

Implementasi dan Analisis Metode Deteksi Tepi Canny Menggunakan OpenCV

Muyassar Ahmad Taqy^{1*}, Churul Ain Yahya², Muhammad Al Hafizh³, Salwa Nurakmalia⁴
Perani Rosyani⁵

¹⁻⁵Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46,
Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ^{1*}muyassar7997@email.com, ²churulainy.21@gmail.com, ³apisapise26@gmail.com,
⁴akmaliasalwa87@gmail.com, ⁵dosen00837@gmail.com

(* : coressponding author)

Abstrak—Metode deteksi tepi Canny adalah teknik pengolahan citra digital yang populer untuk mengidentifikasi tepi objek dalam gambar. Artikel ini membahas penggunaan metode ini menggunakan OpenCV dan mengevaluasi kinerjanya pada gambar digital. Ini mencakup pemrosesan gambar sebelumnya, penerapan algoritma Canny, dan analisis hasil deteksi tepi. Hasil menunjukkan bahwa teknik ini memiliki kemampuan untuk mendeteksi tepi dengan akurat dan efektif.

Kata Kunci: OpenCV, Deteksi Tepi Canny, Pengolahan Citra, Analisis Gambar, Deteksi Tepi

Abstract—The Canny Edge Detection method is a popular digital image processing technique used to identify object tepi in a picture. This article describes how to apply this method using OpenCV and evaluate its kinerjanya on digital images. This includes the previous picture analysis, the Canny algorithm, and the tepi detection result analysis. Results show that this technique has the ability to accurately and efficiently detect fingerprints.

Keywords: OpenCV, Canny Edge Detection, Image Processing, Image Analysis, Edge Detection

1. PENDAHULUAN

Pengolahan citra melibatkan analisis dan perubahan gambar digital untuk meningkatkan kualitas gambar atau mengekstrak informasi penting dari gambar. Berbagai teknik pengolahan gambar melakukan proses ini, seperti peningkatan gambar, restorasi gambar, segmentasi, dan deteksi tepi. Pengolahan gambar digunakan dalam berbagai industri, termasuk medis, pengawasan keamanan, pengenalan pola, dan otomotif. Salah satu komponen penting pengolahan citra adalah deteksi tepi, yang membantu memahami struktur dan bentuk objek dalam gambar. (“Pengolahan Citra Digital: Konsep Dan Teknik,” 2023)

Algoritma detektif tepi Canny diciptakan oleh John F. Canny pada tahun 1986. Algoritma ini dibuat untuk memenuhi tiga kriteria utama: deteksi tepi yang baik, penempatan yang tepat, dan respons minimal. Beberapa langkah utama diambil dalam proses ini: penggunaan filter Gaussian untuk menghaluskan gambar, menghitung gradien intensitas, penghapusan non-maksimal untuk menghilangkan piksel yang bukan merupakan bagian dari tepi, dan penggunaan ambang hysteresis untuk mengidentifikasi dan menghubungkan tepi yang kuat dan lemah. Keunggulan utama metode ini adalah kemampuan untuk mendeteksi tepi dengan presisi tinggi bahkan pada gambar yang mengandung noise. (Krishna, 2023)

Aplikasi visi komputer dan pengolahan citra digunakan oleh OpenCV, pustaka open-source yang awalnya dikembangkan oleh Intel dan saat ini didukung oleh komunitas open-source. Berbagai fitur pemrosesan gambar dan video ditawarkan oleh OpenCV, termasuk deteksi tepi, segmentasi gambar, dan pelacakan objek, antara lain. Kemampuannya untuk bekerja dengan berbagai bahasa pemrograman, seperti C++, Python, dan Java, serta mendukung berbagai sistem operasi, merupakan fitur utama OpenCV. (SAINS, 2023)

2. METODE PENELITIAN

2.1 Persyaratan dan Persiapan

Sebelum menerapkan deteksi tepi Canny, gambar perlu dikonversi ke skala abu-abu dan difilter untuk mengurangi noise. Langkah-langkah ini bertujuan untuk meningkatkan akurasi deteksi tepi.

- a. **Prasyarat:** Instalasi OpenCV dan pustaka pendukung lainnya, seperti Matplotlib untuk visualisasi.
- b. **Persiapan Data:** Pengambilan gambar yang akan dianalisis dan konversi ke format yang sesuai.

2.2 Implementasi Algoritma Canny

Langkah-langkah implementasi deteksi tepi Canny adalah sebagai berikut:

- a. **Menghaluskan Gambar:** Filter Gaussian digunakan untuk mengurangi noise.
- b. **Menghitung Gradien Intensitas:** Sobel operator digunakan untuk menghitung gradien intensitas dalam arah x dan y.
- c. **Non-Maximum Suppression:** Menghilangkan piksel yang bukan bagian dari tepi.
- d. **Hysteresis Thresholding:** Dua ambang batas digunakan untuk mendeteksi dan menghubungkan tepi.

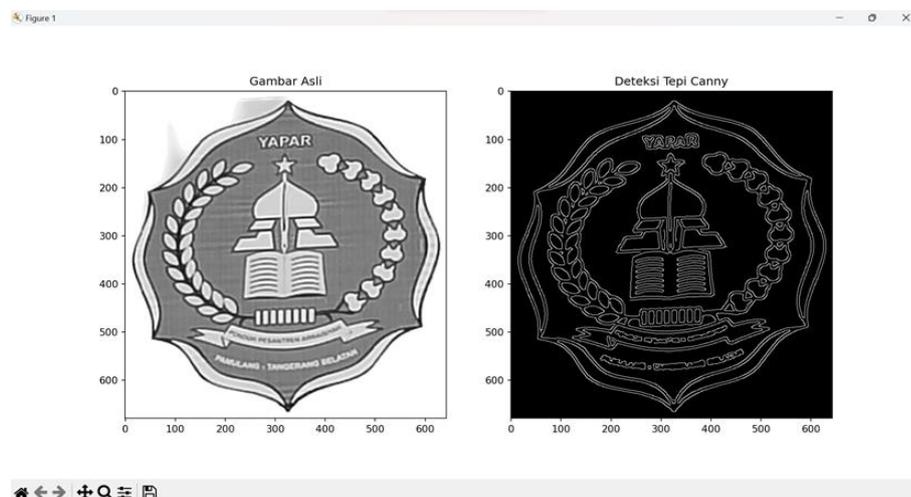
2.3 Implementasi dengan OpenCV

Berikut adalah implementasi dari deteksi tepi Canny OpenCV dan Python:

```

Canny.py X
C: > Users > alyar > Downloads > Canny.py > ...
1 import cv2
2 import matplotlib.pyplot as plt
3
4 # Baca dan konversi gambar ke skala abu-abu
5 image = cv2.imread('C:/Users/alyar/Downloads/tes/logo.jpg', cv2.IMREAD_GRAYSCALE)
6
7 # Terapkan Gaussian blur untuk mengurangi noise
8 blurred_image = cv2.GaussianBlur(image, (5, 5), 1.4)
9
10 # Terapkan Canny Edge Detection
11 edges = cv2.Canny(blurred_image, threshold1=50, threshold2=150)
12
13 # Tampilkan hasil
14 plt.figure(figsize=(10, 5))
15
16 plt.subplot(1, 2, 1)
17 plt.title('Gambar Asli')
18 plt.imshow(image, cmap='gray')
19
20 plt.subplot(1, 2, 2)
21 plt.title('Deteksi Tepi Canny')
22 plt.imshow(edges, cmap='gray')
23
24 plt.show()
    
```

Gambar 1. Source Code



Gambar 2. Hasil Output

Hasil penggunaan Algoritma Canny mendeteksi tepi dengan sangat baik pada gambar asli yang diubah ke skala abu-abu dan difilter dengan blur Gaussian. Hasilnya menunjukkan tepi yang jelas dan terdefinisi dengan baik.

- Gambar Asli: Gambar awal dalam skala abu-abu.
- Gambar Hasil Canny: Gambar yang menunjukkan tepi yang terdeteksi oleh algoritma Canny.

Analisis Hasil Deteksi Tepi Canny menghasilkan hasil yang akurat dengan sedikit noise. Algoritma Canny secara efektif mendeteksi tepi yang signifikan dalam gambar, sedangkan filter Gaussian mengurangi noise dengan baik. Parameter deteksi tepi juga dipengaruhi oleh pemilihan parameter ambang batas, dan memilih parameter yang tepat sangat penting untuk mendapatkan hasil terbaik.

3. ANALISA DAN PEMBAHASAN

3.1 Kelebihan Metode Canny Edge Detection

- a. Deteksi Tepi yang Efektif: Algoritma Canny dapat mendeteksi tepi bahkan pada gambar dengan gangguan atau suara.
- b. Pengurangan Suara: Algoritma ini meningkatkan akurasi hasilnya dengan mengurangi suara sebelum mendeteksi tepi dengan menggunakan filter Gaussian.
- c. Penentuan Tepi yang Tepat: Karena algoritma ini menggunakan teknik penghapusan non-maximum, hanya tepi yang paling penting yang dipertahankan.
- d. Penggunaan Dua Ambang Batas: Algoritma ini mendeteksi tepi yang kuat dan menghubungkan tepi yang lemah, meningkatkan keandalan deteksi tepi. (Sekehravani et al., 2020).

3.2 Kekurangan Metode Canny Edge Detection

- a. Komputasi yang Intensif: Dibandingkan dengan beberapa metode deteksi tepi lainnya, Algoritma Canny lebih kompleks dan memerlukan lebih banyak waktu komputasi.
- b. Sensitivitas terhadap Parameter: Pemilihan parameter, seperti ambang batas dan ukuran filter Gaussian, sangat penting untuk kinerja algoritma. Deteksi tepi yang buruk dapat disebabkan oleh parameter yang salah.
- c. Tidak Berhasil pada Gambar dengan Kontras Rendah: Algoritma ini mungkin kesulitan menemukan tepi pada gambar dengan kontras yang sangat rendah.
- d. Keterbatasan pada Tepi yang Tidak Jelas: Algoritma mungkin tidak menemukan tepi yang tidak jelas atau kabur dengan baik. (Saluky & Yoni Marine, 2023).

3.3 Matplotlib

Matplotlib, yang dikembangkan oleh John D. Hunter, adalah pustaka plotting yang digunakan untuk bahasa pemrograman Python dan pustaka numerik NumPy. Ini menawarkan cara untuk membuat berbagai jenis grafik, termasuk garis, batang, lingkaran, histogram, scatter plot, dan grafik lainnya. Matplotlib sangat bermanfaat untuk visualisasi data dalam bidang teknik, penelitian ilmiah, dan ekonomi, antara lain. Pyplot, salah satu modul Matplotlib yang paling populer, menyediakan antarmuka sederhana untuk membuat grafik dengan sintaks yang mirip dengan MATLAB. (Melanie, 2023).

3.4 Kelebihan Matplotlib

- a. Fleksibel: yang berarti bahwa itu dapat digunakan untuk membuat berbagai jenis grafik.
- b. Kompatibilitas: Pandas, NumPy, dan pustaka Python lainnya dapat digunakan bersama.
- c. Customization: Anda dapat mengubah warna, label, dan gaya garis untuk grafik Anda.
- d. Cross-platform: dapat digunakan pada Windows, macOS, dan Linux.
- e. Interaktivitas: Mendukung pembuatan grafik interaktif yang dapat diintegrasikan ke dalam aplikasi berbasis GUI atau web. (Reyvan, 2021)

3.5 Kekurangan Matplotlib

- a. Kekurangan Kurva Belajar Matplotlib: Kurva belajar Matplotlib agak sulit untuk pemula dalam visualisasi data, meskipun memiliki banyak fitur.
- b. Verbosity: Membuat grafik yang kompleks memerlukan banyak baris kode, yang dapat menjadi verbose dan sulit diatur.
- c. Interaktivitas Terbatas: Matplotlib kurang mendukung interaksi yang kompleks dibandingkan dengan pustaka visualisasi seperti Plotly atau Bokeh.
- d. Kinerja: Matplotlib mungkin lambat dan tidak efisien dibandingkan dengan pustaka yang lebih baru ketika menggunakan data yang sangat besar atau visualisasi real-time.
- e. Desain Standar: Tanpa modifikasi apa pun, desain grafik default Matplotlib cenderung kurang menarik dibandingkan pustaka lain. (Agnes, 2021)

4. KESIMPULAN

Dengan bantuan OpenCV, metode Canny Edge Detection adalah alat yang efektif untuk mendeteksi tepi dalam gambar digital. Implementasi ini menunjukkan bahwa algoritma Canny dapat mendeteksi tepi dengan akurat dan efisien, meskipun hasilnya bergantung pada parameter yang tepat yang dipilih. Pengenalan objek dan analisis gambar adalah dua dari banyak aplikasi pengolahan gambar di mana metode ini sangat bermanfaat.

REFERENCES

- Agnes. (2021). Mengenal Matplotlib untuk Visualisasi Data dengan Python. In *Dqlab.id*. DQLab | Kursus Data Science Online Indonesia R Python. <https://dqlab.id/mengenal-matplotlib-untuk-visualisasi-data-dengan-python>
- Krishna, R. (2023). Canny Edge Detection Algorithm From Scratch. - Rohit Krishna - Medium. In *Medium*. Medium. <https://medium.com/@rohit-krishna/coding-canny-edge-detection-algorithm-from-scratch-in-python-232e1fdceac7>
- Melanie. (2023). Matplotlib: Master Data Visualization in Python. In *Data Science Courses / DataScientest*. <https://datascientest.com/en/matplotlib-master-data-visualization-in-python>
- Pengolahan Citra Digital: Konsep dan Teknik. (2023). In *BINUS UNIVERSITY MALANG | Pilihan Universitas Terbaik di Malang*. <https://binus.ac.id/malang/2023/07/pengolahan-citra-digital-konsep-dan-teknik/>
- Reyvan. (2021). Intip Library Terbaik Python untuk Membuat Visualisasi Data. In *Dqlab.id*. DQLab | Kursus Data Science Online Indonesia R Python. <https://dqlab.id/intip-library-terbaik-python-untuk-membuat-visualisasi-data>
- SAINS, U. (2023). Mengenal library computer vision dan jenis library yang populer | D3 Komputer Grafis A.Md.Ds. In *Stekom.ac.id*. <https://komputer-grafis-d3.stekom.ac.id/informasi/baca/Mengenal-library-computer-vision-dan-jenis-library-yang-populer/50b28b313f7957640c72f0d4db9a128ccfb9fd53>
- Saluky, & Yoni Marine. (2023). Penerapan Algoritma Deteksi Tepi Canny Menggunakan Python Dan Opencv. *Smart Techno (Smart Technology, Informatics and Technopreneurship)*, 5(1), 1–7. <https://doi.org/10.59356/smart-techno.v5i1.73>
- Sekehravani, E. A., Babulak, E., & Masoodi, M. (2020). Implementing canny edge detection algorithm for noisy image. *Bulletin of Electrical Engineering and Informatics*, 9(4), 1404–1410. <https://doi.org/10.11591/eei.v9i4.1837>