

Implementasi Pengolahan Citra Gambar Dengan Program Menggunakan Streamlit

Chaerul Hidayat¹, Rini Ariza², Aan Fadillah Putra³,
Muhammad Reza Maulana⁴

Program Studi Teknik Informatika, Universitas Pelita Bangsa
¹chaerulhidayat70@gmail.com, ²riniariza33@gmail.com,
³aanfadillahputra@gmail.com, ⁴rezamaulana9569@gmail.com

Corresponding Author: Muhammad Fatchan, S.Kom., M.Kom., MTCNA.

ABSTRACT

Image processing is one of the important fields in computer science and information technology. This article discusses the implementation of image processing applications using Streamlit, a Python framework that allows easy creation of interactive web applications. This application includes several main features such as image conversion to grayscale, application of Gaussian Blur, conversion to binary images and inverted binary images, as well as histogram visualization. The application also displays an RGB table of the input image for further analysis.

Keywords: Image processing, Streamlit, Python

ABSTRAK

Pengolahan citra adalah salah satu bidang penting dalam ilmu komputer dan teknologi informasi. Artikel ini membahas tentang implementasi aplikasi pengolahan citra menggunakan Streamlit, sebuah framework Python yang memungkinkan pembuatan aplikasi web interaktif dengan mudah. Aplikasi ini mencakup beberapa fitur utama seperti konversi citra ke grayscale, penerapan Gaussian Blur, konversi ke gambar biner dan gambar biner terbalik, serta visualisasi histogram. Aplikasi ini juga menampilkan tabel RGB dari citra input untuk analisis lebih lanjut.

Kata Kunci: Pengolahan citra, Streamlit, Python

1. PENDAHULUAN

Pengolahan citra digital merupakan teknologi yang telah mengalami perkembangan pesat dan memiliki berbagai aplikasi praktis dalam bidang medis, industri, dan rekayasa [1]. Penggunaan alat dan framework yang tepat sangat penting untuk mengembangkan aplikasi pengolahan citra yang efektif dan efisien. OpenCV (Open Source Computer Vision Library) adalah salah satu pustaka yang paling populer digunakan dalam pengolahan citra digital [2]. Dalam artikel ini, kita memanfaatkan Streamlit, sebuah framework Python, untuk membuat aplikasi web interaktif yang dapat melakukan berbagai operasi pengolahan citra.

2. METODE

2.1. Framework dan Pustaka

Aplikasi ini dibangun menggunakan Streamlit, OpenCV, dan NumPy. Streamlit adalah framework yang memungkinkan pembuatan aplikasi web dengan cepat dan mudah menggunakan Python [3]. OpenCV digunakan untuk berbagai operasi pengolahan citra seperti konversi ke grayscale, Gaussian Blur, dan binarization [4]. NumPy digunakan untuk manipulasi array dan operasi matematika lainnya [5].

2.2. Implementasi Aplikasi

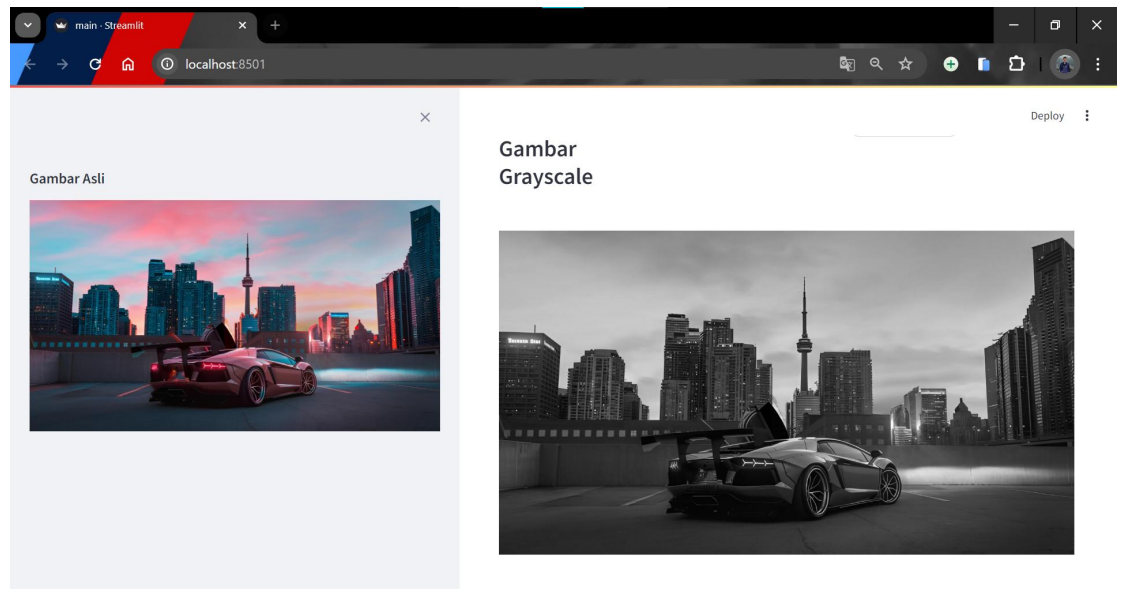
Aplikasi ini menyediakan antarmuka pengguna yang intuitif untuk mengunggah gambar dan menerapkan berbagai operasi pengolahan citra. Berikut adalah fitur-fitur utama dari aplikasi ini:

1. Unggah Gambar: Pengguna dapat mengunggah gambar dalam format JPG, JPEG, atau PNG.
2. Tabel RGB: Menampilkan nilai-nilai RGB dari citra input dalam bentuk tabel untuk analisis lebih lanjut.
3. Konversi ke Grayscale: Mengubah citra berwarna menjadi citra grayscale menggunakan fungsi `cv2.cvtColor`.
4. Gaussian Blur: Menerapkan Gaussian Blur pada citra untuk mengurangi noise dan detail menggunakan fungsi `cv2.GaussianBlur`.
5. Konversi ke Gambar Biner: Mengubah citra grayscale menjadi citra biner menggunakan thresholding dengan fungsi `cv2.threshold`.
6. Konversi ke Gambar Biner Terbalik: Mengubah citra grayscale menjadi citra biner terbalik menggunakan thresholding dengan parameter `cv2.THRESH_BINARY_INV`.
7. Histogram Gambar: Menampilkan histogram dari citra grayscale dan citra berwarna (RGB).

3. HASIL DAN PEMBAHASAN

3.1. Konversi ke Grayscale

Konversi citra ke grayscale adalah operasi dasar dalam pengolahan citra yang mengubah citra berwarna menjadi citra hitam putih. Hal ini berguna untuk menyederhanakan analisis citra dan mengurangi kompleksitas data [6].

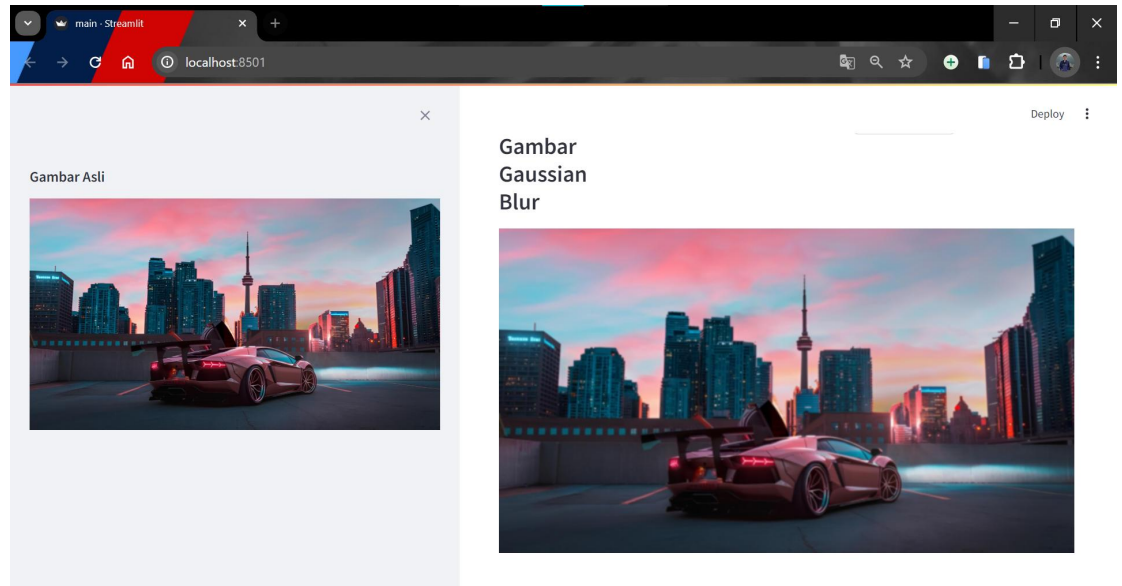


Gambar 1. Gambar Grayscale

Dalam penelitian ini, konversi dilakukan menggunakan fungsi `cv2.cvtColor` dari pustaka OpenCV. Proses ini mengubah citra yang memiliki tiga channel warna (merah, hijau, dan biru) menjadi satu channel intensitas. Hasil dari konversi ini menunjukkan citra grayscale yang hanya memiliki nilai intensitas, yang sering kali lebih mudah untuk dianalisis lebih lanjut dibandingkan citra berwarna. Konversi ke grayscale menyederhanakan citra dengan mengurangi informasi warna, yang sering kali tidak diperlukan untuk operasi pengolahan citra lanjutan seperti deteksi tepi dan segmentasi.

3.2. Gaussian Blur

Gaussian Blur adalah filter yang digunakan untuk mengurangi noise dan detail dalam citra. Filter ini bekerja dengan menghaluskan citra menggunakan kernel Gaussian [7].

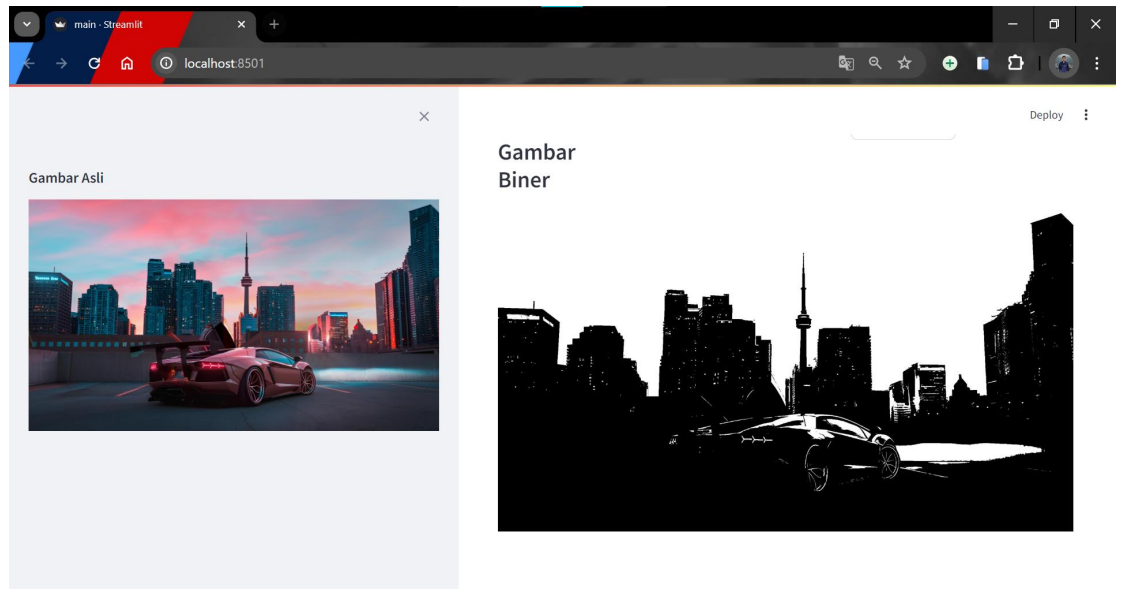


Gambar 2. Gambar Gaussian Blur

Dalam penelitian ini, Gaussian Blur diterapkan menggunakan fungsi `cv2.GaussianBlur`, di mana ukuran kernel yang digunakan adalah 15×15 . Hasil dari penerapan Gaussian Blur menunjukkan citra yang lebih halus dengan detail halus dan noise yang berkurang. Teknik ini sangat berguna sebelum operasi pengolahan citra lainnya, seperti deteksi tepi, untuk mengurangi gangguan yang disebabkan oleh noise dan detail yang tidak diinginkan.

3.3. Konversi ke Gambar Biner

Konversi ke gambar biner adalah proses mengubah citra grayscale menjadi citra hitam putih berdasarkan nilai threshold tertentu. Teknik ini sering digunakan dalam segmentasi citra dan ekstraksi fitur [8].

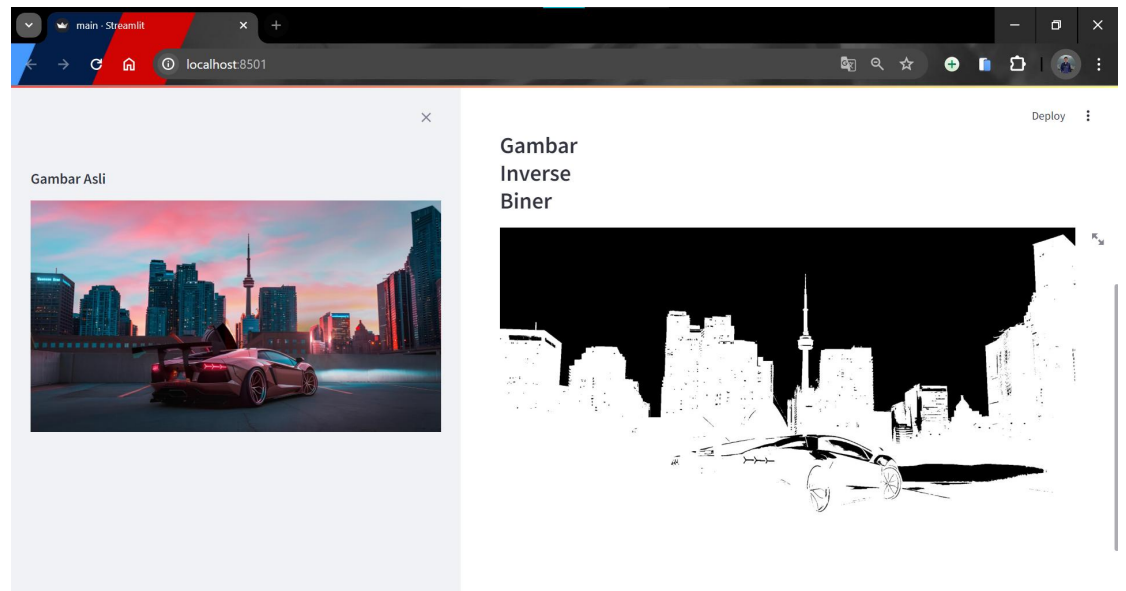


Gambar 3. Gambar Biner

Dalam penelitian ini, konversi dilakukan menggunakan fungsi `cv2.threshold` dengan nilai threshold 127. Hasil dari konversi ini menunjukkan citra yang hanya memiliki dua nilai piksel, yaitu hitam dan putih. Gambar biner membantu dalam memisahkan objek dari latar belakang berdasarkan nilai intensitas. Teknik ini sangat berguna dalam segmentasi dan pengenalan pola di mana informasi detail dari intensitas tidak diperlukan.

3.4. Konversi ke Gambar Biner Terbalik

Gambar biner terbalik adalah hasil dari membalikkan nilai piksel dari gambar biner, di mana hitam menjadi putih dan putih menjadi hitam. Ini dilakukan dengan menggunakan parameter `cv2.THRESH_BINARY_INV` dalam fungsi `cv2.threshold`. Berikut adalah contoh hasil dari konversi ke gambar biner terbalik:

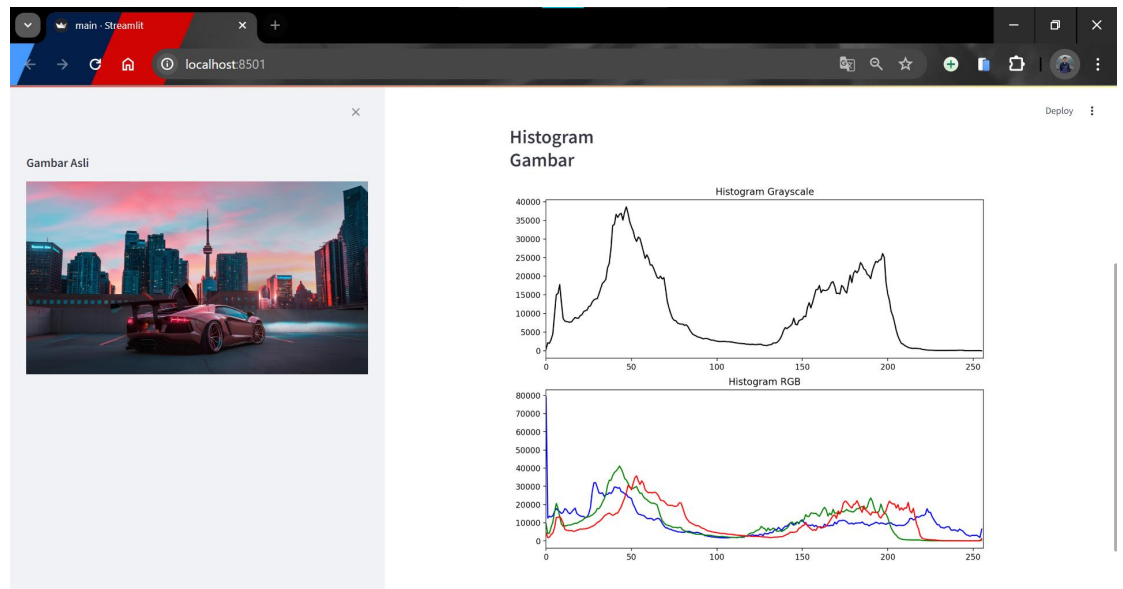


Gambar 4. Gambar Inverse Biner

Dalam penelitian ini, konversi dilakukan dengan menggunakan fungsi `cv2.threshold` dengan parameter `cv2.THRESH_BINARY_INV`. Hasil dari konversi ini menunjukkan gambar biner terbalik yang sangat berguna ketika objek utama memiliki intensitas lebih rendah daripada latar belakang. Teknik ini membantu dalam menyoroti objek gelap pada latar belakang terang, memudahkan deteksi dan analisis objek.

3.5. Histogram Gambar

Histogram citra memberikan informasi tentang distribusi intensitas piksel dalam citra. Histogram grayscale menunjukkan distribusi intensitas dalam citra hitam putih, sedangkan histogram RGB menunjukkan distribusi intensitas dalam setiap channel warna [9].



Gambar 4. Histogram Gambar

Dalam penelitian ini, histogram grayscale dihitung menggunakan fungsi `cv2.calcHist`, dan histogram RGB dihitung untuk setiap channel warna. Hasil dari histogram grayscale menunjukkan distribusi intensitas dalam citra hitam putih, sementara histogram RGB menunjukkan distribusi intensitas dalam setiap channel warna. Histogram citra memberikan informasi penting tentang distribusi intensitas piksel yang membantu dalam analisis kontras dan kecerahan citra. Analisis histogram dapat digunakan untuk operasi pengolahan citra lainnya seperti kontras stretching dan thresholding.

4. KESIMPULAN

Artikel ini telah membahas implementasi aplikasi pengolahan citra menggunakan Streamlit dan OpenCV. Aplikasi ini mencakup berbagai fitur pengolahan citra yang penting seperti konversi ke grayscale, Gaussian Blur, konversi ke gambar biner, konversi ke gambar biner terbalik, dan histogram gambar. Implementasi ini menunjukkan betapa kuatnya penggunaan Python dalam pengembangan aplikasi pengolahan citra interaktif. Aplikasi ini tidak hanya berguna untuk keperluan akademis dan penelitian tetapi juga dapat digunakan dalam berbagai aplikasi praktis seperti inspeksi industri dan analisis medis.

DAFTAR PUSTAKA

- [1] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed. Pearson, 2018.
- [2] G. Bradski and A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly Media, 2008.
- [3] Streamlit Inc., "Streamlit," [Online]. Available: <https://streamlit.io/>.
- [4] OpenCV, "OpenCV," [Online]. Available: <https://opencv.org/>.
- [5] T. Oliphant, Guide to NumPy, USA: Trelgol Publishing, 2006.
- [6] Z. Zhang, "Image Processing and Analysis," in Handbook of Research on Computational Intelligence Applications in Bioinformatics, IGI Global, 2016.
- [7] D. Marr and E. Hildreth, "Theory of edge detection," Proceedings of the Royal Society of London. Series B. Biological Sciences, vol. 207, no. 1167, pp. 187-217, 1980.
- [8] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62-66, 1979.
- [9] R. C. Gonzalez and R. E. Woods, Digital Image Processing Using MATLAB, 2nd ed. Gatesmark Publishing, 2009.