**UTS DATA MINING**

**PREPROCESSING GAMBAR**

Nama : Syahril Akbar

Nim : 105841103821

Kelas : 6B Informatika

Program Python Preprocessing Gambar

import cv2

import numpy as np

*# Baca gambar dari file*

gambar = cv2.imread('gambar\_awal.jpg', cv2.**IMREAD\_COLOR**)

*# 1. Resizing (Mengubah Ukuran)*

ukuran\_baru = (200, 200)

gambar\_resized = cv2.resize(gambar, ukuran\_baru)

*# 2. Grayscaling (Konversi ke Grayscale)*

gambar\_gray = cv2.cvtColor(gambar, cv2.**COLOR\_BGR2GRAY**)

*# 3. Noise Reduction (Pengurangan Noise)*

gambar\_blur = cv2.GaussianBlur(gambar\_gray, (5, 5), 0)

*# 4. Normalization (Normalisasi)*

gambar\_normalized = cv2.normalize(gambar\_blur, None, 0, 255, cv2.**NORM\_MINMAX**)

*# 5. Binarization (Konversi ke Hitam-Putih)*

\_, gambar\_binary = cv2.threshold(gambar\_normalized, 128, 255, cv2.**THRESH\_BINARY**)

*# 6. Contrast Enhancement (Peningkatan Kontras)*

gambar\_equalized = cv2.equalizeHist(gambar\_gray)

*# Simpan gambar hasil preprocessing*

cv2.imwrite('gambar\_resized.jpg', gambar\_resized)

cv2.imwrite('gambar\_gray.jpg', gambar\_gray)

cv2.imwrite('gambar\_blur.jpg', gambar\_blur)

cv2.imwrite('gambar\_normalized.jpg', gambar\_normalized)

cv2.imwrite('gambar\_binary.jpg', gambar\_binary)

cv2.imwrite('gambar\_equalized.jpg', gambar\_equalized)

print("Preprocessing selesai! Hasil tersimpan dalam file gambar\_resized.jpg, gambar\_gray.jpg, gambar\_blur.jpg, gambar\_normalized.jpg, gambar\_binary.jpg, dan gambar\_equalized.jpg.")

Program Python yang diatas menggunakan OpenCV untuk melakukan preprocessing pada gambar. Berikut penjelasan langkah demi langkahnya:

*Catatan: klik gambar screensoot untuk melihat hasilnya*

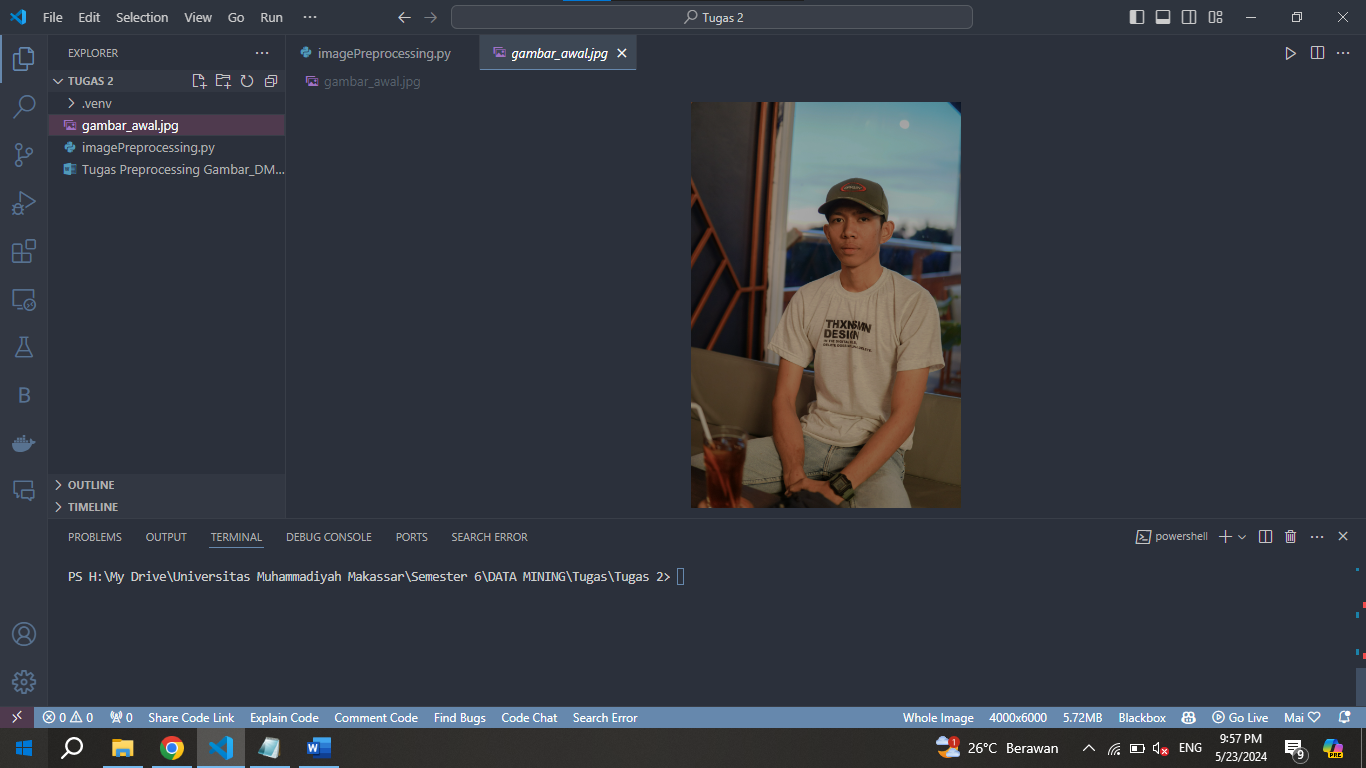
1. Membaca Gambar

`cv2.imread('gambar\_awal.jpg', cv2.IMREAD\_COLOR)`: Membaca gambar dari file `gambar\_awal.jpg` dengan mode warna `cv2.IMREAD\_COLOR`. Hal ini menghasilkan array NumPy yang merepresentasikan piksel gambar.

1. Resizing (Mengubah Ukuran)

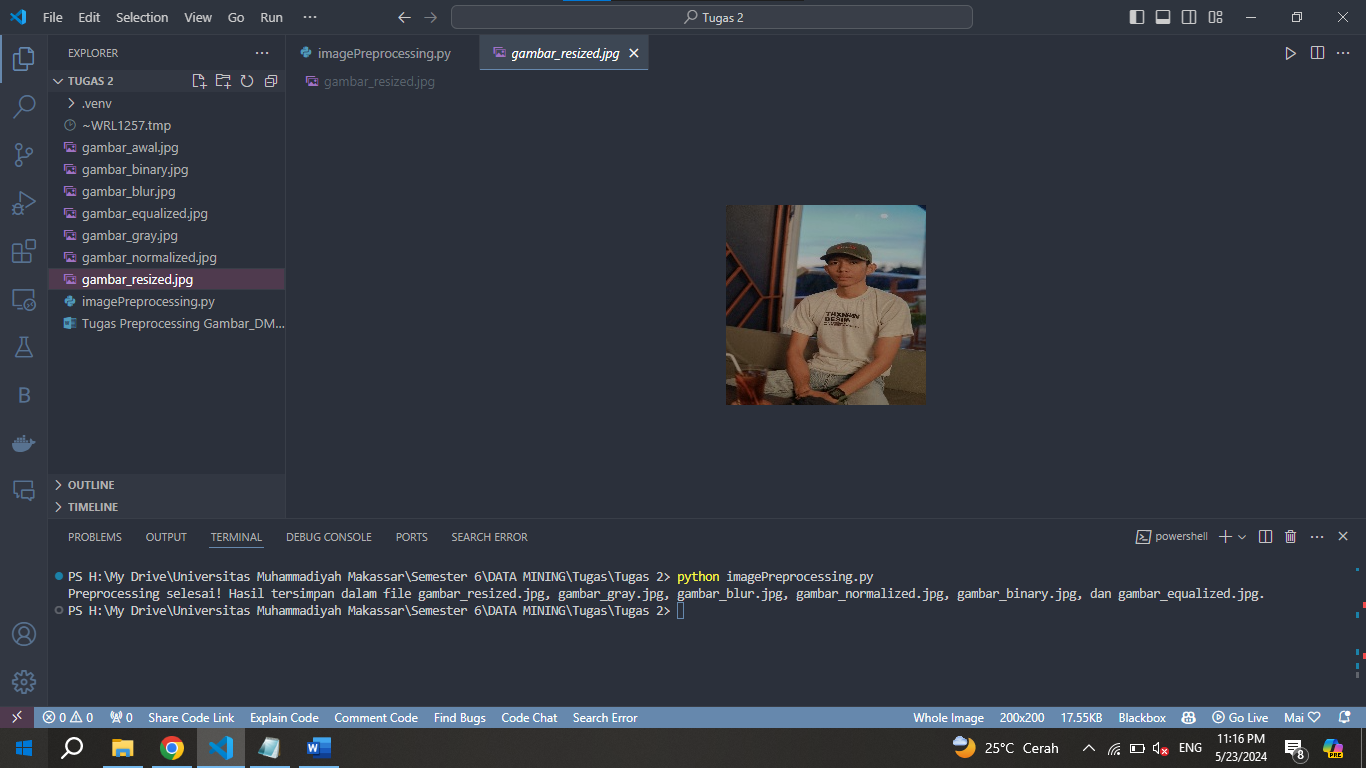
`cv2.resize(gambar, (200, 200))`: Mengubah ukuran gambar menjadi 200x200 piksel.

Data sebelum:

[](https://drive.google.com/open?id=1PbVsZ8Ej-jBOd4vfL5kr3H5l1ndAnNYZ&usp=drive_fs)

Ukuran gambar asli (4000 x 6000 piksel).

Data setelah:

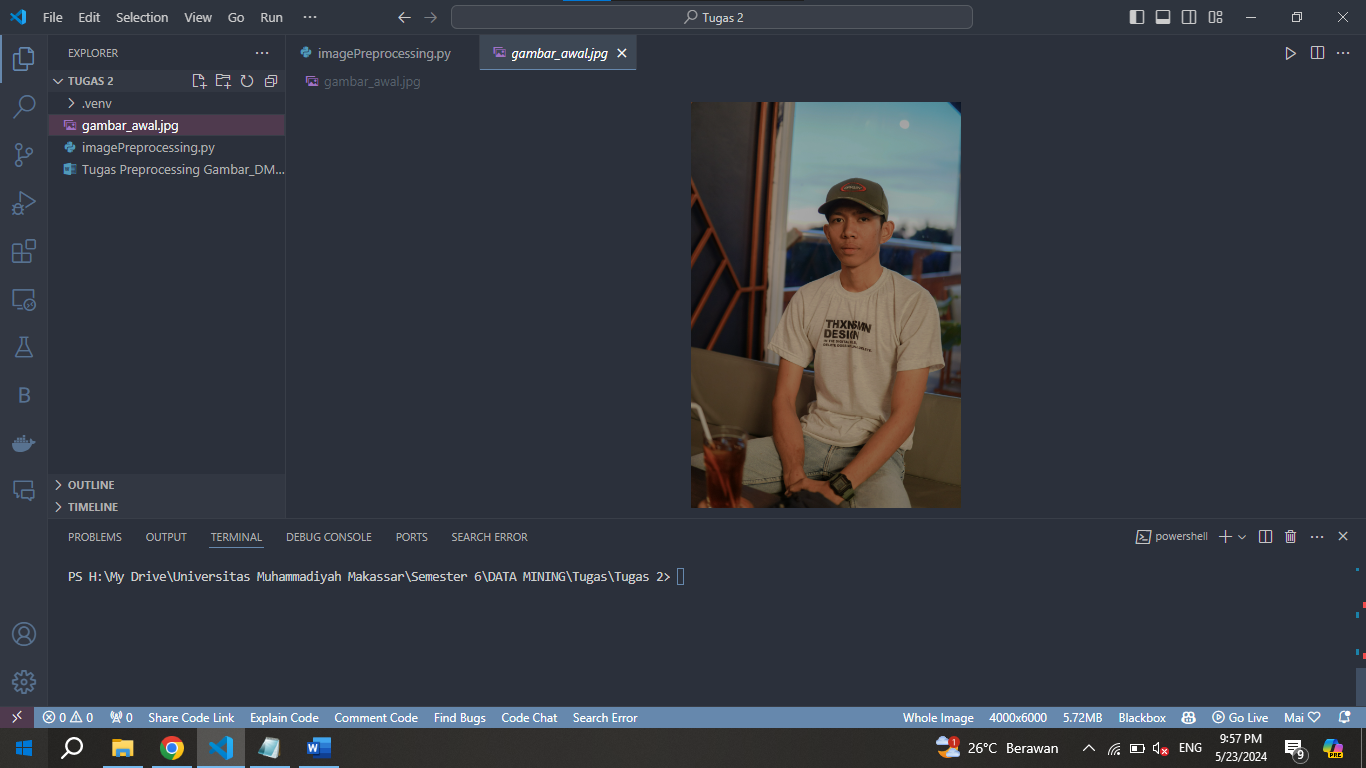
[](https://drive.google.com/open?id=1Q-y6oMFZdN5FiFUE-xj8xN4eisq7zIY-&usp=drive_fs)

Ukuran gambar baru 200x200 piksel.

1. Grayscaling (Konversi ke Grayscale)

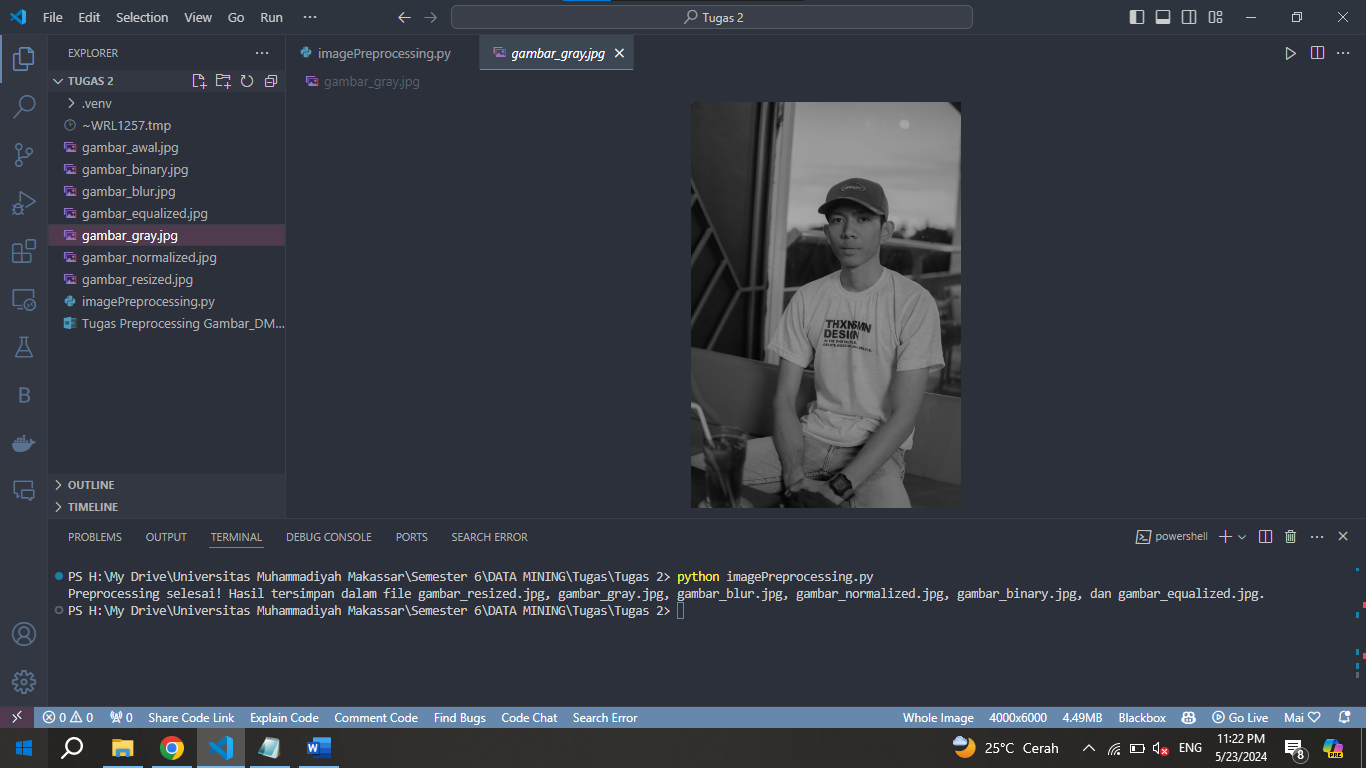
`cv2.cvtColor(gambar, cv2.COLOR\_BGR2GRAY)`: Mengubah gambar dari mode warna BGR (Blue-Green-Red) menjadi grayscale (hitam putih).

Data sebelum:

[](https://drive.google.com/open?id=1PbVsZ8Ej-jBOd4vfL5kr3H5l1ndAnNYZ&usp=drive_fs)

Nilai piksel merepresentasikan warna BGR (masing-masing 0-255).

Data setelah

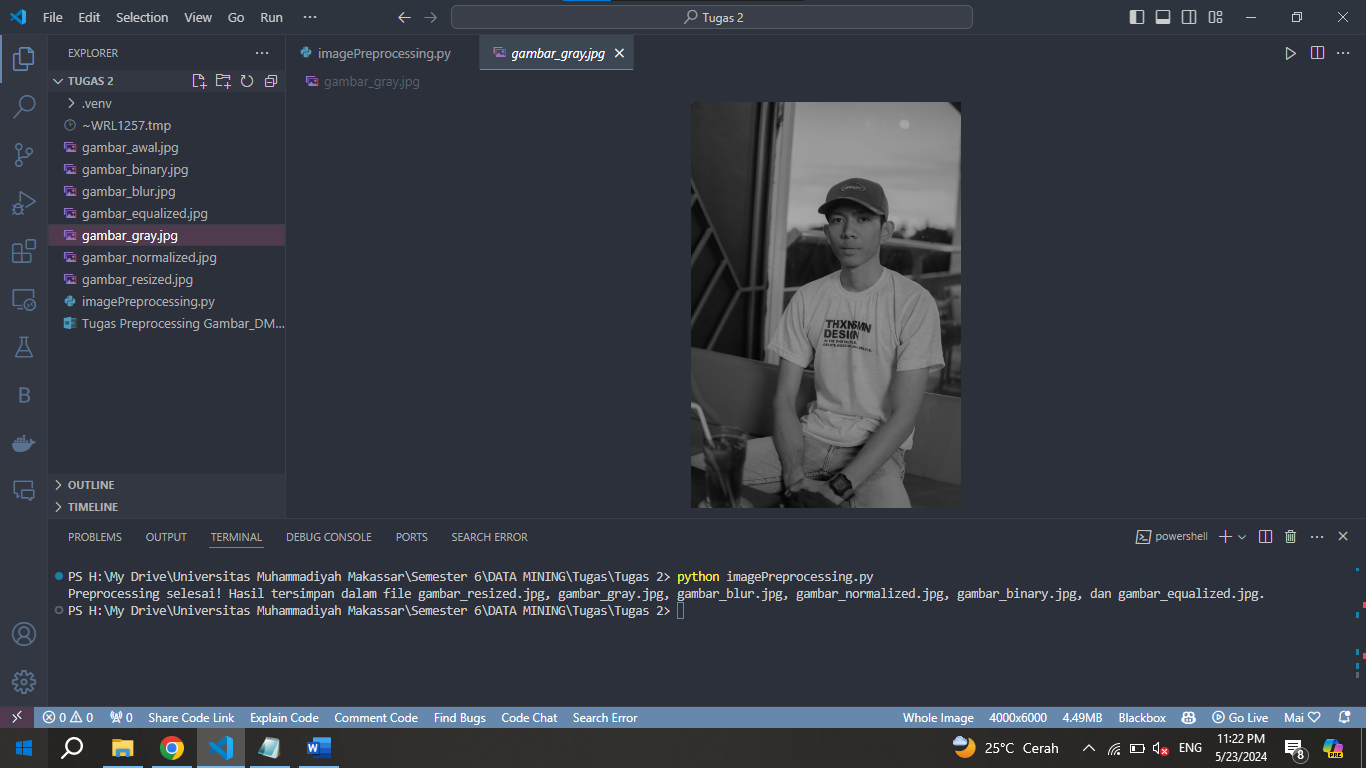
[](https://drive.google.com/open?id=1Q-_tKfJu7wUWeMY-LqClFFx4sr0MDpmT&usp=drive_fs)

Nilai piksel merepresentasikan intensitas cahaya (0-255, hitam-putih).

1. Noise Reduction (Pengurangan Noise):

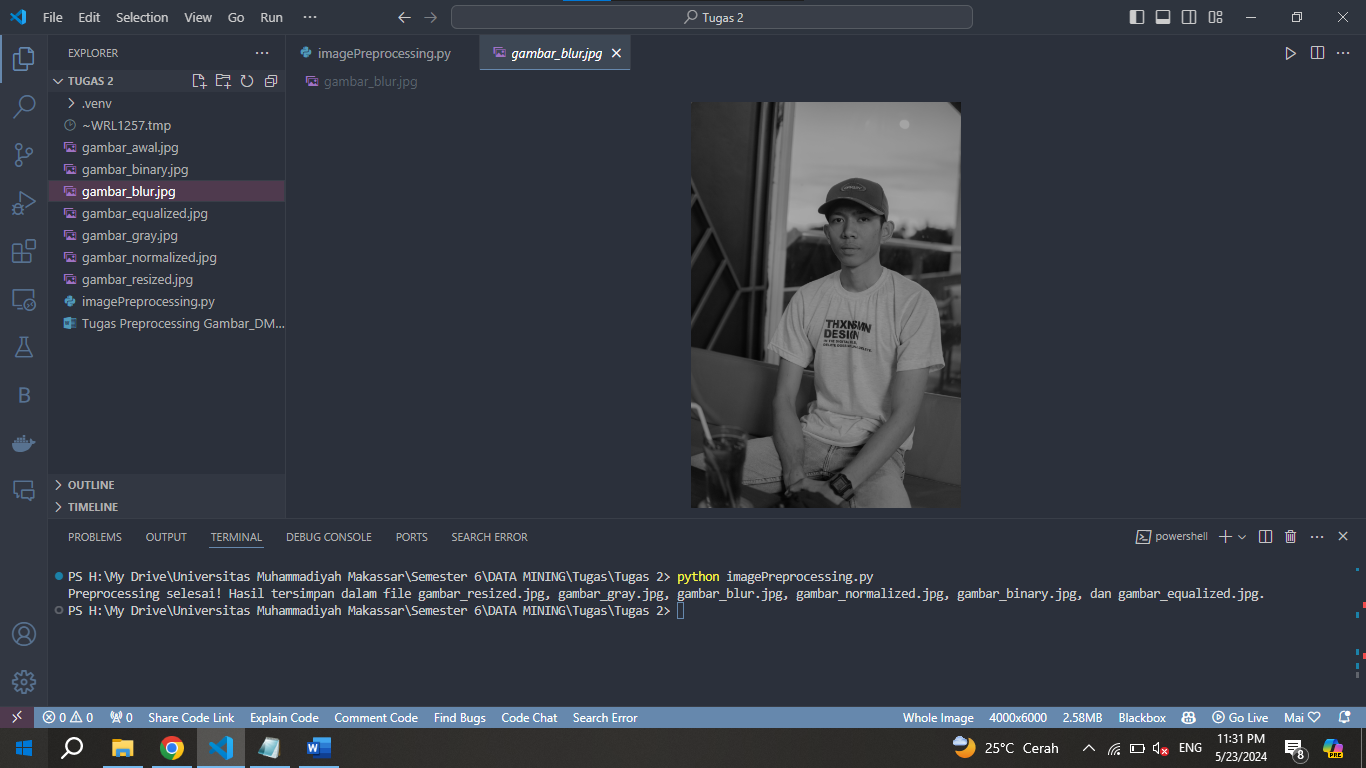
`cv2.GaussianBlur(gambar\_gray, (5, 5), 0)`: Mengurangi noise (bintik-bintik acak) pada gambar grayscale dengan filter Gaussian Blur berukuran kernel 5x5.

Data sebelum:

[](https://drive.google.com/open?id=1Q-_tKfJu7wUWeMY-LqClFFx4sr0MDpmT&usp=drive_fs)

Gambar grayscale dengan noise.

Data setelah:

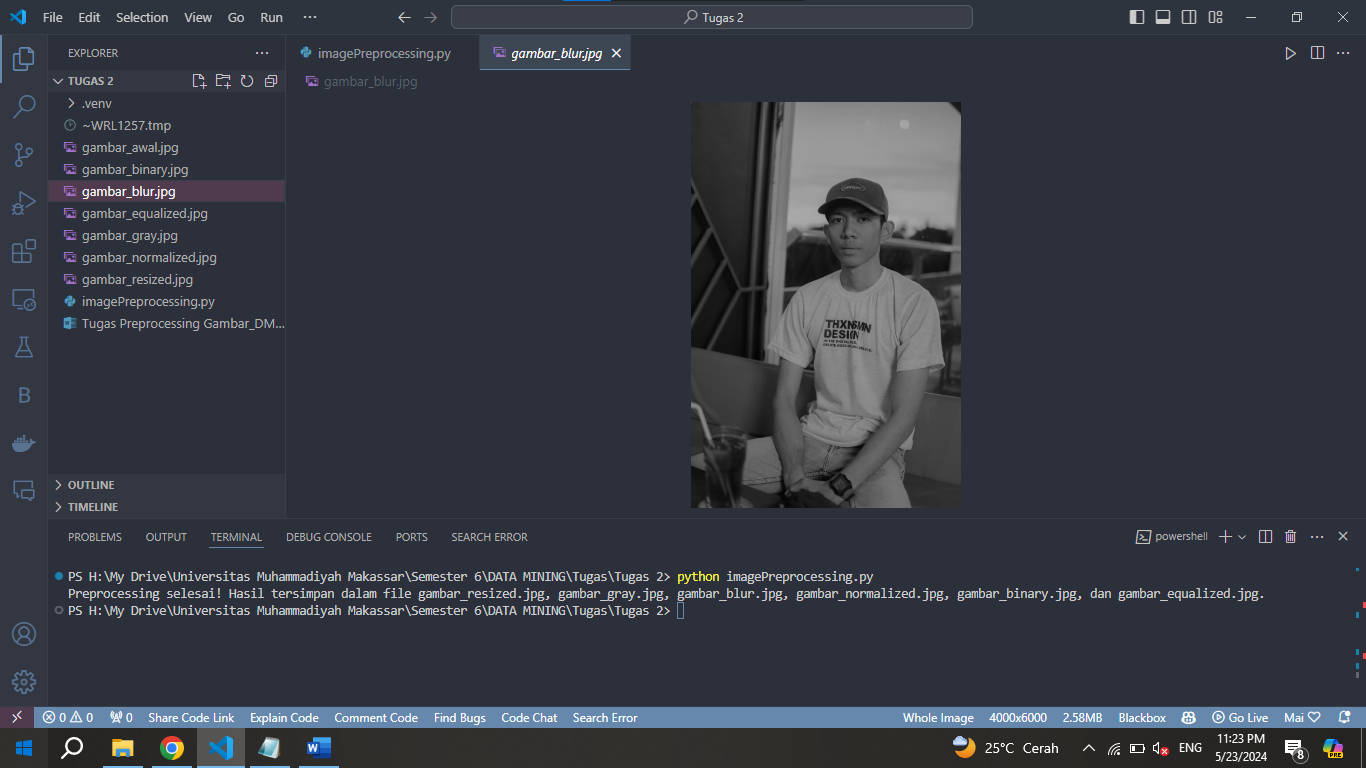
[](https://drive.google.com/open?id=1PyZ9VsAv0D3YA_4LsDM8C3qjfdw41KX1&usp=drive_fs)

Gambar grayscale yang lebih halus dan bebas noise.

1. Normalization (Normalisasi)

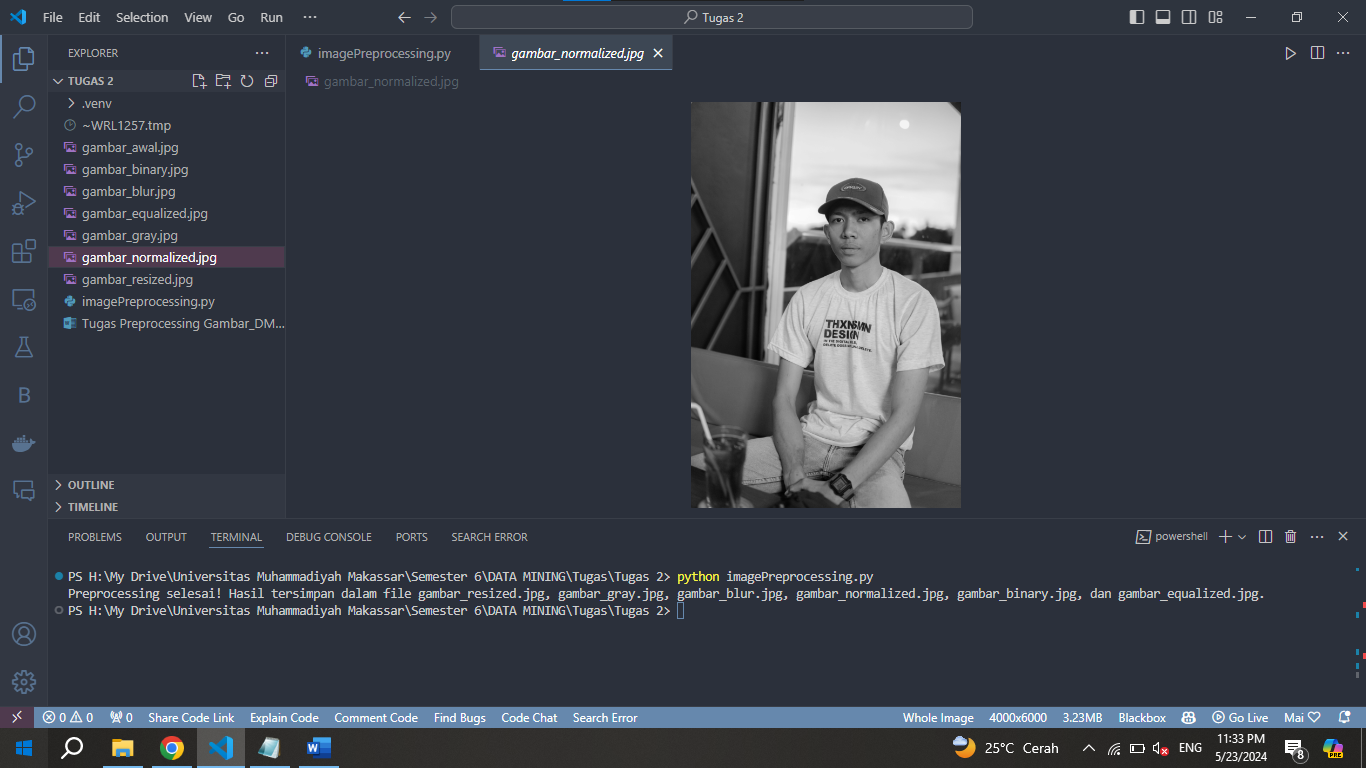
`cv2.normalize(gambar\_blur, None, 0, 255, cv2.NORM\_MINMAX)`: Menormalkan nilai piksel gambar blur antara 0 dan 255.

Gambar sebelum:

[](https://drive.google.com/open?id=1PyZ9VsAv0D3YA_4LsDM8C3qjfdw41KX1&usp=drive_fs)

Nilai piksel gambar blur mungkin tidak dalam rentang 0-255.

Gambar setelah:

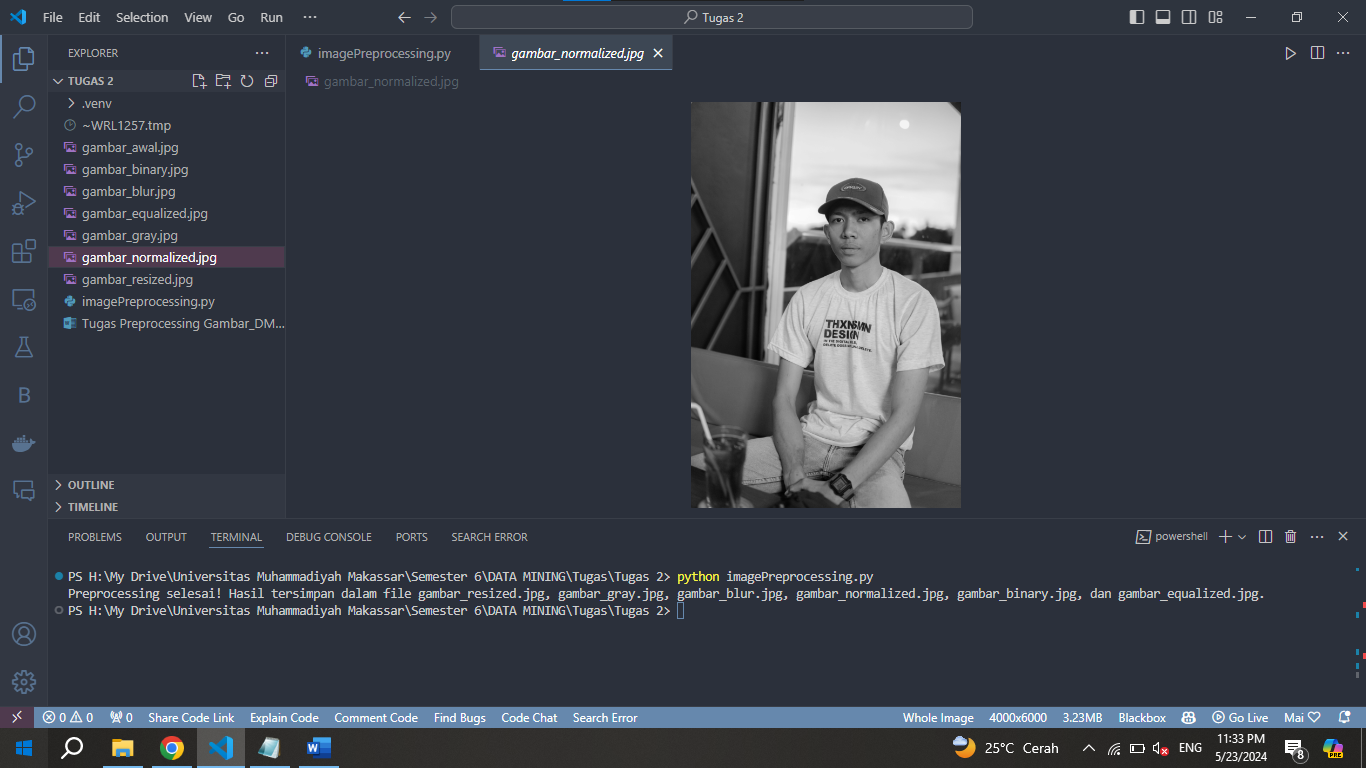
[](https://drive.google.com/open?id=1Px10DDXQtV-0uqDb5wymteENDLumZ8tC&usp=drive_fs)

Nilai piksel gambar blur diubah agar berada dalam rentang 0-255.

1. Binarization (Konversi ke Hitam-Putih)

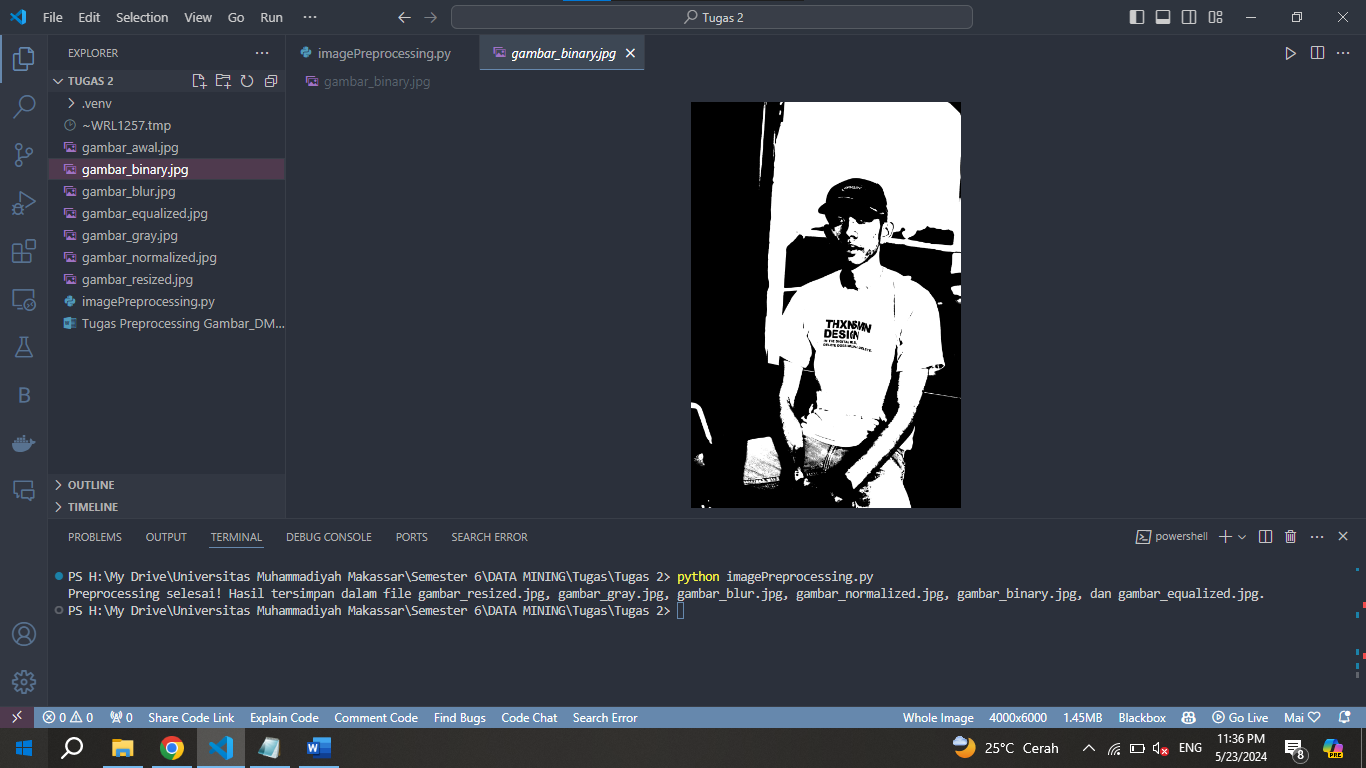
`cv2.threshold(gambar\_normalized, 128, 255, cv2.THRESH\_BINARY)`: Mengubah gambar normalized menjadi gambar hitam-putih dengan threshold 128. Piksel dengan nilai lebih besar dari 128 menjadi putih, sedangkan sisanya menjadi hitam.

Gambar sebelum:

[](https://drive.google.com/open?id=1Px10DDXQtV-0uqDb5wymteENDLumZ8tC&usp=drive_fs)

Gambar grayscale dengan nilai piksel 0-255.

Gambar setelah:

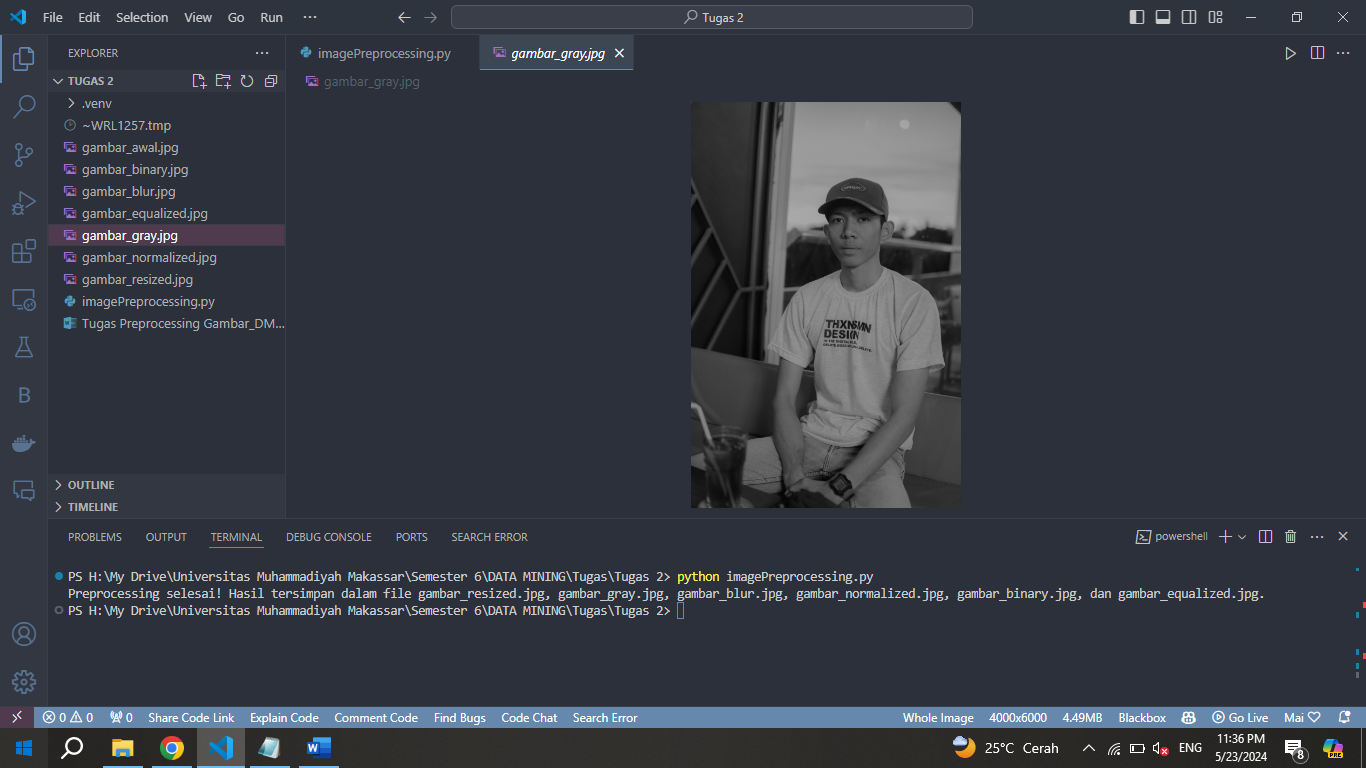
[](https://drive.google.com/open?id=1PwL6weWTqMBMYvQzEJHsazkeRukTDLny&usp=drive_fs)

Gambar hitam-putih dengan dua nilai piksel (0 dan 255).

1. Contrast Enhancement (Peningkatan Kontras)

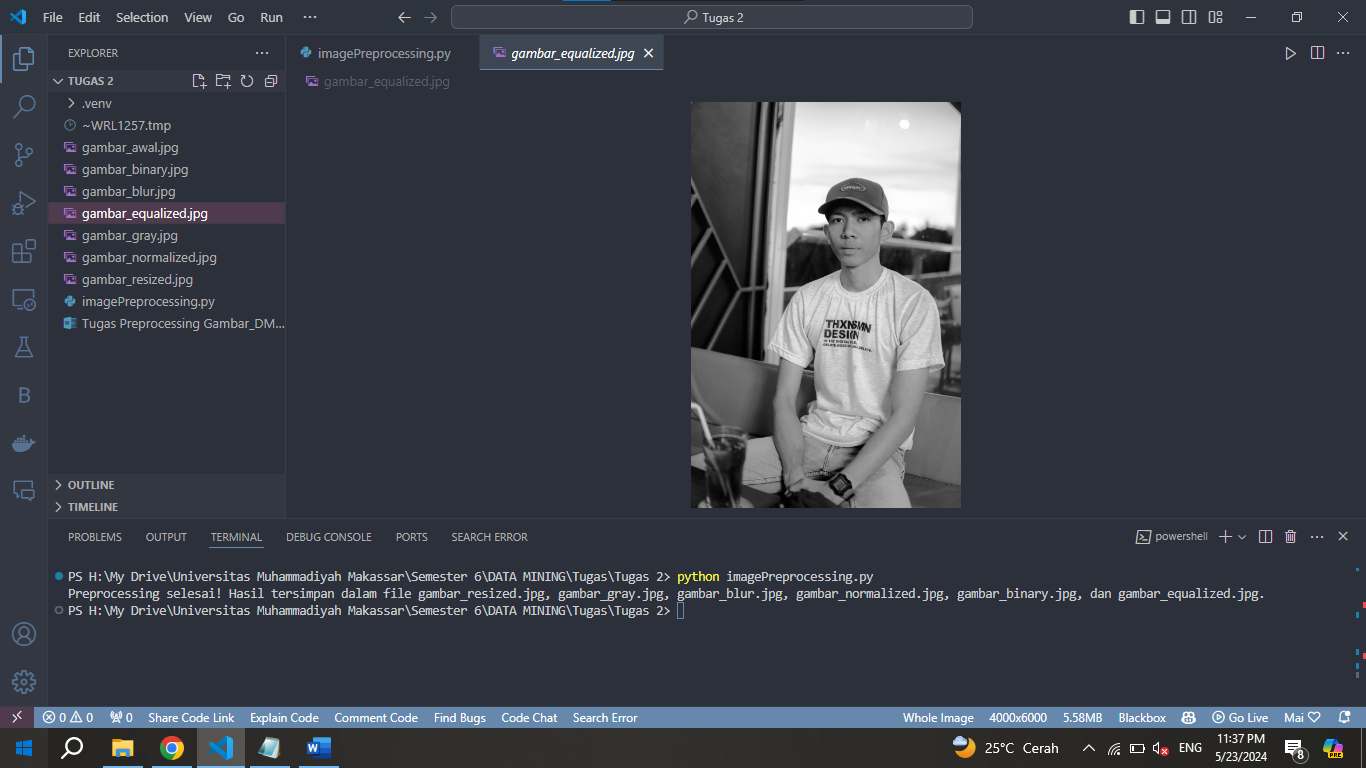
`cv2.equalizeHist(gambar\_gray)`: Meningkatkan kontras gambar grayscale dengan mendistribusikan kembali nilai piksel secara merata.

Gambar sebelum:

[](https://drive.google.com/open?id=1Q-_tKfJu7wUWeMY-LqClFFx4sr0MDpmT&usp=drive_fs)

Gambar grayscale dengan distribusi kontras yang mungkin tidak merata.

Gambar setelah:

[](https://drive.google.com/open?id=1Q0qu_Dx8UgyoZh4Z6uF9SS8vINZO18VY&usp=drive_fs)

Gambar grayscale dengan kontras yang lebih ditingkatkan dan distribusi piksel yang lebih merata.

Penyimpanan Hasil Preprocessing:

* `cv2.imwrite('gambar\_resized.jpg', gambar\_resized)`: Menyimpan gambar yang telah diubah ukurannya ke file `gambar\_resized.jpg`.
* `cv2.imwrite('gambar\_gray.jpg', gambar\_gray)`: Menyimpan gambar grayscale ke file `gambar\_gray.jpg`.
* `cv2.imwrite('gambar\_blur.jpg', gambar\_blur)`: Menyimpan gambar blur ke file `gambar\_blur.jpg`.
* `cv2.imwrite('gambar\_normalized.jpg', gambar\_normalized)`: Menyimpan gambar normalized ke file `gambar\_normalized.jpg`.
* `cv2.imwrite('gambar\_binary.jpg', gambar\_binary)`: Menyimpan gambar binary ke file `gambar\_binary.jpg`.
* `cv2.imwrite('gambar\_equalized.jpg', gambar\_equalized)`: Menyimpan gambar equalized ke file `gambar\_equalized.jpg`.