. 2017. “Perbandingan Deteksi Tepi Citra Digital Dengan Menggunakan Metode Prewitt, Sobel Dan Canny.” *KOPERTIP : Jurnal Ilmiah Manajemen Informatika dan Komputer* 1(1): 1–4. doi:10.32485/kopertip.v1i1.3.

Supiyandi, Supiyandi. 2024. “Analisis Performa Algoritma Pendeteksian Tepi Pada Citra Multispektral.” *Jurnal Sistem Informasi dan Ilmu Komputer* 8(2). https://ifrelresearch.org/index.php/jusiik-widyakarya/article/view/3472.

(Sukatmi 2017)(Qisthiano and Pratiwi 2025)

Gonzalez, Rafael C, and Richard E Woods. 2018. *Digital Image Processing*. 4th ed. Pearson Education.

Herlambang, Andriana, and Alam Rahmatulloh. 2025. “Peningkatan Akurasi Deteksi Tepi Pada Citra Inversi Menggunakan Sobel‑Canny‑Prewitt.” *Jurnal Informatika dan Multimedia* 17(1). https://jurnal.polinema.ac.id/index.php/jtim/article/view/6590.

Inc., Streamlit. 2023. “Streamlit Documentation.” https://docs.streamlit.io.

McKinney, Wes. 2022. *Python for Data Analysis*. 3rd ed. O’Reilly Media.

Qisthiano, M Riski, and Ayu Okta Pratiwi. 2025. “MENGGUNAKAN ALGORITMA SOBEL DAN PREWITT.” 13(2): 1115–22.

Rosebrock, Adrian. 2017. *Practical Python and OpenCV: An Introductory, Example-Driven Guide to Image Processing*. 3rd ed. PyImageSearch.

Sukatmi, Sukatmi. 2017. “Perbandingan Deteksi Tepi Citra Digital Dengan Menggunakan Metode Prewitt, Sobel Dan Canny.” *KOPERTIP : Jurnal Ilmiah Manajemen Informatika dan Komputer* 1(1): 1–4. doi:10.32485/kopertip.v1i1.3.

Supiyandi, Supiyandi. 2024. “Analisis Performa Algoritma Pendeteksian Tepi Pada Citra Multispektral.” *Jurnal Sistem Informasi dan Ilmu Komputer* 8(2). https://ifrelresearch.org/index.php/jusiik-widyakarya/article/view/3472.

(Gonzalez and Woods 2018)

(Herlambang and Rahmatulloh 2025)

(Rosebrock 2017)

(Inc. 2023)

(McKinney 2022)

(Supiyandi 2024)