

Pengolahan Citra Digital Dengan Penerapan Teknik Ambang Batas: Studi Kasus Menggunakan OpenCV

**Allyssa Zahra Salsa Azizah^{1*}, Fuaz Nazar To Yalis², Ikade Shiva Narayana³,
Karen Azelia Syalom⁴, Perani Rosyani⁵**

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

Email: ^{1*}allyssazahra19@gmail.com, ²fuaznazartoyalis08@gmail.com,

³shivanarayana08081995@gmail.com, ⁴karen.azelia29@gmail.com, ⁵dosen00837@unpam.ac.id.

(* : coresponding author)

Abstrak— Pemrosesan citra digital adalah bidang yang berkembang pesat dalam ilmu komputer dan teknologi informasi. Tujuannya adalah untuk meningkatkan kualitas gambar, mengekstrak informasi berguna, dan melakukan analisis kompleks yang tidak dapat dilakukan secara manual. Salah satu teknik dasar pengolahan citra adalah Thresholding. Hal ini bertujuan untuk memisahkan objek dari latar belakang dengan mengubah gambar menjadi gambar biner. Jurnal ini menjelaskan implementasi program pengolahan citra digital menggunakan Python dan OpenCV dengan menerapkan tiga teknik ambang batas: Global, Adaptif (Average), dan Adaptif (Gaussian). Kami juga menyajikan metode pengurangan noise yang menggunakan keburaman rata-rata sebelum menerapkan ambang batas. Proses implementasinya dilakukan dengan membaca gambar dalam mode skala abu-abu, mereduksi noise dan menerapkan berbagai teknik ambang batas. Hasil dari masing-masing metode dievaluasi dan dibandingkan, dan efektivitasnya dalam memisahkan objek dari latar belakang dianalisis.

Kata Kunci: Pemrosesan Citra Digital, Ambang Batas

Abstract— *Digital image processing is a rapidly growing field in computer science and information technology. Its main goals are to enhance image quality, extract useful information, and perform complex analyses that cannot be done manually. One fundamental technique in image processing is Thresholding. Its purpose is to separate objects from the background by converting an image into a binary image. This journal explains the implementation of digital image processing using Python and OpenCV, focusing on three thresholding techniques: Global, Adaptive (Average), and Adaptive (Gaussian). We also present a noise reduction method using average blurring before applying thresholding. The implementation process involves reading images in grayscale mode, reducing noise, and applying various thresholding techniques. The results of each method are evaluated and compared, analyzing their effectiveness in separating objects from the background.*

Keywords: *Digital Image Processing, Thresholding*

1. PENDAHULUAN

Pemrosesan gambar digital adalah bidang yang berkembang pesat dalam dunia ilmu komputer dan teknologi informasi. Pertumbuhannya didorong oleh meningkatnya kebutuhan untuk meningkatkan kualitas gambar, mengekstrak informasi yang relevan, dan melakukan analisis kompleks yang sulit dilakukan secara manual. Kemajuan ini dapat diterapkan di berbagai bidang seperti pencitraan medis, penginderaan jauh, dan sistem pengawasan. Salah satu teknik dasar pengolahan citra digital adalah penentuan ambang batas. Teknik ini berperan penting dalam memisahkan objek dari latar belakang dengan mengubah gambar skala abu-abu atau berwarna menjadi gambar biner. Menetapkan ambang batas mengklasifikasikan piksel dalam gambar sebagai latar depan (objek) atau latar belakang berdasarkan tingkat intensitasnya. Metode ambang batas dapat dibagi menjadi dua kategori utama: global dan adaptif. Ambang batas global menerapkan satu ambang batas ke seluruh gambar, sedangkan ambang batas adaptif menyesuaikan ambang batas secara lokal, biasanya berdasarkan rata-rata intensitas piksel lokal atau Gaussian. (binus.ac.id, 2023)

Metode ini sangat penting dalam berbagai aplikasi seperti segmentasi citra, deteksi objek, dan ekstraksi fitur. Penerapan teknik ambang batas sering kali mencakup langkah prapemrosesan untuk meningkatkan kualitas gambar dan mengurangi noise. Teknik seperti pemfilteran median dan pemfilteran Gaussian biasanya digunakan untuk mencapai hal ini. Selain itu, metode ambang batas dan pemilihan parameter yang digunakan memiliki dampak signifikan terhadap keakuratan dan efisiensi tugas pemrosesan gambar selanjutnya. Dalam konteks ini, penelitian ini bertujuan untuk

mengeksplorasi dan menerapkan teknik pemrosesan gambar digital menggunakan bahasa pemrograman Python dan perpustakaan OpenCV. (Nida Zaitun, 2015)

Penelitian ini berfokus secara khusus pada perbandingan dan analisis efektivitas global ambang batas, ambang batas adaptif (berdasarkan metode rata-rata dan Gaussian), dan strategi pengurangan kebisingan yang digunakan. Hasil eksperimen ini dievaluasi untuk menentukan penerapan dan kinerjanya dalam berbagai skenario dunia nyata. Melalui penelitian ini, kami berharap dapat memberikan wawasan berharga mengenai pemilihan dan penerapan teknik ambang batas yang optimal untuk meningkatkan efisiensi dan akurasi tugas pemrosesan gambar digital.

2. METODE PENELITIAN

2.1 Implementasi Metode

Thresholding adalah teknik OpenCV yang menentukan nilai piksel relatif terhadap ambang batas tertentu. Thresholding membandingkan setiap nilai piksel dengan ambang batasnya. Jika nilai piksel kurang dari ambang batas, maka disetel ke 0, jika tidak maka disetel ke nilai maksimum (biasanya 255). Thresholding adalah teknik segmentasi yang sangat umum yang memisahkan objek yang dianggap latar depan dari latar belakang. Ambang batas memiliki dua rentang di kedua sisinya: nilai di bawah dan di atas ambang batas. Dalam visi komputer, teknik ambang batas ini diterapkan pada gambar skala abu-abu. (rishabhhsingh1304, 2023)

Oleh karena itu, perlu mengonversi gambar menjadi ruang warna skala abu-abu terlebih dahulu. Teknik dasar ambang batas adalah teknik ambang batas biner. Ambang batas yang sama diterapkan pada setiap piksel. Jika nilai piksel kurang dari ambang batas, maka disetel ke 0, jika tidak maka disetel ke nilai maksimum. (Cheng Lu, 2018)

Berbagai teknik ambang batas sederhana meliputi:

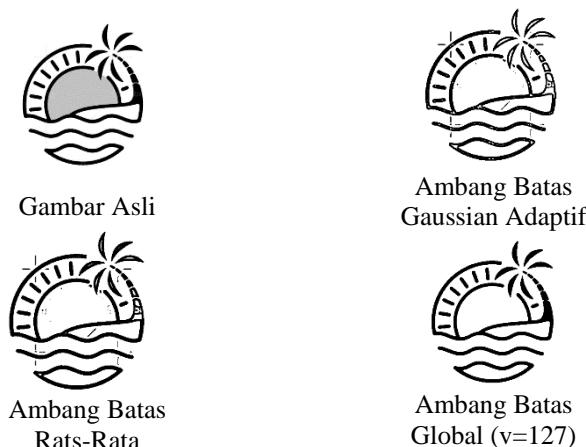
- cv2.THRESH_BINARY: Jika intensitas piksel lebih besar dari ambang batas yang ditentukan, nilainya disetel ke 255, jika tidak maka disetel ke 0 (hitam).
- cv2.THRESH_BINARY_INV: Kebalikan atau kebalikan dari cv2.THRESH_BINARY.
- cv.THRESH_TRUNC: Jika nilai intensitas piksel lebih besar dari ambang batas, maka akan terpotong hingga mencapai ambang batas. Nilai piksel diatur ke ambang batas. Semua nilai lainnya tetap sama.
- cv.THRESH_TOZERO: Intensitas piksel disetel ke 0 untuk semua intensitas piksel di bawah ambang batas.
- cv.THRESH_TOZERO_INV: Kebalikan atau kebalikan dari cv2.THRESH_TOZERO.

2.2 Implementasi Menggunakan OpenCV

Berikut adalah langkah-langkah yang diambil dalam implementasi kode untuk memahami dan menerapkan teknik ambang batas menggunakan OpenCV:

1. Membaca Gambar dalam Mode Grayscale
 - a. Membaca gambar 'pp.jpg' dalam mode grayscale menggunakan OpenCV.
 - b. Memeriksa kesuksesan pembacaan gambar untuk memastikan kelengkapan proses.
2. Mengurangi Noise dengan Median Blur
 - a. Mengurangi noise pada gambar sebelum menerapkan ambang batas.
 - b. Tujuannya adalah untuk memastikan hasil ambang batas lebih bersih dan lebih akurat.
3. Menggunakan Ambang Batas Global
 - a. Menggunakan ambang batas global dengan nilai ambang batas 127.
 - b. Menghasilkan citra biner dengan nilai 0 untuk piksel yang kurang dari 127 dan nilai 255 untuk piksel yang lebih dari 127.
4. Menggunakan Ambang Batas Adaptif (Mean)
 - a. Menggunakan ambang batas adaptif dengan rata-rata lokal intensitas piksel.

- b. Menghasilkan citra biner dengan nilai 0 untuk piksel yang kurang dari rata-rata dan nilai 255 untuk piksel yang lebih dari rata-rata.
- 5. Menggunakan Ambang Batas Adaptif (Gaussian)
 - a. Menggunakan ambang batas adaptif dengan rata-rata tertimbang Gaussian intensitas piksel.
 - b. Menghasilkan citra biner dengan nilai 0 untuk piksel yang kurang dari rata-rata dan nilai 255 untuk piksel yang lebih dari rata-rata.
- 6. Menampilkan Hasil
 - a. Menampilkan gambar asli dan hasil dari tiga teknik ambang batas menggunakan jendela OpenCV untuk visualisasi.



Gambar 1. Proses Implementasi

Keterangan:

- a. Gambar asli : Gambar grayscale yang sudah dihaluskan dengan media blur.
- b. Ambang Batas Gaussian Adaptif : Menampilkan hasil adaptif thresholding dengan metode Gaussian
- c. Ambang Batas Rata-Rata : Hasil adaptif thresholding dengan metode mean.
- d. Ambang Batas Global (V=127) : Hasil adaptif thresholding dengan nilai ambang batas 127.

2.3 Implementasi Kode

Berikut Source code yg kami gunakan dalam membuat project ini :

```

  Ambang Gambar.py
C:\> Users > tian > Downloads > python > Ambang Gambar.py > 
1  import cv2 as cv
2  import numpy as np
3  import os
4
5  # Membaca gambar dalam mode grayscale
6  img = cv.imread('sunset.jpg', cv.IMREAD_GRAYSCALE)
7  assert img is not None, "File tidak dapat dibaca, periksa dengan os.path.exists()"
8
9  # Mengurangi noise dengan median blur
10 img = cv.medianBlur(img, 5)
11
12 # Menggunakan ambang batas global
13 ret, th1 = cv.threshold(img, 127, 255, cv.THRESH_BINARY)
14
15 # Menggunakan ambang batas adaptif (mean)
16 th2 = cv.adaptiveThreshold(img, 255, cv.ADAPTIVE_THRESH_MEAN_C, cv.THRESH_BINARY, 11, 2)
17
18 # Menggunakan ambang batas adaptif (Gaussian)
19 th3 = cv.adaptiveThreshold(img, 255, cv.ADAPTIVE_THRESH_GAUSSIAN_C, cv.THRESH_BINARY, 11, 2)
20
21 # Menampilkan hasil menggunakan OpenCV
22 cv.imshow('Gambar Asli', img)
23 cv.imshow('Ambang Batas Global (v = 127)', th1)
24 cv.imshow('Ambang Rata-rata Adaptif', th2)
25 cv.imshow('Ambang Batas Gaussian Adaptif', th3)
26
27 # Memunggu sampai tombol sembarang ditekan
28 cv.waitKey(0)
29 cv.destroyAllWindows()
30

```

Gambar 2. Sourcecode Program

Berikut penjelasan singkat mengenai kode Python di atas yang berfungsi untuk membaca dan memproses gambar menggunakan OpenCV dan NumPy:

1. Import Library:

```
python
import cv2 as cv
import numpy as np
import os
```

Baris ini mengimpor library yang diperlukan: OpenCV (cv2), NumPy (numpy), dan os untuk operasi sistem.

2. Membaca Gambar dalam Mode Grayscale:

```
python
img = cv.imread('sunset.jpg', cv.IMREAD_GRAYSCALE)
assert img is not None, "File tidak dapat dibaca, periksa dengan os.path.exists()"
```

Kode ini membaca gambar sunset.jpg dalam mode grayscale. Jika gambar tidak terbaca, program akan menampilkan pesan kesalahan.

3. Mengurangi Noise dengan Median Blur:

```
python
img = cv.medianBlur(img, 5)
```

Baris ini mengurangi noise pada gambar menggunakan filter median dengan ukuran kernel 5.

4. Menggunakan Ambang Batas Global:

```
python
ret, th1 = cv.threshold(img, 127, 255, cv.THRESH_BINARY)
```

Kode ini menerapkan ambang batas global pada gambar dengan nilai ambang batas 127 dan nilai maksimum 255.

5. Menggunakan Ambang Batas Global:

```
python
th2 = cv.adaptiveThreshold(img, 255, cv.ADAPTIVE_THRESH_MEAN_C,
cv.THRESH_BINARY, 11, 2)
```

Baris ini menerapkan ambang batas adaptif menggunakan mean pada gambar.

6. Menggunakan Ambang Batas Adaptif (Gaussian):

```
python
th3 = cv.adaptiveThreshold(img, 255, cv.ADAPTIVE_THRESH_GAUSSIAN_C,
cv.THRESH_BINARY, 11, 2)
```

Baris ini menerapkan ambang batas adaptif menggunakan Gaussian pada gambar.

7. Menampilkan Hasil dengan OpenCV:

```
python
cv.imshow('Gambar Asli', img)
cv.imshow('Ambang Batas Global (v = 127)', th1)
cv.imshow('Ambang Rata-rata Adaptif', th2)
cv.imshow('Ambang batas Gaussian Adaptif', th3)
cv.waitKey(0)
cv.destroyAllWindows()
```

Kode ini menampilkan gambar asli dan gambar hasil proses ambang batas menggunakan OpenCV. Program akan menunggu hingga tombol sembarang ditekan sebelum menutup semua jendela gambar. Kode ini secara keseluruhan bertujuan untuk membandingkan berbagai teknik thresholding (ambang batas) pada gambar, baik secara global maupun adaptif, dan menampilkan hasilnya.

3. ANALISA DAN PEMBAHASAN

Dalam proses pengolahan gambar, Anda dapat melakukan langkah-langkah berikut untuk mendapatkan hasil yang lebih baik:

- Pertama, gambar "sunset.jpg" dibaca dalam mode skala abu-abu menggunakan OpenCV untuk memastikan keberhasilan pembacaan dan memastikan integritas proses.
- Selain itu, kami menggunakan teknik blur terpusat untuk mengurangi noise sebelum menerapkan ambang batas, menjadikan hasil ambang batas lebih bersih dan akurat.
- Metode ambang batas global kemudian diterapkan dengan ambang batas 127, menghasilkan gambar biner dimana piksel dengan nilai kurang dari 127 ditetapkan ke 0 dan piksel dengan nilai lebih tinggi ditetapkan ke 255 akan dihasilkan.
- Untuk lebih mengakomodasi variasi intensitas lokal, ambang batas adaptif tambahan digunakan berdasarkan rata-rata, dengan piksel dengan nilai di bawah rata-rata dipetakan ke 0 dan nilai di atasnya dipetakan ke 255.
- Terakhir, dalam pendekatan adaptif berbasis Gaussian, intensitas piksel dievaluasi dengan bobot Gaussian dan gambar biner serupa dihasilkan dengan mempertimbangkan rata-rata tertimbang.

Langkah-langkah ini dirancang untuk meningkatkan akurasi dan kebersihan hasil pengolahan citra digital.

4. KESIMPULAN

Beberapa kesimpulan dapat ditarik dari penelitian di atas ialah:

- Teknik thresholding sangat penting dalam berbagai aplikasi seperti segmentasi citra, deteksi objek, dan ekstraksi fitur.
- Penerapan teknik ambang batas seringkali mencakup langkah pra-pemrosesan untuk meningkatkan kualitas gambar dan mengurangi noise yang ada.
- Pemilihan metode ambang batas dan parameter yang sesuai memiliki pengaruh yang signifikan terhadap keakuratan dan efisiensi tugas pemrosesan gambar selanjutnya.
- Hasil eksperimen ini dievaluasi untuk menilai penerapan dan kinerja teknik dalam berbagai skenario dunia nyata.
- Penelitian ini diharapkan dapat memberikan wawasan berharga dalam pemilihan dan penerapan teknik ambang batas yang optimal untuk meningkatkan efisiensi dan akurasi pemrosesan citra digital.

REFERENCES

- (PPKP), P. P. (t.thn.). Diambil kembali dari <https://uls.ubaya.ac.id/course/info.php?id=1442>
- Anggraeni, D. T. (2021, December 27). Perbaikan Citra Dokumen Hasil Pindai Menggunakan Metode Simple, Adaptive Gaussian dan Otsu Binarization Thresholding. Diambil kembali dari <https://media.neliti.com/media/publications/391993-none-dbc2f496.pdf>
- binus.ac.id. (2023, July). Pengolahan Citra Digital: Konsep dan Teknik. Diambil kembali dari <https://binus.ac.id/malang/2023/07/pengolahan-citra-digital-konsep-dan-teknik/>
- Cheng Lu, A. M. (2018). Analisis Citra Medis Berbasis Soft Computing. Diambil kembali dari <https://www.sciencedirect.com/topics/engineering/global-thresholding>
- Nida Zaitun, M. J. (2015, December). Teknik Segmentasi Citra. Survei Teknik Segmentasi Citra. doi: 10.1016/j.procs.2015.09.027
- rishabhhsingh1304. (2023). piton ular | Teknik Thresholding Menggunakan OpenCV | Set-1 (Ambang Batas Sederhana). Diambil kembali dari <https://www.geeksforgeeks.org/python-thresholding-techniques-using-opencv-set-1-simple-thresholding/>
- University, B. (2023, July). Pengolahan Citra Digital: Konsep dan Teknik. Diambil kembali dari <https://binus.ac.id/malang/2023/07/pengolahan-citra-digital-konsep-dan-teknik/>
- Yuhandri, Y. R. (2022). Pengenalan Teknologi Pengolahan Citra Digital. Pengenalan Teknologi Pengolahan Citra Digital (Digital Image Processing) Untuk Santri Di Rahmatan Lil'Alamin International Islamic Boarding School. doi:<https://doi.org/10.31004/cdj.v3i2.5868>