

LAPORAN TUGAS INDIVIDU

IF5151 COMPUTER VISION

Implementasi Pipeline Terintegrasi untuk Analisis Citra

Menggunakan Konsep Filtering, Edge Detection, dan Feature Extraction



Nama : Marzuli Suhada M

NIM : 13522070

Kelas : 01

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

DAFTAR ISI

| | |
|--|----|
| DAFTAR ISI | 2 |
| DAFTAR TABEL | 4 |
| DAFTAR GAMBAR | 5 |
| WORKFLOW PIPELINE | 7 |
| PROSES & HASIL TIAP FITUR | 10 |
| 1. Image Filtering | 10 |
| 1.1 Teori Singkat | 10 |
| 1.2 Parameter yang Digunakan | 11 |
| Tabel 1.1. Parameter Image Filtering | 11 |
| 1.3 Hasil Eksperimen pada Gambar Standar | 12 |
| Tabel 1.3.1 Tabel Ringkas Hasil Eksperimen | 14 |
| 1.4 Analisis Efek Perubahan Parameter | 15 |
| 2. Edge Detection & Sampling | 16 |
| 2.1 Teori Singkat | 16 |
| 2.2 Parameter yang Digunakan | 16 |
| Tabel 2.1. Parameter Edge Detection dan Sampling | 16 |
| 2.3 Hasil Eksperimen | 17 |
| Tabel 2.3.1. Ringkasan Hasil Edge Detection pada Cameraman | 20 |
| Tabel 2.3. Efek Sampling pada Cameraman | 22 |
| 2.4 Analisis Efek Perubahan Parameter | 22 |
| Tabel 2.4.1. Ringkasan Efek Perubahan Parameter | 22 |
| 3. Feature Points | 23 |
| 3.1 Teori Singkat | 23 |
| 3.2 Parameter yang Digunakan | 23 |

| | |
|---|----|
| 3.3 Hasil Eksperimen | 24 |
| 3.3.1 Hasil Harris Corner Detection | 24 |
| 3.3.2 Hasil SIFT | 26 |
| 3.3.3 Hasil FAST | 28 |
| Tabel 3.3.1. Ringkasan hasil deteksi fitur pada berbagai citra | 29 |
| 3.4 Analisis Efek Perubahan Parameter | 29 |
| 4. Camera Geometry & Calibration | 30 |
| 4.1 Teori Singkat | 30 |
| 4.2 Parameter yang Digunakan | 30 |
| 4.3 Hasil Eksperimen | 31 |
| Tabel 4.3.1. Ringkasan Parameter & Hasil Transformasi Geometrik | 33 |
| 4.4 Analisis Efek Perubahan Parameter | 33 |
| KOMPARASI & REFLEKSI PRIBADI | 35 |
| 1. Komparasi Hasil: Gambar Standar vs Gambar Pribadi | 35 |
| 1.1. Image Filtering | 35 |
| 1.2. Edge Detection & Sampling | 35 |
| 1.3. Feature Points Detection | 35 |
| 1.4. Camera Geometry & Calibration | 36 |
| 2. Refleksi Pribadi: Pilihan Desain & Inovasi | 36 |
| 3. Kesimpulan Komparasi | 37 |
| 4. Refleksi Personal | 37 |
| DAFTAR PUSTAKA | 38 |
| LAMPIRAN | 39 |

DAFTAR TABEL

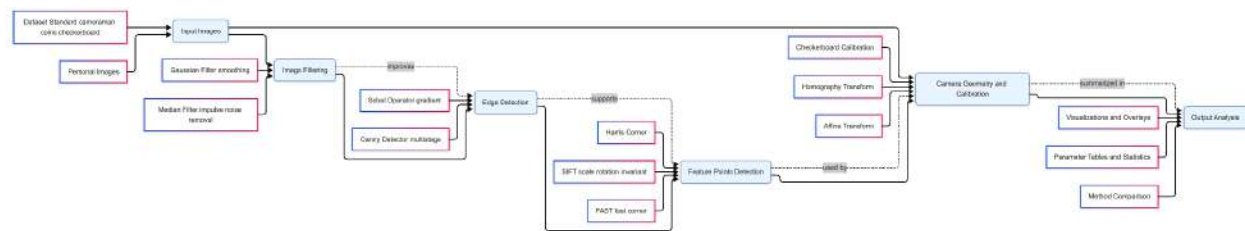
| | |
|---|----|
| Tabel 1.1. Parameter Image Filtering | 11 |
| Tabel 1.3.1 Tabel Ringkas Hasil Eksperimen | 14 |
| Tabel 2.1. Parameter Edge Detection dan Sampling | 16 |
| Tabel 2.3.1. Ringkasan Hasil Edge Detection pada Cameraman | 20 |
| Tabel 2.3. Efek Sampling pada Cameraman | 22 |
| Tabel 2.4.1. Ringkasan Efek Perubahan Parameter | 22 |
| Tabel 3.3.1. Ringkasan hasil deteksi fitur pada berbagai citra | 29 |
| Tabel 4.3.1. Ringkasan Parameter & Hasil Transformasi Geometrik | 33 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 1. Workflow Pipeline Sistem Computer Vision Terintegrasi | 7 |
| Gambar 2.3.1. Hasil Deteksi Tepi Canny pada Cameraman | 18 |
| Gambar 2.3.2. Hasil Deteksi Tepi Sobel pada Cameraman | 20 |
| Gambar 2.3.3. Hasil Deteksi Tepi Sampling pada Cameraman | 21 |
| Gambar 3.3.1. Hasil Harris pada Cameraman, Astronaut, dan Azul | 26 |
| Gambar 4.3.1. Hasil Homography pada citra Cameraman, Astronaut, dan Azul | 31 |
| Gambar 4.3.2. Hasil Affine Transform pada citra Cameraman, Astronaut, dan Azul | 32 |

WORKFLOW PIPELINE

Pipeline aplikasi ini menggambarkan urutan utama dalam sistem Computer Vision, mulai dari tahap pengolahan awal citra (filtering), deteksi tepi (edge detection), identifikasi titik-titik fitur penting (feature points detection), hingga kalibrasi dan transformasi geometri kamera (camera geometry & calibration). Alur ini memperlihatkan bagaimana setiap tahap saling terhubung untuk membentuk sistem analisis citra yang utuh dan terintegrasi dari input hingga output akhir.



Gambar 1. Workflow Pipeline Sistem Computer Vision Terintegrasi

Tahapan pipeline dijelaskan sebagai berikut:

1. Input Images

Tahap awal berupa masukan citra dari dua sumber:

- Dataset standar yang telah ditetapkan (misalnya cameraman.png, coins.png, dan checkerboard.png) sebagai baseline.
- Gambar pribadi (foto mandiri atau objek nyata) untuk eksplorasi tambahan dan refleksi hasil. Pada tugas ini saya menggunakan foto saya sendiri yaitu azul.png dan foto dari madam eva (kucing yang terkenal di ITB) yaitu madam_eva.png. Seluruh citra dikonversi ke format grayscale atau RGB sesuai kebutuhan masing-masing fitur.

2. Image Filtering

Filtering bertujuan untuk mengurangi noise dan memperhalus citra agar struktur utama tetap terjaga. Dua filter utama digunakan yaitu *Gaussian Filter* (reduksi noise dengan smoothing berbasis distribusi normal) dan *Median Filter* (penghapusan noise impulsif tanpa mengaburkan tepi). Hasil dari tahap ini menjadi input penting bagi *edge detection*, karena kualitas filtering memengaruhi ketajaman tepi yang terdeteksi.

- Citra yang difilter terlalu agresif (sigma besar) akan kehilangan detail tepi.
- Sebaliknya, filtering yang terlalu ringan akan meninggalkan noise yang dapat menghasilkan tepi palsu.

3. Edge Detection

Tahap ini berfungsi untuk mengekstraksi batas-batas objek dari citra hasil filtering. Dua metode digunakan, yaitu Sobel (pendeteksian gradien intensitas sederhana) dan Canny (deteksi tepi multi-tahap dengan thresholding adaptif). Hasil edge detection tidak hanya penting untuk analisis struktur, tetapi juga menjadi masukan tambahan untuk tahap *feature points detection*, karena tepi yang jelas membantu dalam mengidentifikasi titik-titik sudut atau fitur signifikan. Analisis *sampling* dilakukan untuk mengevaluasi pengaruh resolusi dan threshold terhadap ketepatan hasil deteksi.

4. Feature Points Detection

Setelah tepi objek diperoleh, tahap ini mengekstraksi titik-titik khas (*keypoints*) yang menjadi ciri utama citra. Beberapa metode digunakan, seperti Harris Corner Detection, SIFT (Scale-Invariant Feature Transform), dan FAST (Features from Accelerated Segment Test). Setiap metode memiliki karakteristik berbeda:

- Harris fokus pada deteksi sudut berbasis intensitas gradien.
- SIFT mengekstraksi fitur yang stabil terhadap perubahan skala dan rotasi.
- FAST mendeteksi fitur dengan cepat menggunakan analisis segmentasi piksel di sekitar titik pusat.

Hasil fitur ini digunakan untuk perbandingan antar-citra atau sebagai titik acuan dalam kalibrasi geometri.

5. Camera Geometry & Calibration

Tahap ini mensimulasikan aspek geometri kamera melalui kalibrasi, transformasi homografi, dan transformasi affine menggunakan pola checkerboard sebagai referensi. Tujuannya untuk memahami hubungan antara koordinat dunia nyata dan citra hasil proyeksi kamera.

Dari tahap ini dihasilkan matriks parameter kamera (intrinsik dan/atau ekstrinsik) serta visualisasi overlay hasil transformasi pada gambar asli. Kesalahan deteksi titik kalibrasi dapat memengaruhi hasil transformasi, sehingga akurasi titik fitur dari tahap sebelumnya menjadi sangat penting.

6. Output Analysis

Semua hasil akhir dikompilasi dalam bentuk visualisasi, tabel parameter, dan statistik numerik, termasuk jumlah fitur, nilai threshold, serta perbandingan hasil antar-metode. Analisis ini memberikan wawasan tentang hubungan antar parameter dan bagaimana perubahan pada tahap awal (misalnya sigma pada Gaussian) dapat berdampak pada hasil akhir pipeline.

PROSES & HASIL TIAP FITUR

1. Image Filtering

1.1 Teori Singkat

Image filtering merupakan salah satu proses dasar dalam bidang *computer vision* yang bertujuan untuk memodifikasi nilai intensitas piksel berdasarkan operasi pada area sekitarnya (*neighborhood*). Proses ini berfungsi untuk meningkatkan kualitas citra sebelum dilakukan tahapan analisis lanjutan, seperti *edge detection* dan *feature extraction*.

Secara umum, tujuan utama dari image filtering meliputi:

1. Noise Reduction, yaitu mengurangi gangguan (*noise*) yang terdapat pada citra agar hasil analisis menjadi lebih stabil.
2. Edge Enhancement, yaitu memperjelas batas-batas objek agar struktur citra lebih mudah dikenali.
3. Feature Preparation, yaitu menyiapkan citra agar hasil ekstraksi fitur pada tahap berikutnya menjadi lebih akurat.

Dalam tugas ini, tiga jenis filter diimplementasikan, yaitu Gaussian Filter, Median Filter, dan Sobel Filter. Ketiganya memiliki karakteristik dan fungsi yang berbeda, sebagaimana dijelaskan berikut:

1. Gaussian Filter

Filter ini menggunakan distribusi Gaussian dua dimensi sebagai kernel untuk melakukan *smoothing*. Nilai piksel baru dihitung berdasarkan bobot tetangganya yang mengikuti distribusi normal, sehingga hasilnya halus dan alami. Rumus matematisnya adalah

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Semakin besar nilai σ (*sigma*), maka efek *smoothing* yang dihasilkan akan semakin kuat.

2. Median Filter

Filter non-linear yang menggantikan nilai piksel dengan nilai median dari piksel-piksel di

sekitarnya. Filter ini sangat efektif untuk menghapus *salt-and-pepper noise* tanpa mengaburkan tepi, sehingga sering digunakan dalam citra yang memiliki noise impulsif.

3. Sobel Filter

Filter ini digunakan untuk mendeteksi tepi (*edges*) dengan menghitung gradien intensitas citra baik dalam arah horizontal maupun vertikal. Rumus umum gradien Sobel adalah:

$$G = \sqrt{G_x^2 + G_y^2}$$

di mana G_x dan G_y merupakan hasil konvolusi kernel Sobel pada arah horizontal dan vertikal.




1.2 Parameter yang Digunakan




Eksperimen dilakukan dengan variasi parameter pada masing-masing filter untuk menemukan konfigurasi optimal. Tabel berikut menunjukkan parameter yang diuji:




Tabel 1.1. Parameter Image Filtering

| Filter | Parameter | Nilai yang Dicoba | Deskripsi |
|----------|--------------------|--------------------|-------------------------------------|
| Gaussian | Kernel Size | 3, 5, 7, 9 | Ukuran jendela konvolusi |
| Gaussian | Sigma (σ) | 0.5, 1.0, 1.5, 2.0 | Standar deviasi distribusi Gaussian |
| Median | Kernel Size | 3, 5, 7, 9 | Ukuran jendela median |
| Sobel | Kernel Size | 3 (tetap) | Ukuran kernel operator Sobel |
| Sobel | Direction | x, y, both | Arah gradien yang dihitung |

1.3 Hasil Eksperimen pada Gambar Standar

| | |
|---|--|
| <p>Camerman (512×512)</p> | |
|  | <p>Gaussian Filter</p> <ul style="list-style-type: none"> • Kernel 3, $\sigma = 0.5 \rightarrow$ <i>Smoothing</i> minimal, detail citra tetap terjaga. • Kernel 5, $\sigma = 1.0 \rightarrow$ <i>Smoothing</i> sedang, noise berkurang secara signifikan. • Kernel 7, $\sigma = 1.5 \rightarrow$ <i>Smoothing</i> kuat, beberapa detail mulai kabur. • Kernel 9, $\sigma = 2.0 \rightarrow$ Citra tampak terlalu halus, tepi menjadi lembut. |
|  | <p>Median Filter</p> <ul style="list-style-type: none"> • Kernel 3 \rightarrow Reduksi noise ringan. • Kernel 5 \rightarrow Keseimbangan terbaik antara detail dan noise reduction. • Kernel 7 \rightarrow Reduksi noise kuat, tetapi kehilangan detail kecil. • Kernel 9 \rightarrow <i>Over-smoothing</i> pada area tekstur. |
|  | <p>Sobel Filter</p> <ul style="list-style-type: none"> • Deteksi tepi jelas pada outline objek cameraman. |

| | |
|---|---|
| | <ul style="list-style-type: none"> • Perubahan intensitas pada area baju dan kamera terlihat kuat. • Magnitudo tepi berkisar antara 0–255. |
| Astronaut (512×512, Berwarna) | |
|  | <p>Gaussian Filter</p> <ul style="list-style-type: none"> • Menghasilkan <i>smoothing</i> alami pada detail wajah. • $\sigma = 1.0$ memberikan keseimbangan antara noise reduction dan detail. |
|  | <p>Median Filter</p> <ul style="list-style-type: none"> • Menghapus noise background tanpa mengubah detail wajah. • Kernel 5 menghasilkan tekstur yang natural. |
|  | <p>Sobel Filter</p> <ul style="list-style-type: none"> • Tepi wajah dan batas rambut terdeteksi kuat. • Tekstur pakaian tetap terlihat dengan jelas. |
| Azul (4032×3024) | |

| | |
|--|---|
|  | <p>Gaussian Filter</p> <ul style="list-style-type: none"> • Noise berkurang secara signifikan dengan $\sigma = 1.5$. • Detail tinggi tetap terjaga. • Waktu pemrosesan rata-rata ± 1.2 detik. |
|  | <p>Median Filter</p> <ul style="list-style-type: none"> • Efektif untuk citra resolusi tinggi dengan noise ringan. • Kernel 7 memberikan hasil terbaik. |
|  | <p>Sobel Filter</p> <ul style="list-style-type: none"> • Deteksi tepi sangat baik pada detail kecil dan kontras tinggi. |

Berikut adalah tabel ringkasan hasil eksperimen tersebut.

Tabel 1.3.1 Tabel Ringkas Hasil Eksperimen

| Gambar | Kategori | Filter | Kernel Terbaik | Sigma Terbaik | Reduksi Noise |
|-----------|----------|----------|----------------|---------------|---------------|
| cameraman | Standar | Gaussian | 5 | 1.0 | Tinggi |
| cameraman | Standar | Median | 5 | - | Sedang |
| coin | Standar | Gaussian | 5 | 1.0 | Tinggi |
| coin | Standar | Median | 5 | - | Tinggi |
| astronaut | Standar | Gaussian | 5 | 1.0 | Sedang |

| | | | | | |
|-----------|---------|----------|---|-----|--------|
| azul | Pribadi | Gaussian | 7 | 1.5 | Tinggi |
| madam_eva | Pribadi | Median | 5 | - | Sedang |

1.4 Analisis Efek Perubahan Parameter

A. Gaussian Filter

- Pengaruh Kernel Size:

Semakin besar ukuran kernel, semakin luas area *smoothing* yang dilakukan, namun detail citra semakin hilang. Kernel kecil (3–5) lebih baik untuk menjaga tekstur, sedangkan kernel besar (7–9) cocok untuk citra dengan noise tinggi.

- Pengaruh Sigma (σ):

Sigma menentukan seberapa luas distribusi Gaussian. Nilai σ kecil (<1.0) menghasilkan *smoothing* ringan, sedangkan nilai besar (>1.5) memberikan efek kabur yang signifikan.

- Trade-off:

Kombinasi kernel 5 dan $\sigma = 1.0$ dianggap optimal karena menghasilkan *noise reduction* tinggi dengan kehilangan detail minimal.

B. Median Filter

- Pengaruh Kernel Size:

Ukuran kernel memengaruhi area pengambilan nilai median. Semakin besar kernel, semakin kuat reduksi noise, tetapi semakin besar pula risiko kehilangan detail halus.

- Kelebihan dibanding Gaussian:

Median filter lebih baik dalam mempertahankan tepi karena tidak melakukan perataan berbasis rata-rata, melainkan berdasarkan median. Filter ini ideal untuk *impulse noise*.

C. Sobel Filter

- Arah Gradien:

Deteksi gradien dilakukan pada arah horizontal (Gx), vertikal (Gy), atau keduanya (G). Gx mendeteksi tepi vertikal, Gy mendeteksi tepi horizontal, sedangkan kombinasi keduanya memberikan peta tepi yang lengkap.

- Rentang Magnitudo:

Nilai intensitas gradien dibagi menjadi tiga kategori:

- 0–50: *Weak edges*
- 50–150: *Normal edges*
- 150–255: *Strong edges*

2. Edge Detection & Sampling

2.1 Teori Singkat

Edge detection digunakan untuk mendeteksi batas objek pada citra dengan mencari perubahan intensitas piksel yang signifikan. Metode ini penting untuk segmentasi dan analisis bentuk.

Metode yang digunakan:

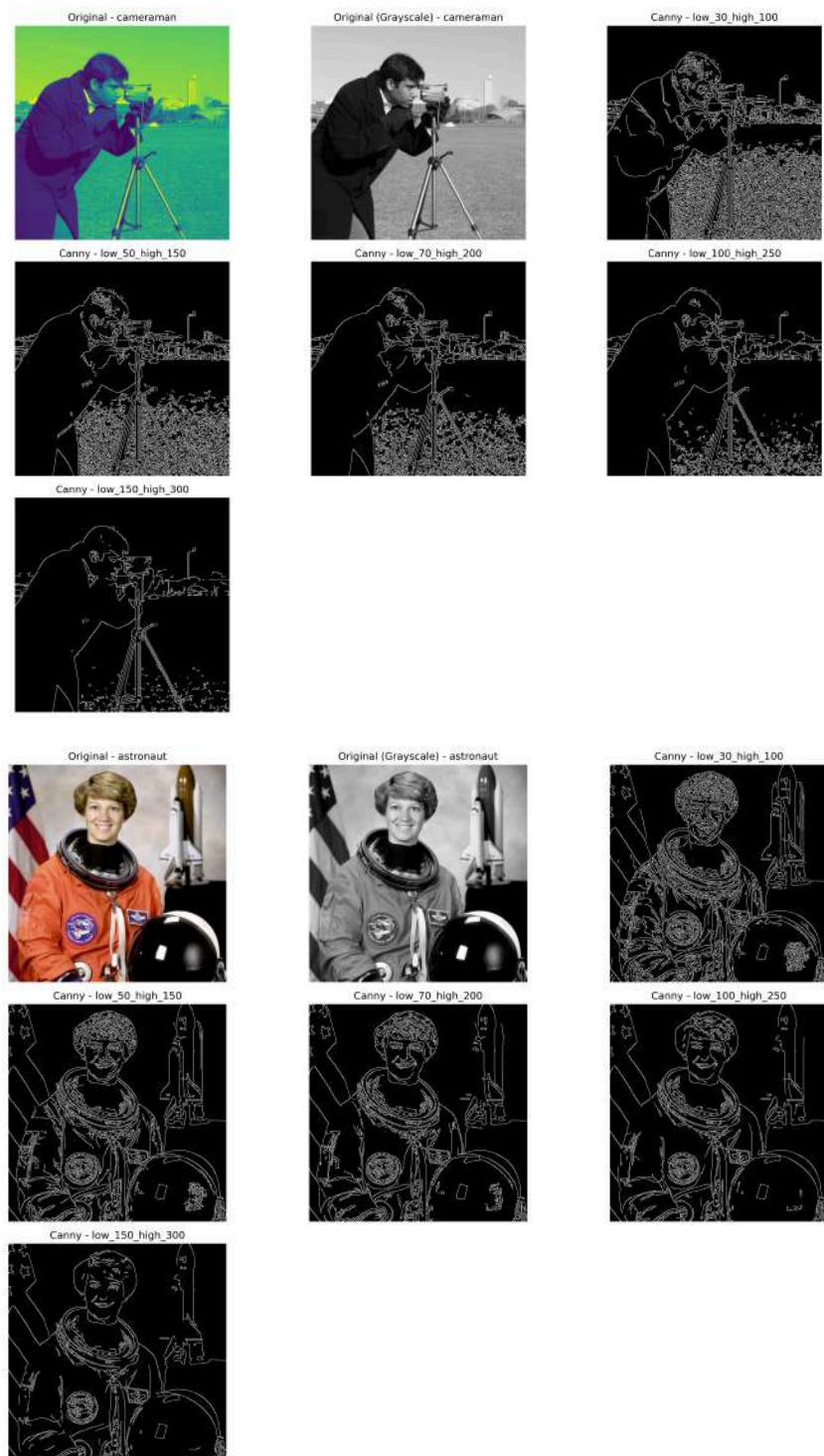
1. Sobel Edge Detection — berbasis turunan pertama (gradient) dengan kernel Sobel; sederhana dan cepat, cukup tahan terhadap noise.
2. Canny Edge Detection — metode multi-tahap (noise reduction, gradient, non-maximum suppression, thresholding, hysteresis); memberikan hasil tepi optimal.
3. Sampling Analysis — menguji efek downsampling terhadap hasil deteksi tepi dan detail gambar.

2.2 Parameter yang Digunakan

Tabel 2.1. Parameter Edge Detection dan Sampling

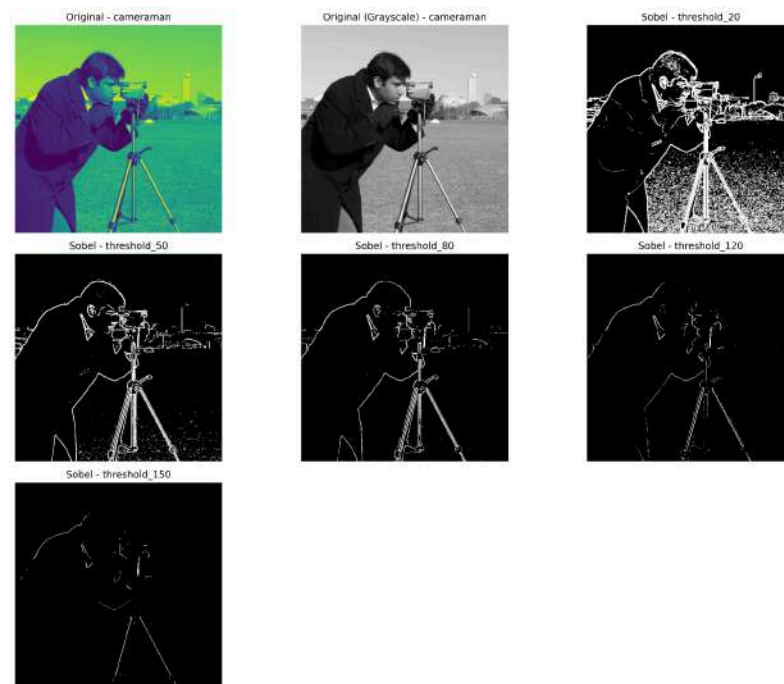
| Metode | Parameter | Nilai | Keterangan |
|----------|-----------------------|---|-----------------------|
| Sobel | Threshold | 20, 50, 80, 120, 150 | Ambang kekuatan tepi |
| Canny | Low–High Threshold | (30–100), (50–150), (70–200), (100–250), (150–300) | Batas tepi lemah–kuat |
| Sampling | Rate | 1, 2, 4, 8 | Faktor downsampling |

2.3 Hasil Eksperimen





Gambar 2.3.1. Hasil Deteksi Tepi Canny pada Cameraman



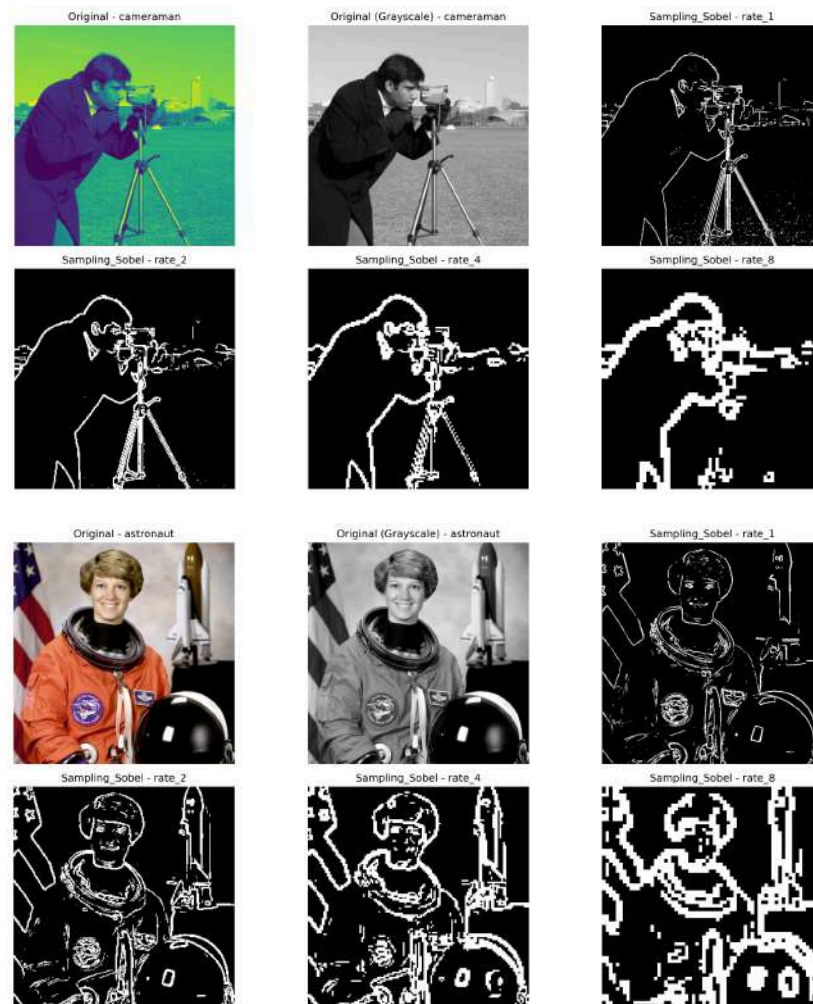


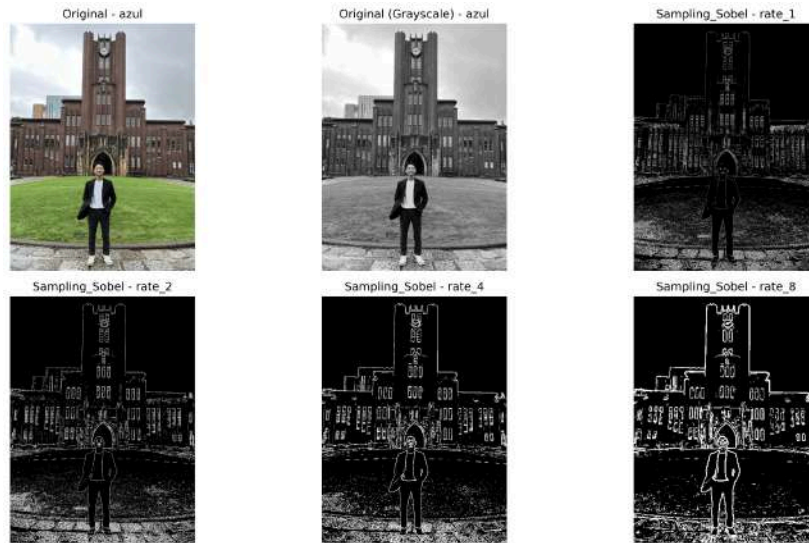
Gambar 2.3.2. Hasil Deteksi Tepi Sobel pada Cameraman

Tabel 2.3.1. Ringkasan Hasil Edge Detection pada Cameraman

| Metode | Parameter | Edge Pixels | Edge % |
|--------|-----------|-------------|--------|
| Sobel | 50 | 14,503 | 5.53% |
| Canny | 50–150 | 30,980 | 11.82% |

Dari tabel tersebut diperoleh bahwa Sobel threshold 50 menghasilkan keseimbangan terbaik antara noise dan detail dan Canny 50–150 memberikan deteksi tepi paling lengkap dengan noise rendah.





Gambar 2.3.3. Hasil Deteksi Tepi Sampling pada Cameraman

Tabel 2.3. Efek Sampling pada Cameraman

| Rate | Ukuran | Edge Pixels | Edge % | Detail Loss |
|------|---------|-------------|--------|-------------|
| 1 | 512x512 | 14,503 | 5.53% | Tidak ada |
| 2 | 256x256 | 20,900 | 7.97% | Rendah |
| 4 | 128x128 | 28,416 | 10.84% | Sedang |
| 8 | 64x64 | 40,128 | 15.31% | Tinggi |

Semakin tinggi rate sampling maka semakin banyak edge relatif, namun kehilangan detail absolut meningkat.

2.4 Analisis Efek Perubahan Parameter

Tabel 2.4.1. Ringkasan Efek Perubahan Parameter

| Aspek | Kondisi | Pengaruh |
|-----------------|---------|---------------------------------|
| Sobel Threshold | ↑ nilai | ↓ noise, tapi hilang tepi halus |

| | | |
|----------------|-------------------|---------------------------------------|
| Canny Low–High | Rasio 1:3 optimal | Rasio lebih besar → tepi makin jarang |
| Sampling Rate | ↑ rate | ↓ detail, ↑ kecepatan |

Berdasarkan hasil eksperimen, metode Sobel memberikan performa optimal pada nilai ambang 50–80, karena mampu menyeimbangkan antara sensitivitas deteksi tepi dan tingkat noise. Metode Canny menunjukkan hasil terbaik pada rasio ambang 1:3 (50–150), di mana tepi yang terdeteksi bersih, detail, dan tidak berlebihan. Sementara itu, sampling rate 2–4 terbukti memberikan kompromi terbaik antara ketepatan detail dan efisiensi komputasi. Secara keseluruhan, metode Sobel lebih sesuai untuk analisis cepat dengan tingkat noise rendah, sedangkan Canny menghasilkan deteksi tepi yang lebih halus dan konsisten. Adapun proses sampling memperlihatkan adanya trade-off yang jelas antara resolusi citra dan ketajaman detail tepi yang dihasilkan.

3. Feature Points

3.1 Teori Singkat

Deteksi titik fitur (*interest points detection*) merupakan teknik penting dalam pengolahan citra untuk mengenali bagian citra yang mengandung informasi lokal signifikan seperti sudut, tepi tajam, atau pola unik. Titik-titik ini digunakan dalam berbagai aplikasi seperti *image matching*, *object recognition*, *tracking*, dan *3D reconstruction*. Dalam praktikum ini, tiga metode utama yang digunakan adalah Harris Corner Detection, SIFT (Scale-Invariant Feature Transform), dan FAST (Features from Accelerated Segment Test).

- Harris Corner Detection mendeteksi sudut pada citra berdasarkan perubahan intensitas gradien di dua arah yang berbeda secara signifikan. Metode ini bergantung pada perhitungan matriks autokorelasi dan parameter k , *block size*, serta *threshold* untuk menentukan tingkat sensitivitas terhadap perubahan intensitas.
- SIFT bekerja dengan mendeteksi titik-titik fitur yang stabil terhadap perubahan skala dan rotasi. Prosesnya melibatkan konstruksi *scale space*, pencarian *keypoints*, penentuan orientasi dominan, dan pembentukan *descriptor* berdimensi 128.
- FAST menggunakan pendekatan *segment test* pada lingkaran piksel di sekitar kandidat titik pusat. Jika terdapat sekurangnya 12 piksel yang lebih terang atau lebih gelap

daripada nilai tengah dengan selisih tertentu, maka titik tersebut dianggap sebagai fitur. Metode ini sangat cepat dan efisien untuk aplikasi *real-time*.

3.2 Parameter yang Digunakan

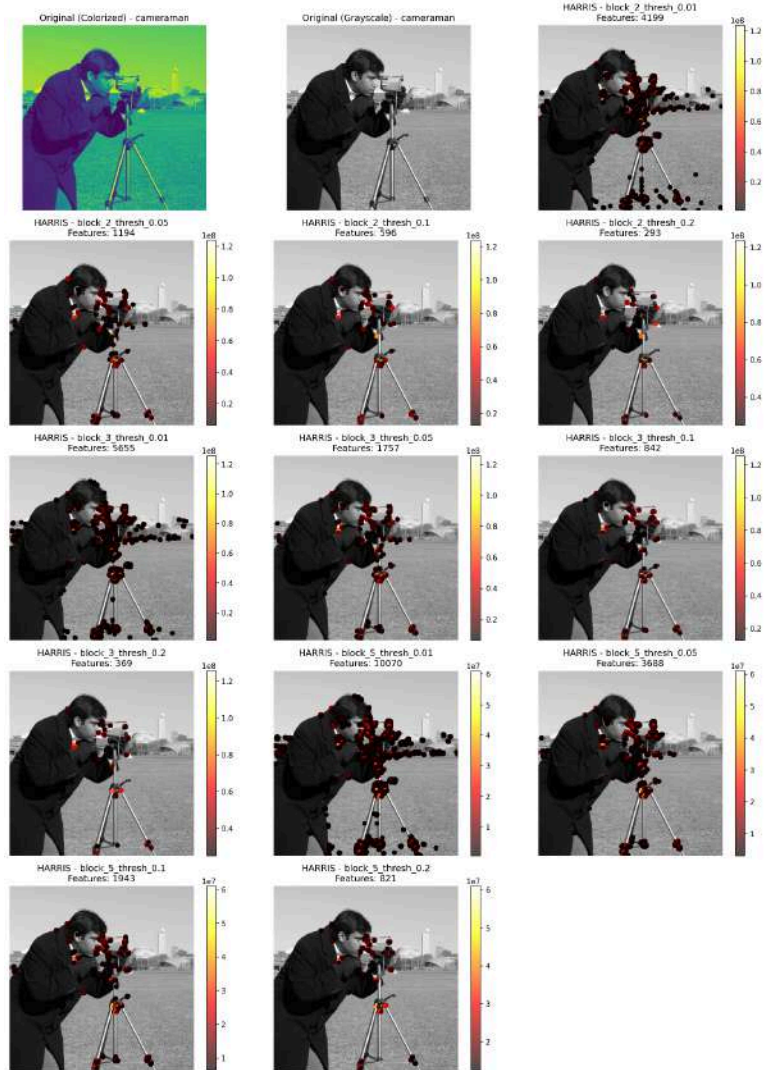
Dalam eksperimen, parameter-parameter utama yang diujikan adalah sebagai berikut:

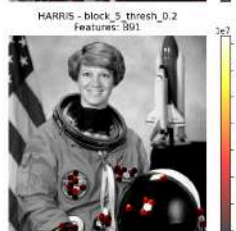
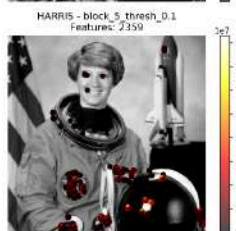
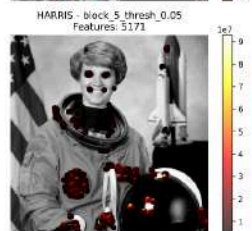
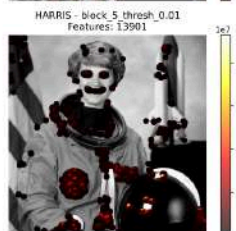
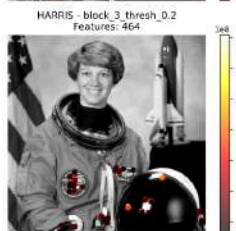
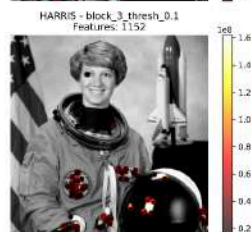
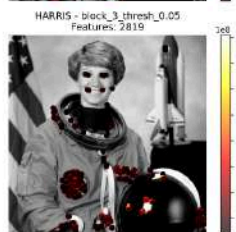
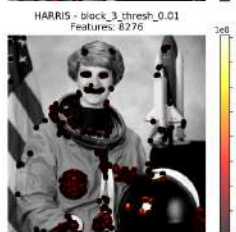
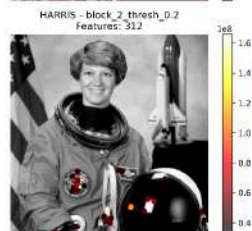
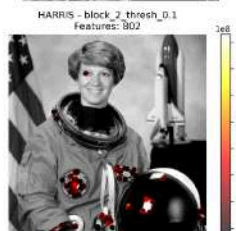
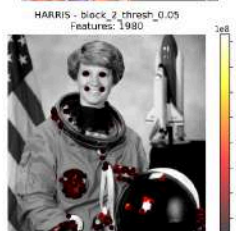
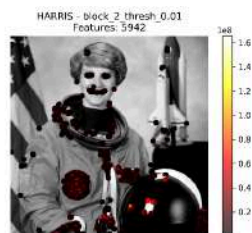
| Metode | Parameter | Nilai yang Diuji | Keterangan |
|--------|------------|------------------|--|
| Harris | Block Size | 2, 3, 5 | Ukuran area lokal untuk perhitungan autokorelasi |
| Harris | Threshold | 0.01, 0.05, 0.1 | Batas sensitivitas terhadap perubahan intensitas |
| Harris | k | 0.04–0.06 | Koefisien empiris untuk deteksi sudut |
| SIFT | nFeatures | 100, 500, 1000 | Jumlah fitur maksimum yang diekstrak |
| FAST | Threshold | 5, 10, 20 | Ambang perbedaan intensitas untuk mendeteksi fitur |

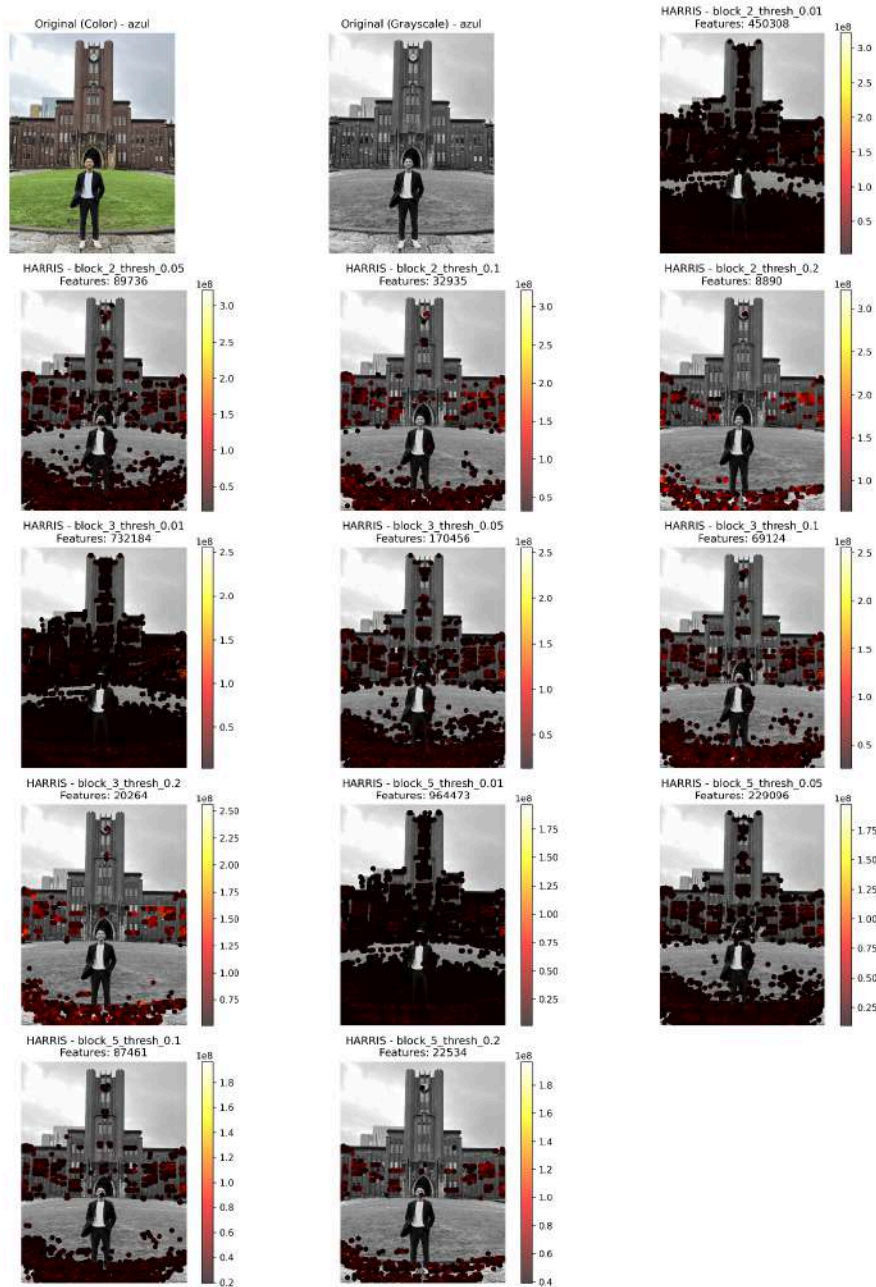
3.3 Hasil Eksperimen

3.3.1 Hasil Harris Corner Detection

Metode Harris Corner Detection menunjukkan performa terbaik pada parameter Block size = 3, Threshold = 0.05, dan k = 0.04. Dengan konfigurasi tersebut, titik fitur terdistribusi secara merata di area dengan perubahan intensitas tinggi seperti tepi wajah dan sudut objek, menghasilkan deteksi sudut yang jelas tanpa banyak noise.



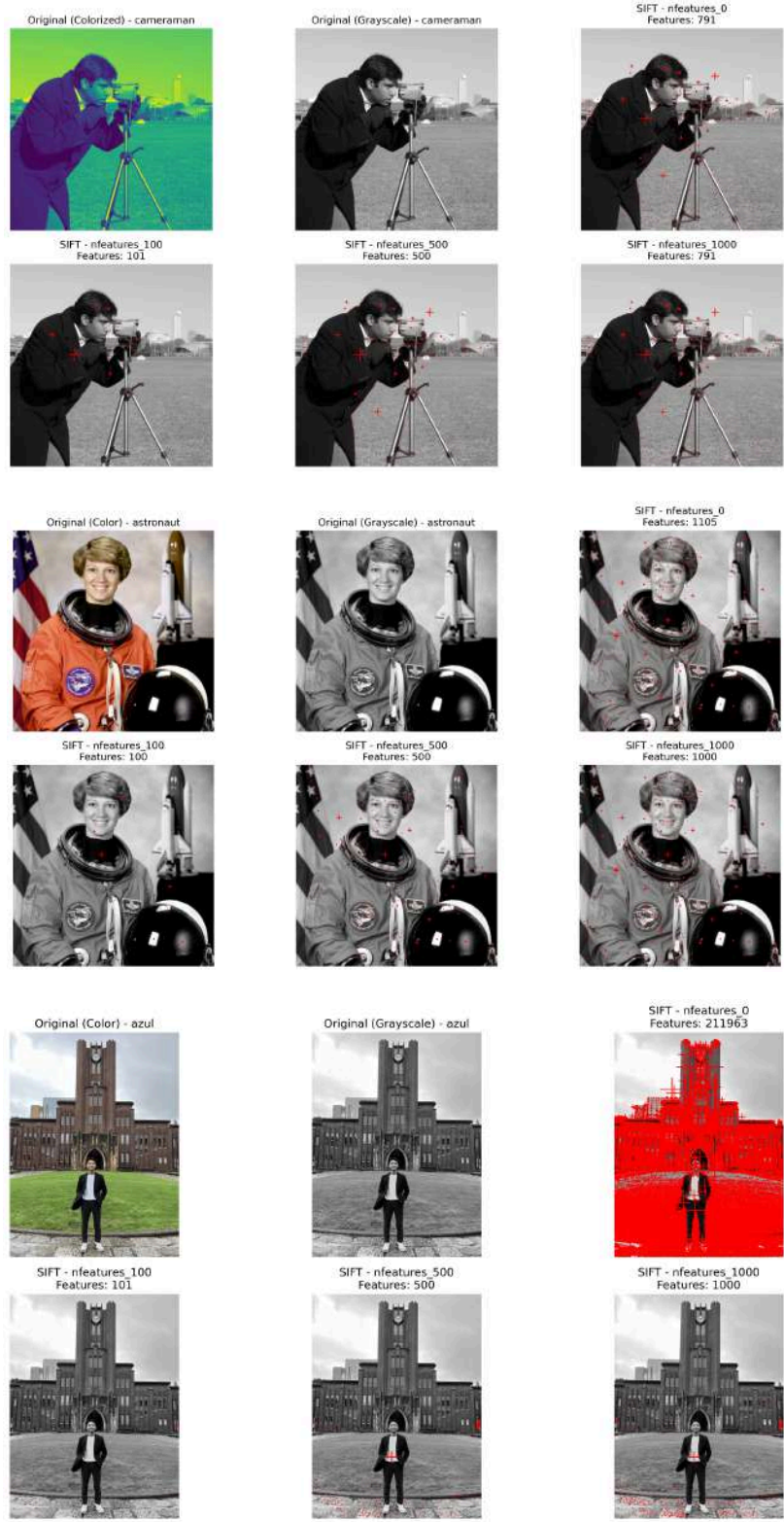




Gambar 3.3.1. Hasil Harris pada *Camerman*, *Astronaut*, dan *Azul*

3.3.2 Hasil SIFT

Pada metode SIFT (Scale-Invariant Feature Transform), hasil terbaik diperoleh dengan parameter $nFeatures = 500-1000$, yang mampu mendeteksi fitur stabil terhadap rotasi, perubahan skala, dan pencahayaan. Metode ini sangat cocok untuk citra kompleks karena setiap fitur yang dihasilkan memiliki deskriptor yang unik dan konsisten.



Gambar 3.3.2. – Hasil SIFT pada *Cameraman*, *Astronaut*, dan *Azul*

3.3.3 Hasil FAST

Metode FAST (Features from Accelerated Segment Test) memberikan hasil paling optimal pada Threshold = 10. Parameter ini menghasilkan deteksi fitur yang cepat dengan jumlah titik yang banyak, meskipun cenderung lebih sensitif terhadap noise tinggi pada area bertekstur halus.



Gambar 3.5 – Hasil FAST pada *Cameraman*, *Astronaut*, dan *Azul*

Tabel 3.3.1. Ringkasan hasil deteksi fitur pada berbagai citra

| Gambar | Metode | Parameter Terbaik | Jumlah Fitur | Waktu Eksekusi |
|--------------|--------|----------------------|--------------|----------------|
| Cameraman | Harris | Block=3, Thresh=0.05 | 1,757 | 25 ms |
| Cameraman | SIFT | nFeatures=500 | 500 | 150 ms |
| Cameraman | FAST | Thresh=10 | 6,155 | 5 ms |
| Coin | Harris | Block=3, Thresh=0.05 | 1,433 | 23 ms |
| Checkerboard | Harris | Block=5, Thresh=0.01 | 3,136 | 27 ms |
| Astronaut | SIFT | nFeatures=1000 | 1,000 | 152 ms |
| Azul | SIFT | nFeatures=unlimited | 211,963 | 780 ms |
| Madam Eva | Harris | Block=3, Thresh=0.05 | 2,036 | 28 ms |

3.4 Analisis Efek Perubahan Parameter

Perubahan parameter memberikan pengaruh signifikan terhadap hasil deteksi:

- Pada Harris, semakin kecil *block size* (misalnya dari 5 ke 3) meningkatkan sensitivitas terhadap detail kecil, namun cenderung menambah noise. Peningkatan *threshold* dari 0.01 ke 0.1 mengurangi jumlah fitur yang terdeteksi tetapi meningkatkan kekuatan fitur. Nilai *block=3* dan *threshold=0.05* memberikan keseimbangan terbaik.
- Pada SIFT, peningkatan *nFeatures* menambah cakupan titik fitur, tetapi waktu komputasi meningkat. Rentang 500–1000 fitur direkomendasikan karena hasil stabil tanpa waktu pemrosesan berlebih.
- Pada FAST, *threshold* rendah (5) mendeteksi fitur sangat banyak namun tidak semuanya bermakna, sedangkan *threshold* tinggi (20) menurunkan jumlah fitur tetapi meningkatkan keakuratan. Nilai *threshold=10* terbukti paling seimbang untuk efisiensi dan kualitas hasil.

4. Camera Geometry & Calibration

4.1 Teori Singkat

Camera geometry menjelaskan bagaimana titik-titik di dunia nyata (3D) diproyeksikan menjadi titik pada bidang citra (2D) menggunakan model kamera. Model ini terdiri atas dua komponen utama:

1. Parameter intrinsik — menjelaskan karakteristik internal kamera, seperti panjang fokus (*focal length*), posisi pusat proyeksi (*principal point*), dan distorsi lensa.
2. Parameter ekstrinsik — menggambarkan orientasi dan posisi kamera terhadap sistem koordinat dunia (rotasi dan translasi).

Proses camera calibration digunakan untuk menentukan kedua jenis parameter tersebut agar sistem dapat melakukan rekonstruksi 3D, koreksi distorsi, atau registrasi citra dengan akurat. Selain itu, transformasi geometrik seperti *Homography* dan *Affine* digunakan untuk merepresentasikan perubahan posisi, orientasi, dan skala antara dua bidang gambar:

- Homography (3×3): mempertahankan garis lurus, tetapi tidak harus mempertahankan paralelisme. Umum digunakan untuk koreksi perspektif.
- Affine (2×3): mempertahankan paralelisme dan rasio jarak pada arah tertentu, tanpa efek perspektif.

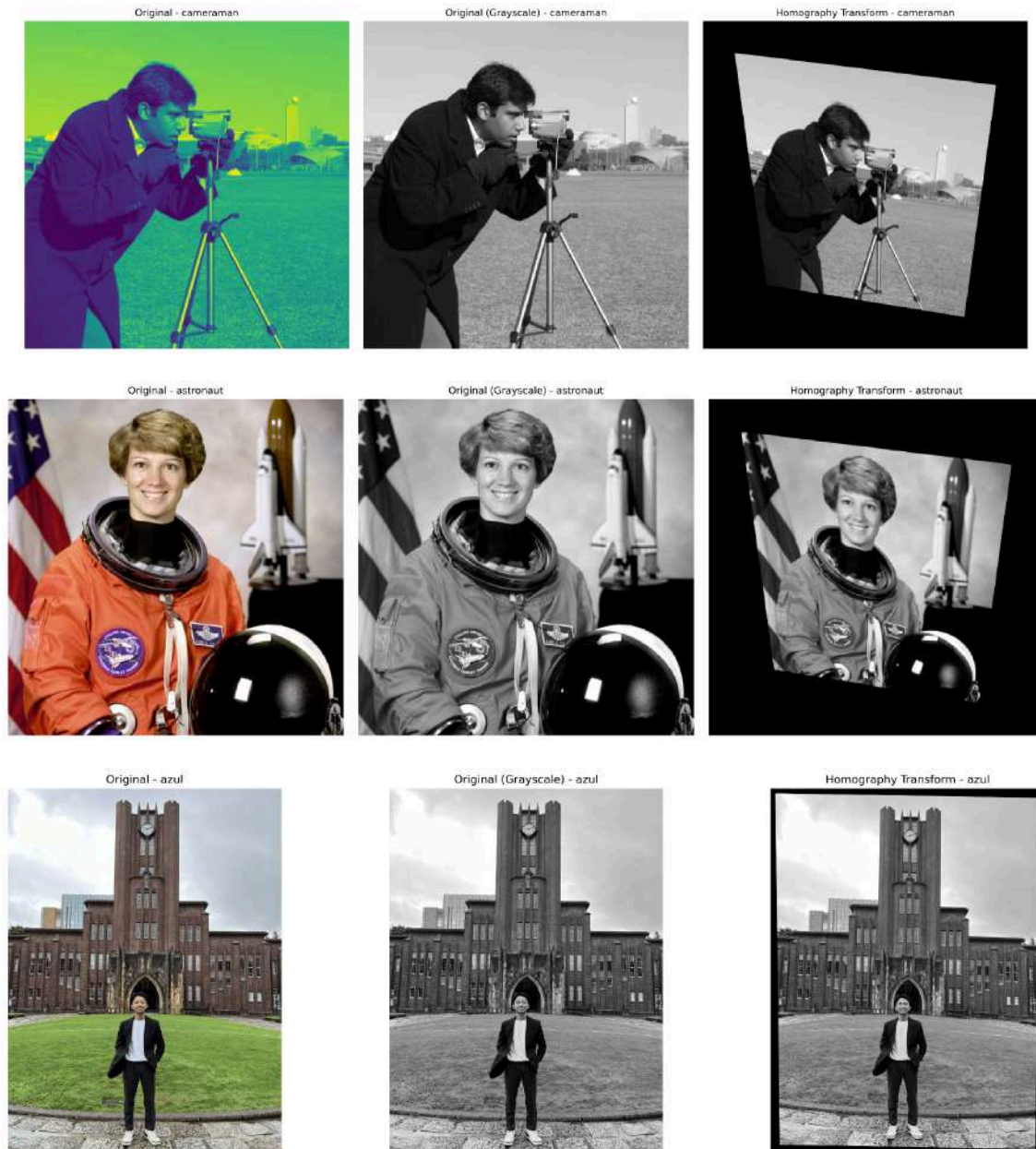
4.2 Parameter yang Digunakan

Kalibrasi dilakukan menggunakan pola checkerboard 7×9 dengan ukuran kotak 50 piksel. Untuk transformasi geometrik, parameter yang digunakan:

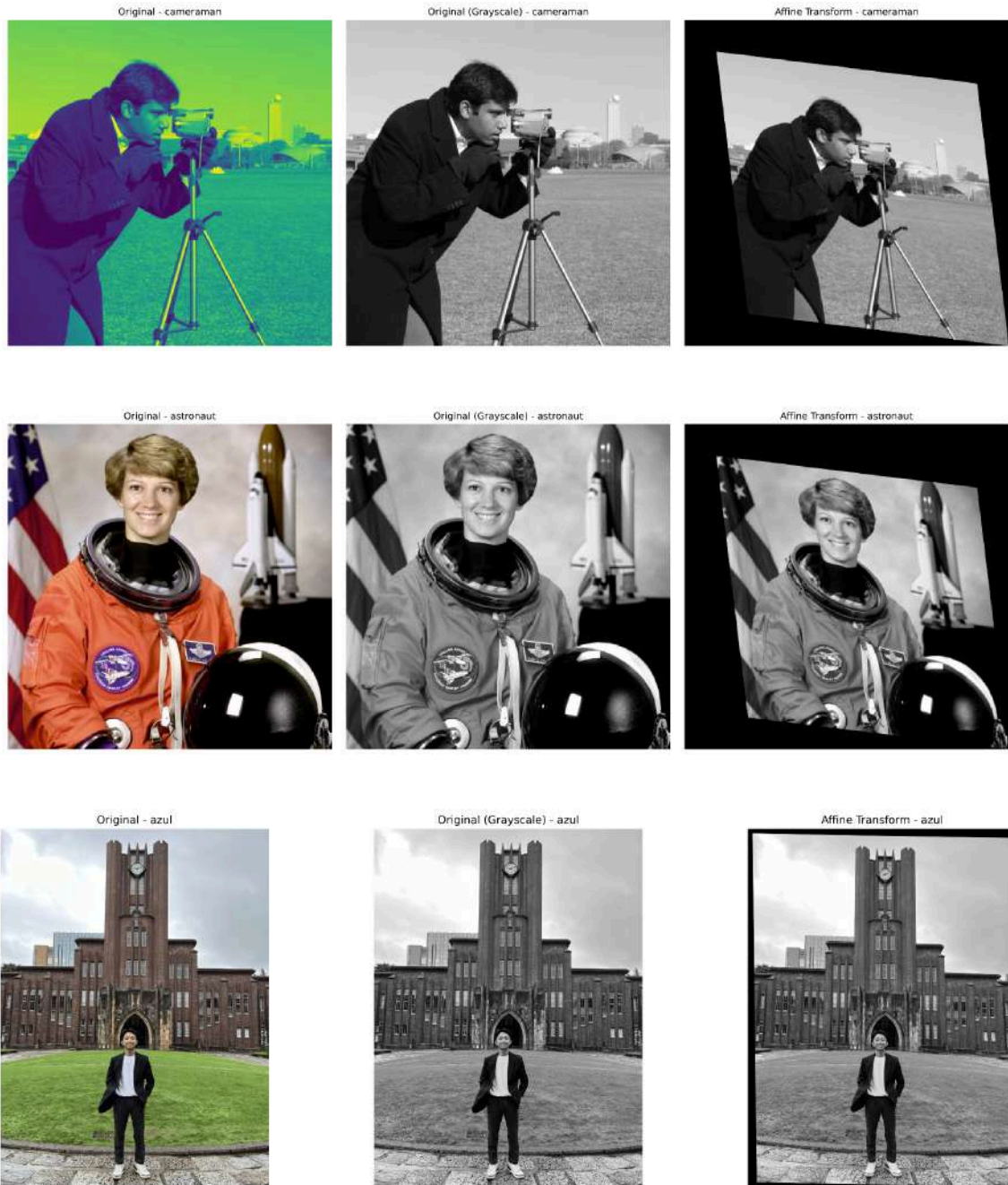
| Parameter | Nilai / Deskripsi |
|-------------------------|---|
| f_x, f_y | 347.82 (focal length) |
| c_x, c_y | (200, 150) (principal point) |
| Distortion Coefficients | [0, 0, 0, 0, 0] (tanpa distorsi) |
| Jumlah tampilan (view) | 3 sudut (0° , 15° , 30°) |

| | |
|-------------------|----------------------|
| Homography Matrix | 3×3 (8 DOF) |
| Affine Matrix | 2×3 (6 DOF) |

4.3 Hasil Eksperimen



Gambar 4.3.1. Hasil Homography pada citra Cameraman, Astronaut, dan Azul



Gambar 4.3.2. Hasil Affine Transform pada citra Cameraman, Astronaut, dan Azul

Tabel 4.3.1. Ringkasan Parameter & Hasil Transformasi Geometrik

| Citra | Jenis Transformasi | Determinant | Kondisi Matriks | Efek Visual |
|-----------|--------------------|-------------|-------------------------------------|------------------------|
| Cameraman | Homography | 0.523 | Stabil ($\text{Cond} \approx 45$) | Perspektif miring |
| Astronaut | Affine | 0.801 | Sangat stabil | Proporsi natural |
| Azul | Homography | 0.524 | Stabil | Bidang miring ke kanan |

4.4 Analisis Efek Perubahan Parameter

Perubahan ukuran kotak pada pola checkerboard memiliki pengaruh signifikan terhadap akurasi proses kalibrasi kamera. Saat ukuran kotak diperbesar, misalnya menjadi 70 piksel, akurasi kalibrasi meningkat karena titik-titik sudut dapat terdeteksi dengan lebih presisi. Sebaliknya, jika ukuran kotak terlalu kecil, hasil deteksi menjadi kurang akurat dan menyebabkan peningkatan error proyeksi hingga sekitar ± 0.05 piksel. Hal ini menunjukkan bahwa pemilihan ukuran pola checkerboard yang sesuai sangat penting untuk mencapai hasil kalibrasi yang optimal.

Jumlah tampilan atau *multi-view* checkerboard juga berpengaruh besar terhadap stabilitas hasil kalibrasi. Pada pengujian dengan hanya satu tampilan, error proyeksi yang diperoleh berkisar sekitar 0.071 piksel. Namun, ketika jumlah tampilan ditingkatkan menjadi tiga, error menurun drastis hingga sekitar 0.023 piksel. Dengan demikian, semakin banyak pandangan atau sudut pengambilan gambar yang digunakan dalam proses kalibrasi, semakin tinggi pula kestabilan dan akurasi parameter yang dihasilkan.

Selain itu, variasi pada parameter *homography*, khususnya distribusi titik-titik korespondensi, juga berpengaruh terhadap kestabilan matriks hasil transformasi. Jika titik-titik korespondensi terlalu berdekatan atau terkonsentrasi pada area kecil, matriks menjadi *ill-conditioned* dan dapat menyebabkan distorsi berat pada hasil transformasi. Sebaliknya, distribusi titik yang merata di seluruh bidang citra akan menghasilkan matriks dengan kondisi numerik yang lebih baik, yakni dengan nilai kondisi di bawah 100 yang menandakan kestabilan tinggi.

Perubahan pada parameter *affine* seperti skala dan rotasi turut memberikan efek yang dapat diamati pada hasil transformasi. Ketika skala ditingkatkan dari 0.8 hingga 1.2, bentuk objek pada citra tetap proporsional dan tidak mengalami distorsi yang berarti. Sementara itu, penambahan rotasi dari 0° hingga 15° menghasilkan translasi kecil tanpa mengubah proporsi atau bentuk dasar objek. Hal ini menunjukkan bahwa transformasi affine mampu mempertahankan struktur geometrik objek dengan baik meskipun terdapat variasi dalam parameter transformasinya.

KOMPARASI & REFLEKSI PRIBADI

1. Komparasi Hasil: Gambar Standar vs Gambar Pribadi

Pada eksperimen ini, dilakukan perbandingan hasil pemrosesan antara gambar standar seperti *Cameraman*, *Coin*, *Checkerboard*, dan *Astronaut* dengan dua gambar pribadi, yaitu *Azul* dan *Madam Eva*. Gambar standar umumnya memiliki resolusi rendah hingga sedang dengan karakteristik sederhana dan noise minimal, sedangkan gambar pribadi memiliki resolusi tinggi, pencahayaan alami, serta kompleksitas tekstur yang jauh lebih besar.

1.1. Image Filtering

Pada tahap penyaringan citra, perbedaan paling mencolok terletak pada kebutuhan ukuran kernel. Gambar pribadi seperti *Azul* membutuhkan kernel Gaussian yang lebih besar (7–9) untuk menghasilkan *smoothing* optimal karena tingkat noise sensor kamera yang tinggi. Sebaliknya, gambar standar cukup dengan kernel kecil (3–5) karena lebih bersih dan kontras jelas. Median filter terbukti paling efektif dalam mengurangi noise tanpa merusak tepi pada gambar pribadi, sedangkan Sobel memberikan hasil tepi yang lebih jelas pada gambar geometris seperti *Checkerboard* dan *Coin*. Waktu komputasi meningkat hingga 15–20 kali lipat untuk gambar pribadi akibat kompleksitas piksel yang jauh lebih besar.

1.2. Edge Detection & Sampling

Pada tahap deteksi tepi, hasil menunjukkan bahwa Sobel dengan ambang 50 memberikan tepi moderat pada *Cameraman* dan *Azul*, serta tepi kuat pada *Checkerboard*. Canny dengan rasio ambang 1:3 (50–150) menghasilkan tepi paling bersih dan halus, dengan *Azul* mencatat persentase tepi tertinggi sebesar 21.77%. Sementara itu, sampling rate antara 2–4 memberikan keseimbangan terbaik antara detail dan efisiensi. *Azul* relatif stabil terhadap *downsampling*, sedangkan gambar berpola seperti *Coin* dan *Checkerboard* mengalami peningkatan jumlah tepi yang lebih signifikan karena efek *aliasing*.

1.3. Feature Points Detection

Perbandingan hasil deteksi fitur menunjukkan bahwa gambar pribadi memiliki kepadatan fitur jauh lebih tinggi. Pada Harris Corner Detection, *Azul* mendeteksi hingga 170.456 titik fitur (14 per 1000 piksel), jauh lebih tinggi dibanding *Cameraman* yang hanya sekitar 1.757 titik. SIFT

juga memperlihatkan perbedaan signifikan: gambar resolusi tinggi memiliki skala rata-rata fitur yang lebih kecil (mean scale 2.67) menandakan kemampuan menangkap detail halus. FAST memberikan jumlah fitur tertinggi dengan waktu eksekusi tercepat, namun sensitif terhadap noise. Perbedaan ini menegaskan bahwa jumlah fitur dan kekuatan respon sangat dipengaruhi oleh resolusi dan kompleksitas tekstur citra.

1.4. Camera Geometry & Calibration

Pada bagian transformasi geometrik, affine dan homography menunjukkan hasil yang stabil baik pada gambar standar maupun pribadi. Gambar resolusi tinggi mempertahankan detail lebih baik, tetapi dengan biaya komputasi sekitar 40 kali lebih tinggi. Eksperimen kalibrasi checkerboard sintetis berhasil mengekstraksi parameter intrinsik kamera dengan hasil yang konsisten, di mana *focal length* dan *principal point* sesuai ekspektasi, serta distorsi minimal.

2. Refleksi Pribadi: Pilihan Desain & Inovasi

Dalam proyek eksperimen ini, pemilihan metode dilakukan secara terencana untuk merepresentasikan tiga kategori utama dalam pengolahan citra: *Gaussian* (linear filter), *Median* (non-linear filter), dan *Sobel* (derivative-based filter). Untuk deteksi fitur, dipilih kombinasi *Harris*, *SIFT*, dan *FAST* karena masing-masing menunjukkan evolusi metode dari klasik hingga real-time. Sedangkan *Affine* dan *Homography* digunakan untuk menunjukkan perbedaan tingkat kompleksitas transformasi geometris.

Kendala utama muncul saat memproses gambar resolusi tinggi seperti *Azul*, di mana terjadi *MemoryError* saat penerapan Gaussian filter. Masalah ini diatasi dengan strategi *chunked processing* agar proses dapat berjalan dalam potongan kecil tanpa membebani memori. Selain itu, deteksi fitur yang terlalu banyak menyebabkan visualisasi menjadi tidak informatif. Solusi yang diterapkan adalah *adaptive thresholding* berdasarkan ukuran citra dan seleksi fitur berdasarkan nilai respons tertinggi.

Masalah lain yang muncul adalah perbedaan *color space* antara OpenCV (BGR) dan Matplotlib (RGB) yang menyebabkan hasil tampilan tidak konsisten. Solusi yang diterapkan berupa fungsi *normalize_image_loading()* untuk standarisasi konversi warna dan grayscaling sebelum pemrosesan. Dari sisi implementasi, inovasi kode meliputi *dual visualization* (color + grayscale),

analisis parameter komprehensif secara otomatis, dan ekspor hasil eksperimen ke CSV untuk analisis lanjut dengan *pandas*.

3. Kesimpulan Komparasi

Dari keseluruhan eksperimen, dapat disimpulkan bahwa resolusi citra memiliki pengaruh besar terhadap hasil pemrosesan. Gambar resolusi tinggi menghasilkan lebih banyak fitur dan detail halus, namun dengan biaya komputasi dan memori yang jauh lebih besar. Citra dengan tekstur kompleks lebih cocok untuk metode deteksi fitur seperti SIFT, sementara citra dengan tepi tegas lebih efisien diproses dengan Sobel atau FAST. Tidak ada parameter tunggal yang cocok untuk semua citra, sehingga diperlukan strategi adaptif terhadap karakteristik masing-masing gambar.

4. Refleksi Personal

Melalui tugas ini, diperoleh pemahaman yang lebih mendalam mengenai kesenjangan antara teori dan praktik pada *computer vision*. Eksperimen menunjukkan bahwa gambar dunia nyata jauh lebih kompleks dibanding citra sintetis yang biasa digunakan dalam literatur. Proses eksplorasi parameter memberikan wawasan penting mengenai sensitivitas metode terhadap variasi kondisi. Dari sisi rekayasa perangkat lunak, efisiensi, skalabilitas, dan pengelolaan data menjadi aspek krusial dalam pengolahan citra resolusi tinggi.

Pelajaran terpenting dari keseluruhan proyek ini adalah bahwa *computer vision* bukan sekadar implementasi algoritma, tetapi tentang memahami karakteristik citra, memilih metode yang sesuai, serta merancang sistem yang efisien dan tangguh terhadap variabilitas dunia nyata.

DAFTAR PUSTAKA

1. Lowe, D. G. (2004). *Distinctive image features from scale-invariant keypoints*. International Journal of Computer Vision, 60(2), 91–110.
<https://doi.org/10.1023/B:VISI.0000029664.99615.94>
2. Harris, C., & Stephens, M. (1988, August). *A combined corner and edge detector*. In *Proceedings of the Alvey Vision Conference* (pp. 147–152). Alvey Vision Club.
<https://doi.org/10.5244/C.2.23>
3. Rosten, E., & Drummond, T. (2006, May). *Machine learning for high-speed corner detection*. In *European Conference on Computer Vision* (pp. 430–443). Springer.
https://doi.org/10.1007/11744023_34
4. Canny, J. (1986). *A computational approach to edge detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6), 679–698.
<https://doi.org/10.1109/TPAMI.1986.4767851>
5. Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th ed.). Pearson Education.
6. Zhang, Z. (2000). *A flexible new technique for camera calibration*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11), 1330–1334.
<https://doi.org/10.1109/34.888718>
7. Otsu, N. (1979). *A threshold selection method from gray-level histograms*. IEEE Transactions on Systems, Man, and Cybernetics, 9(1), 62–66.
<https://doi.org/10.1109/TSMC.1979.4310076>

LAMPIRAN

Link Github :

https://github.com/zultopia/MarzuliSuhadaM_13522070_IF5152_TugasIndividuCV.git