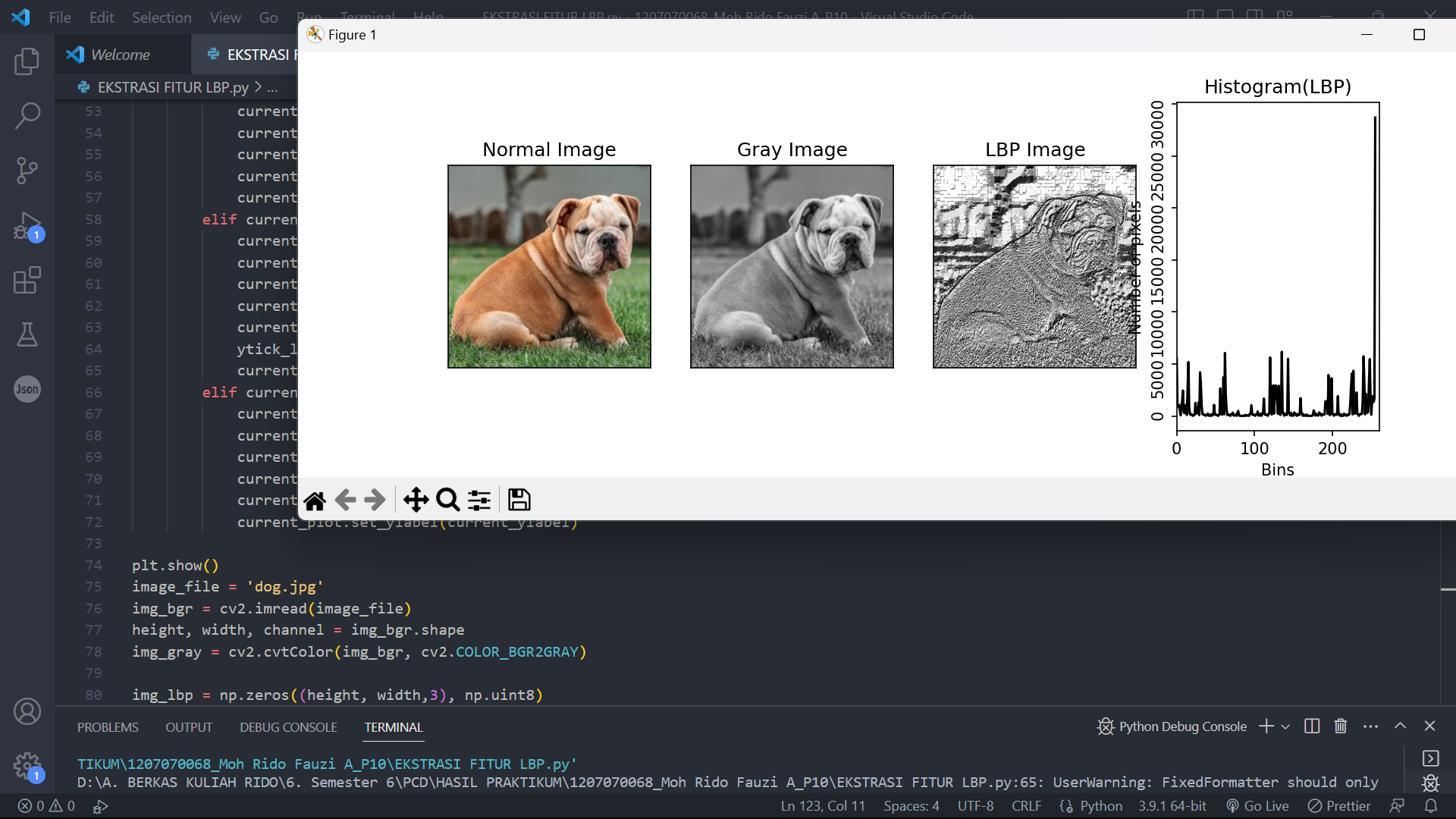
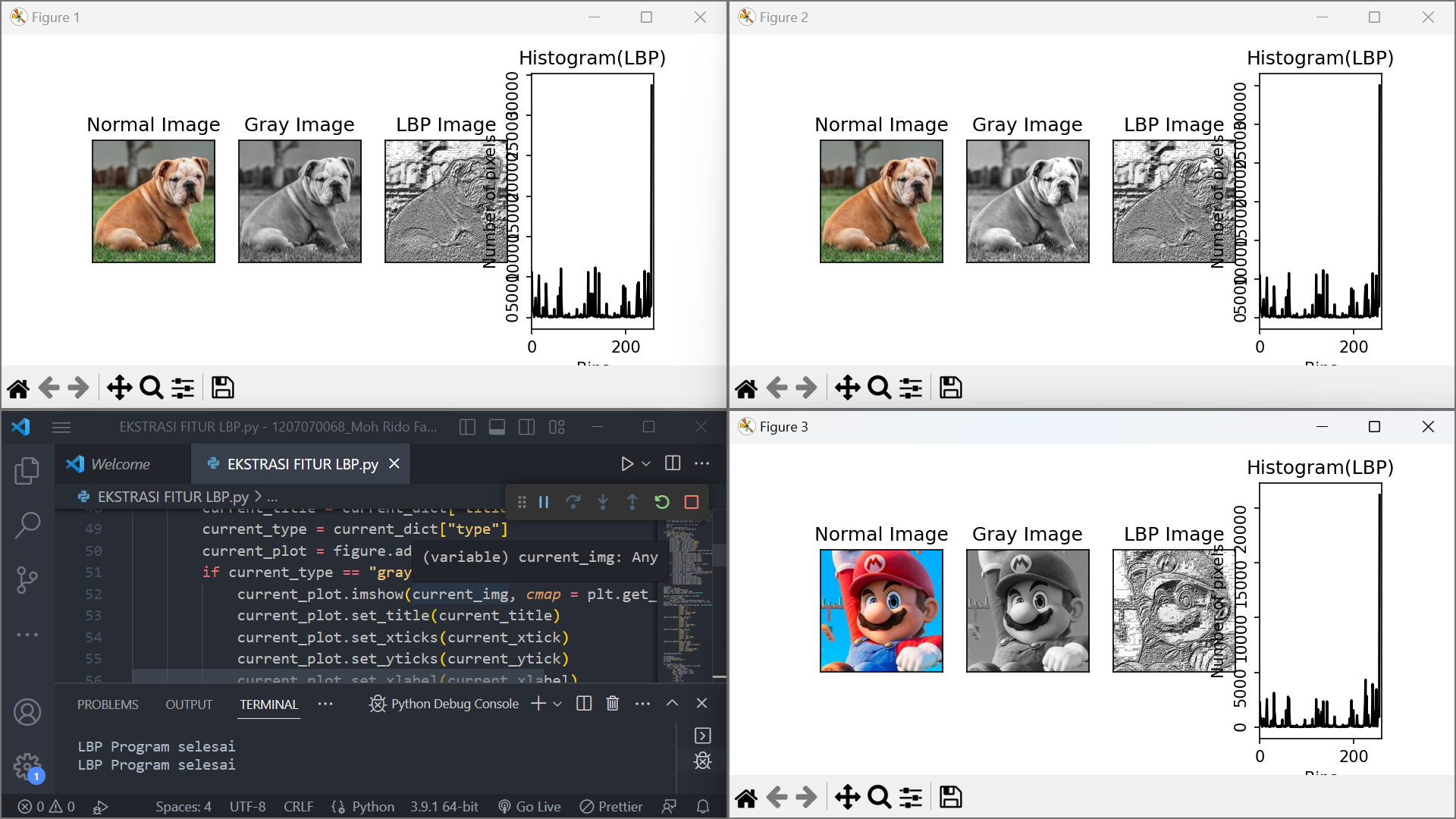
NAMA : **MOH RIDO FAUZI ARBIANSYAH**  
NIM : **1207070068**  
KELAS : **TKK**

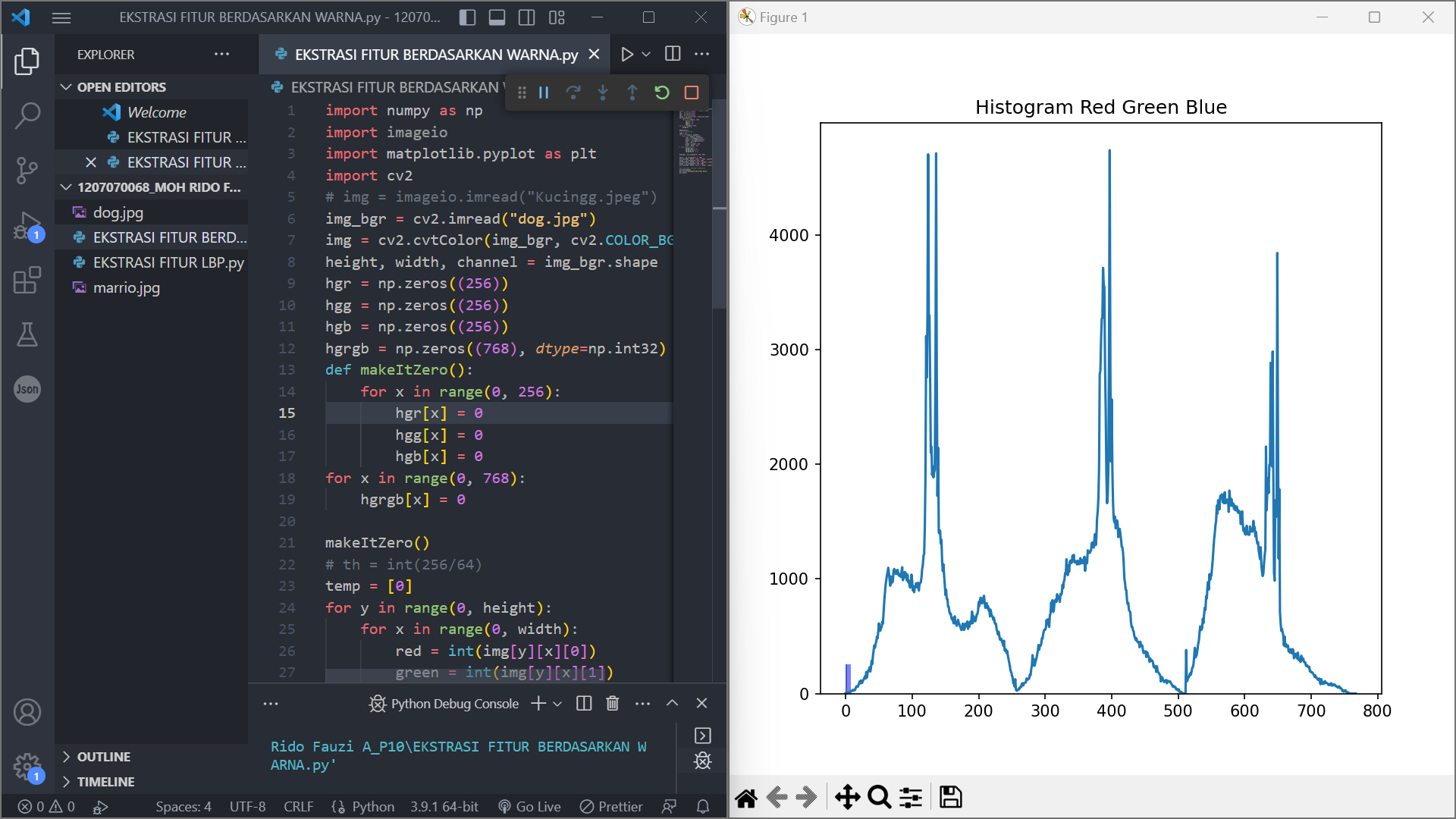
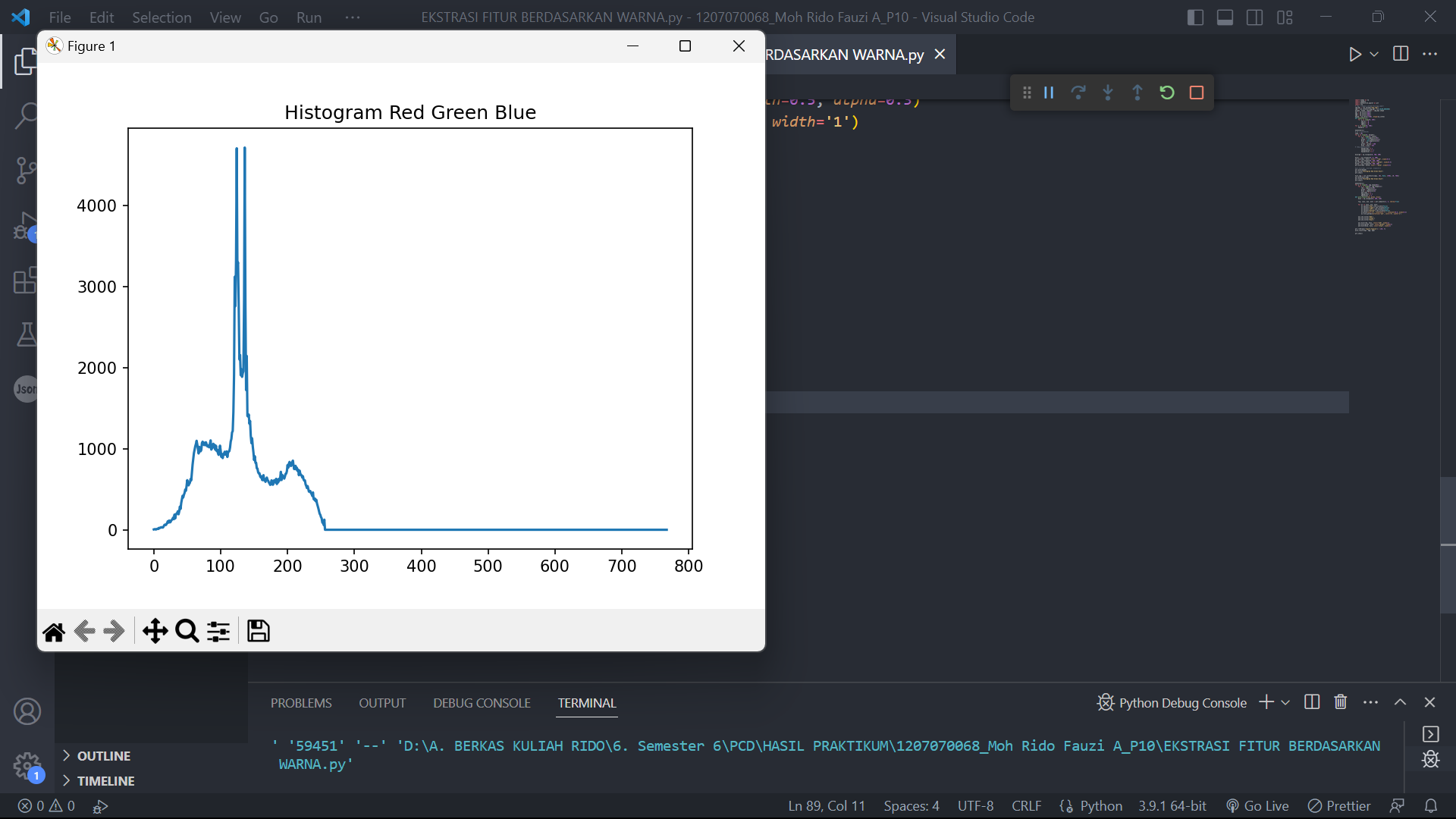
<https://github.com/ridofauzi/1207070068_MohRidoFauziA_P10.git>

Latihan 1

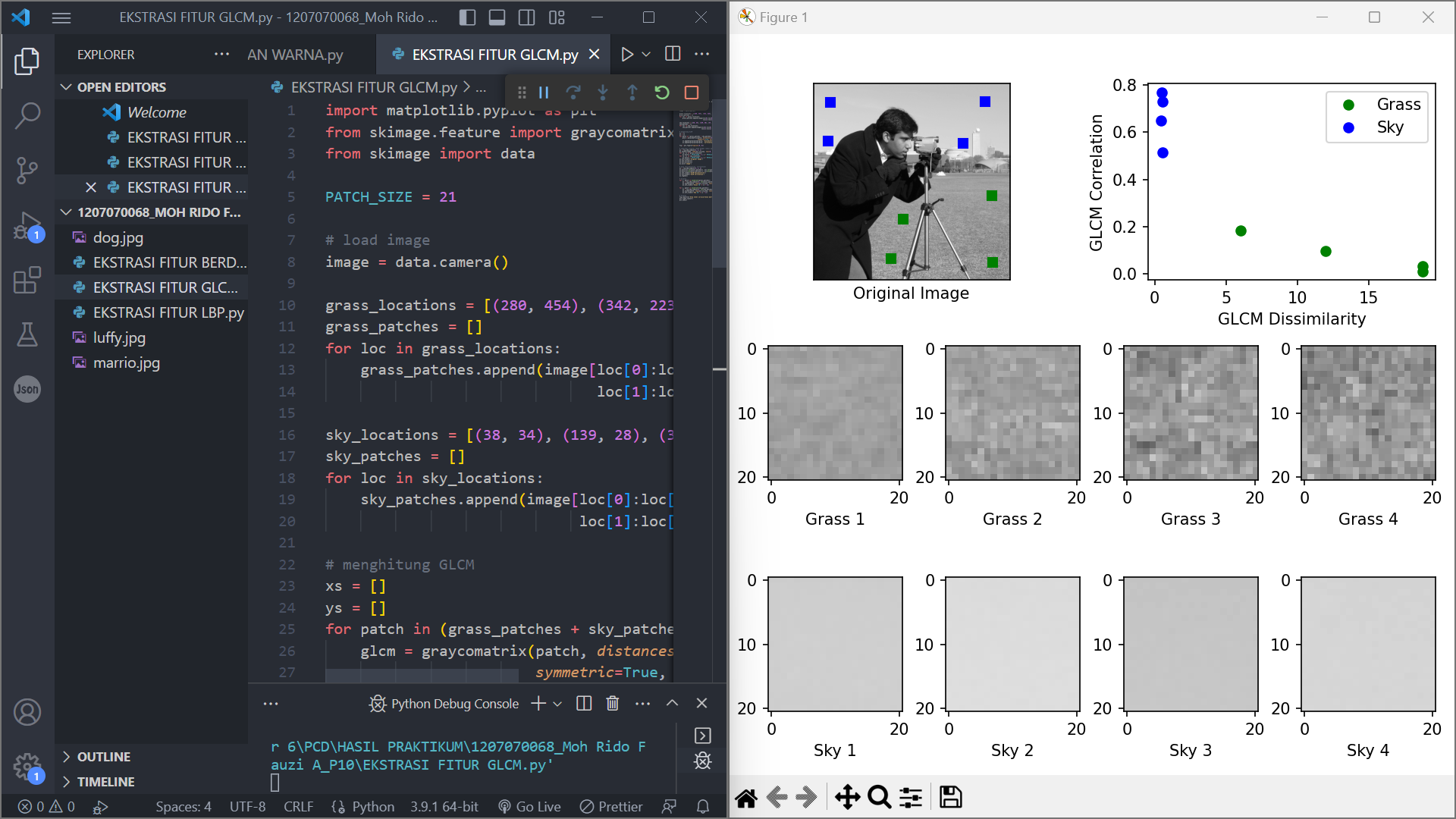


Program ini mengambil gambar (dalam format BGR) dan mengubahnya menjadi gambar skala abu-abu sebelum penghapusan LBP. Hasil ekstraksi LBP ditampilkan dalam bentuk gambar LBP dan histogram LBP. Dalam gambar LBP, setiap piksel dijelaskan oleh nilai LBP yang dihasilkan berdasarkan piksel di sekitarnya. Histogram LBP menunjukkan distribusi frekuensi nilai LBP pada citra. Program ini juga menyertakan fungsi untuk mengubah kecerahan gambar dengan menambahkan nilai ke setiap saluran warna. Ekstraksi dan histogram LBP juga ditampilkan untuk gambar dengan perubahan kecerahan.

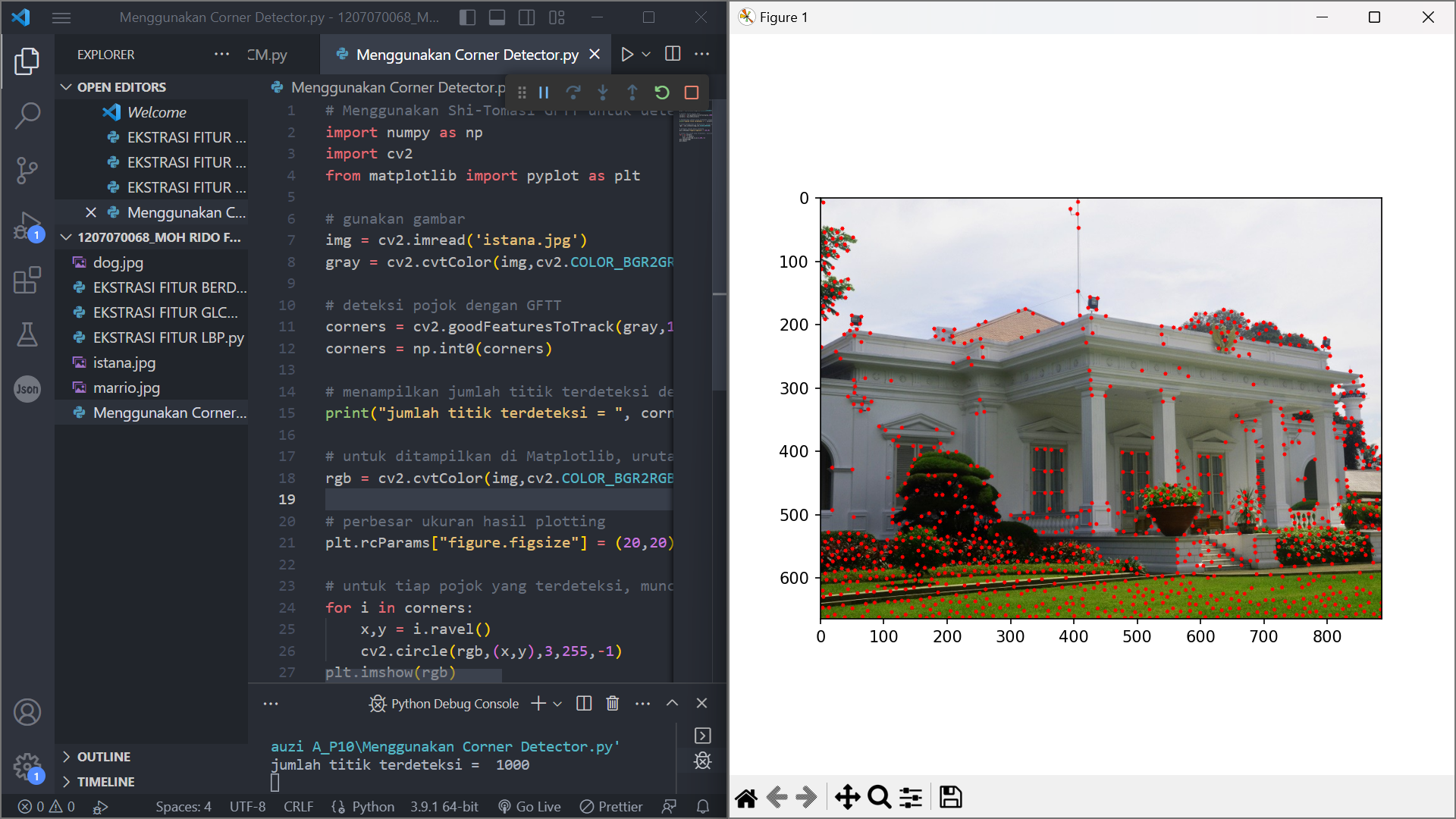
  
Latihan 2



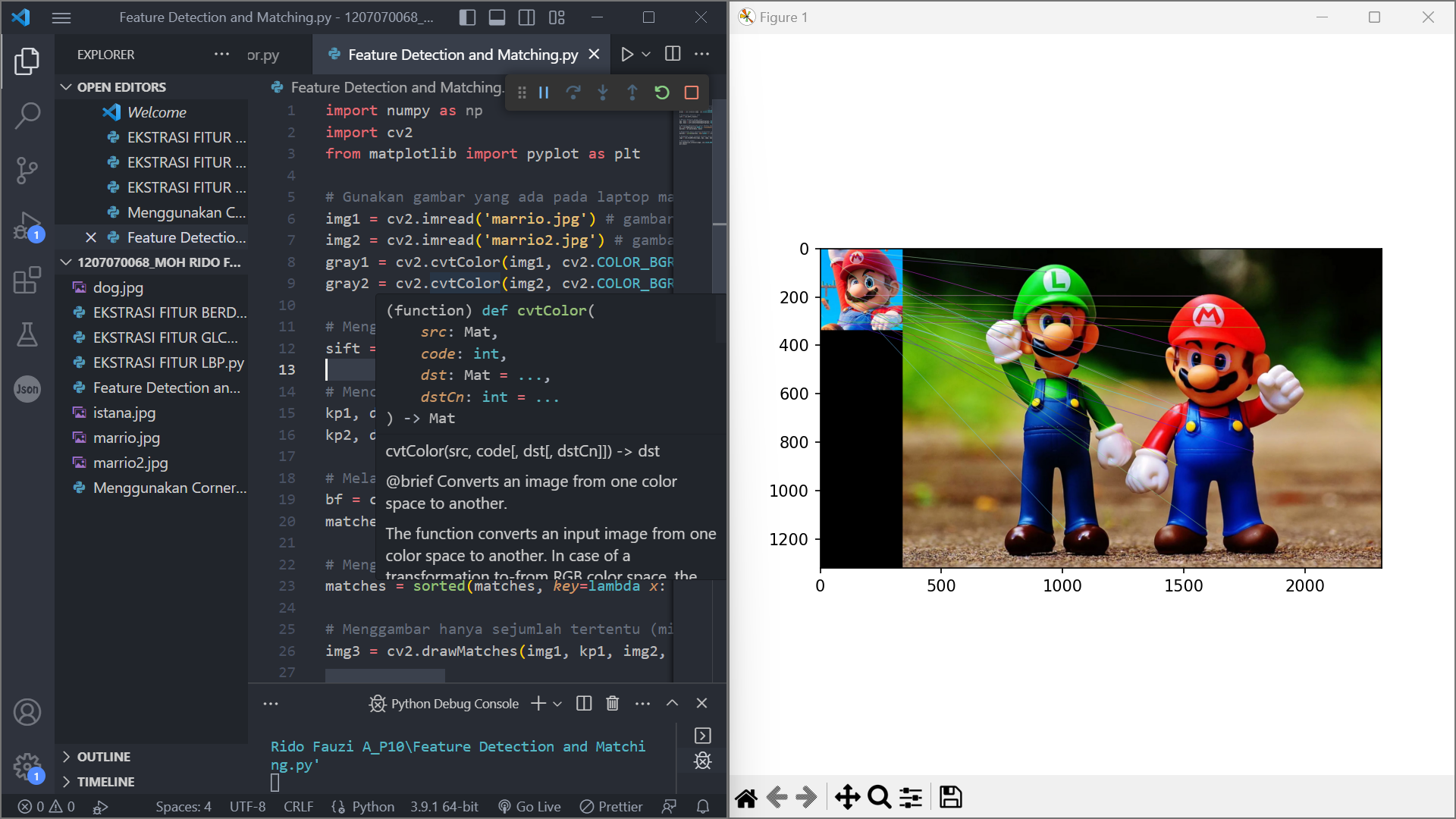
Pada percobaan ini kami menggunakan metode ekstraksi fitur berbasis warna, biasanya melibatkan ruang warna seperti RGB, HSV, atau YUV. Properti seperti histogram warna atau momen warna dapat diekstraksi dari gambar. Keuntungan dari metode ini adalah banyak informasi warna yang dapat ditangkap dalam gambar. Namun, metode ini mungkin tidak peka terhadap perubahan tekstur atau bentuk objek. Tujuan dari program ini adalah menghitung dan menampilkan histogram dari suatu citra berwarna. Histogram digunakan untuk menggambarkan distribusi intensitas warna dalam suatu gambar. Dengan melakukan analisis histogram, kita mendapatkan informasi tentang sebaran warna pada citra. Histogram merah, hijau, dan biru yang terpisah memberi kesan bagaimana setiap komponen warna memengaruhi gambar. Histogram gabungan memberikan gambaran keseluruhan tentang distribusi warna pada gambar.

Latihan 3

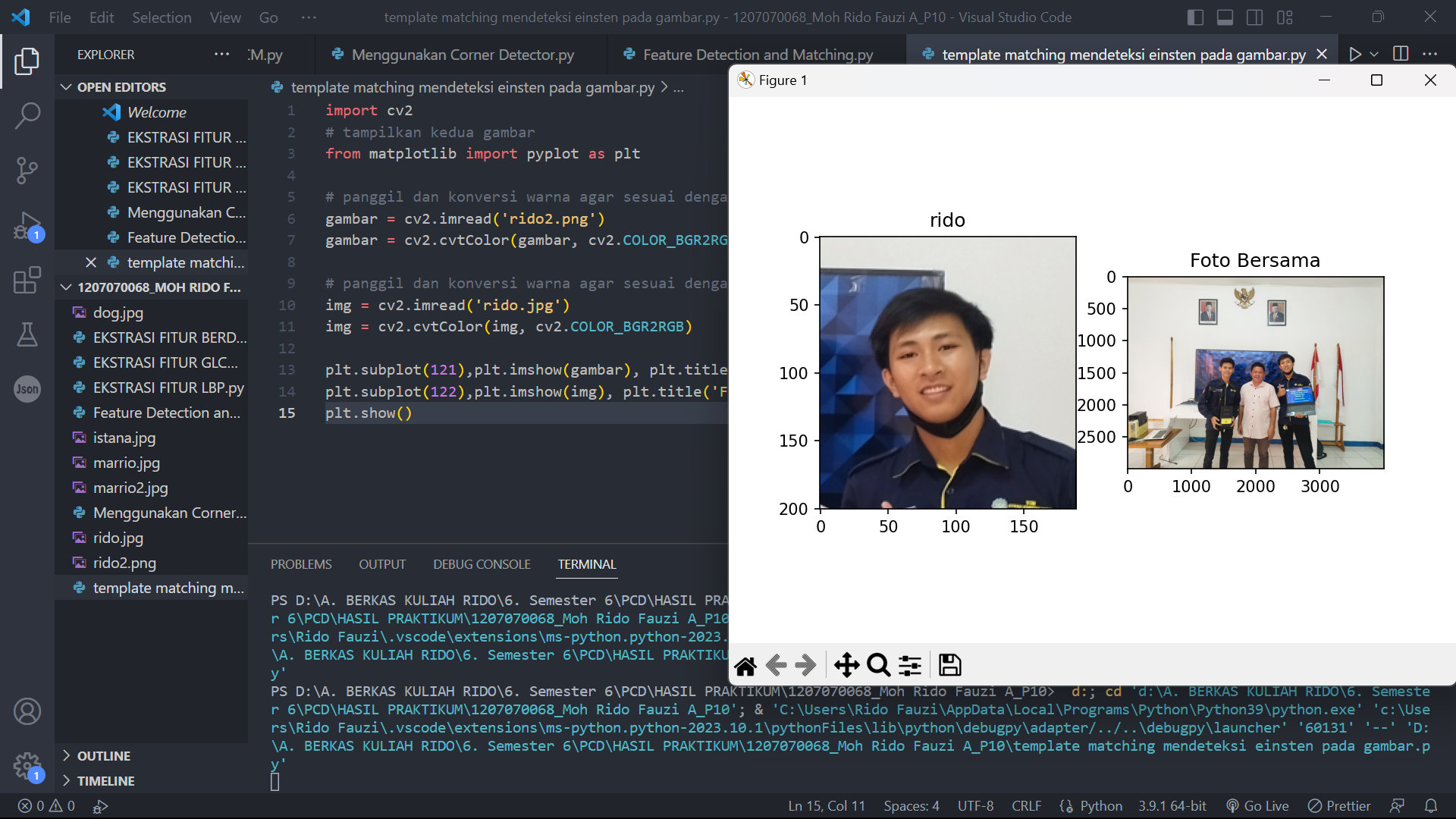
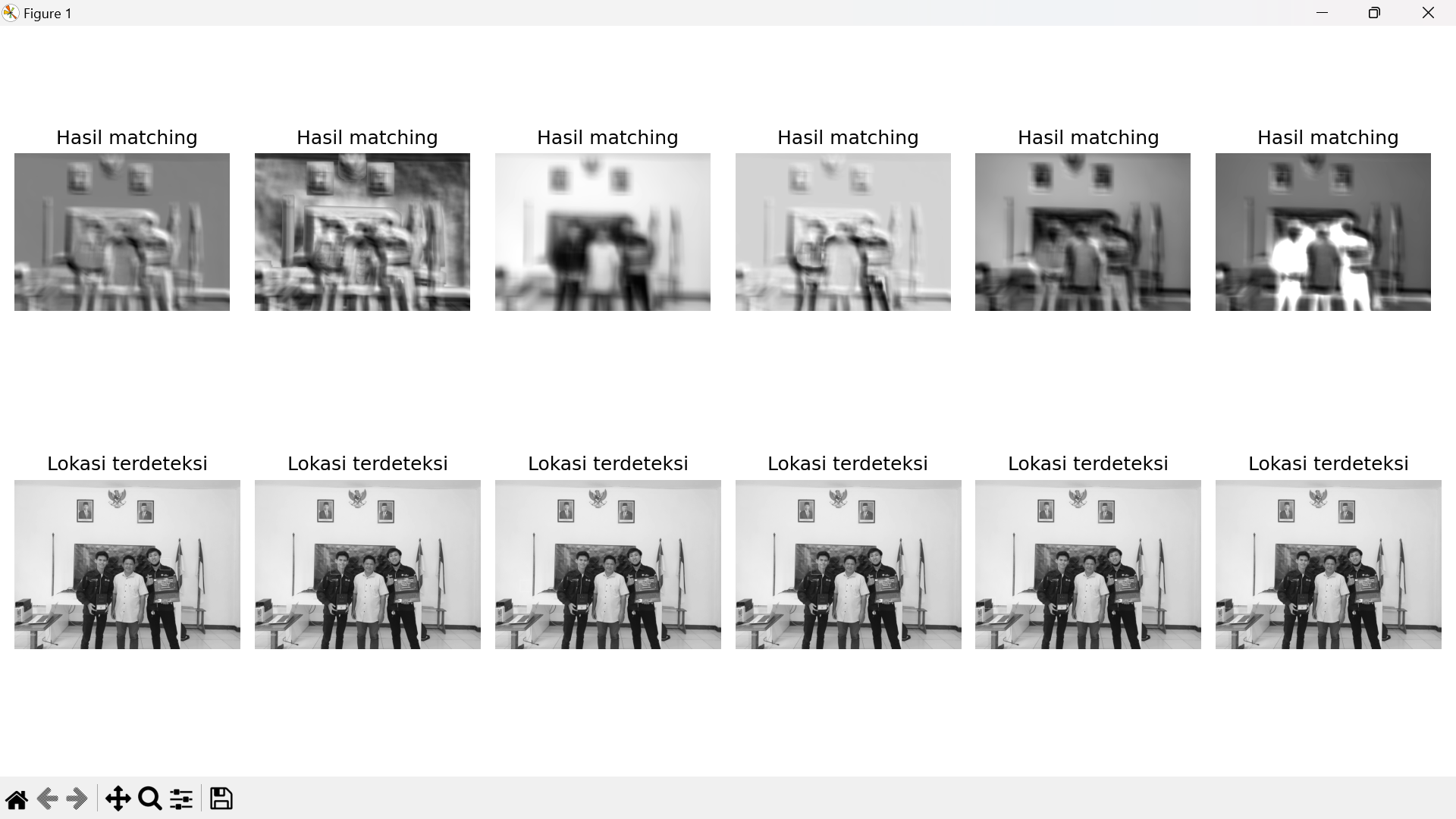
Pada latihan ke 3 ini, metode GLCM digunakan untuk mengekstrak fitur tekstur pada citra dengan menganalisis hubungan spasial antar intensitas piksel. The Gray Level Co-occurrence Matrix (GLCM) menghitung kemunculan sepasang intensitas piksel tertentu pada jarak dan sudut tertentu. Properti seperti kontras, korelasi, energi dan homogenitas dapat dihitung dari matriks ini. Keuntungan dari metode GLCM adalah dapat menangkap sifat struktural yang kompleks dan tidak berubah untuk transformasi kecil. Namun, metode ini membutuhkan komputasi yang lebih banyak dan sensitif terhadap cahaya.

**Latihan 4**

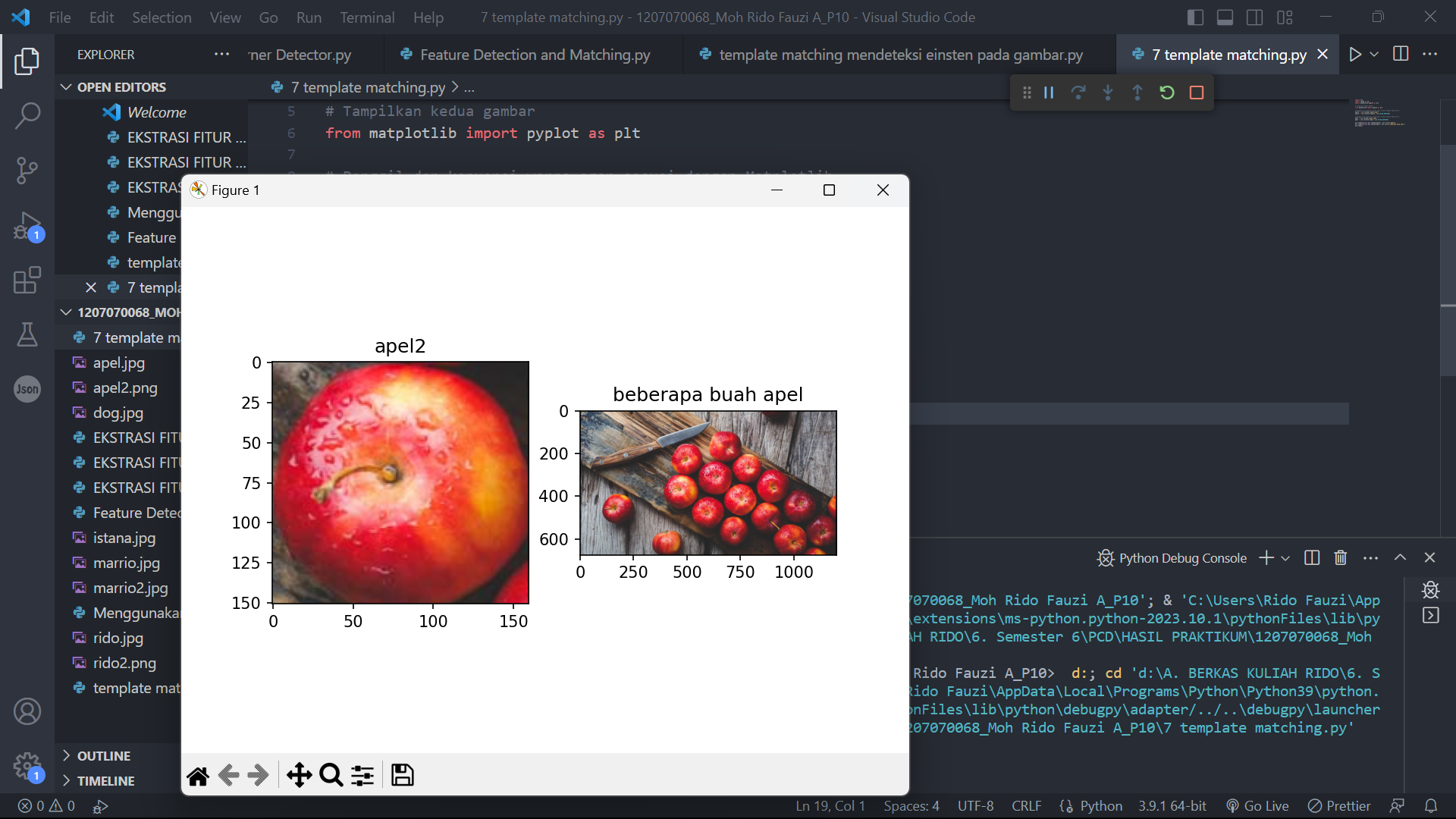
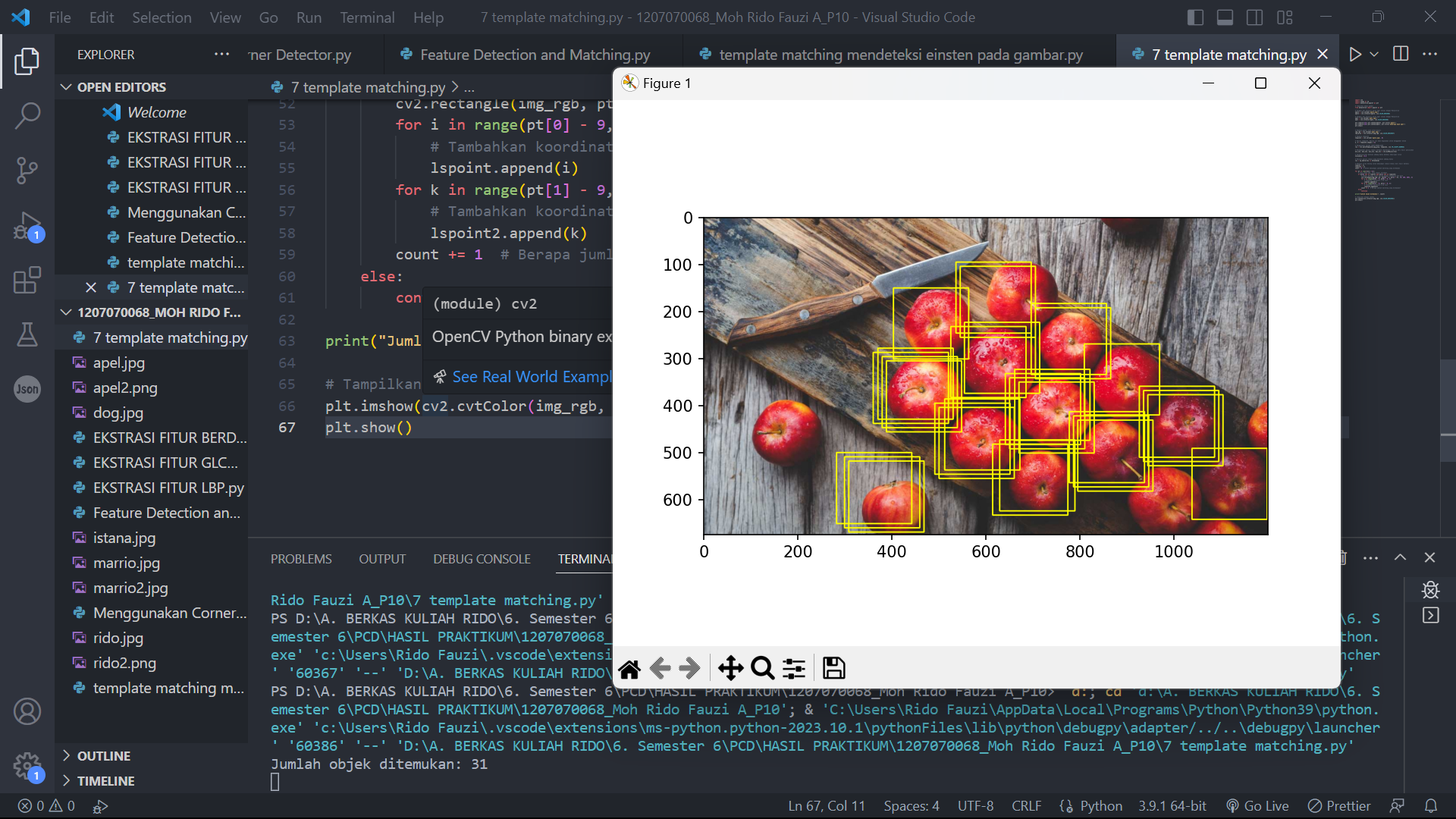
Program ini menggunakan metode Shi-Tomasi Good Features to Track (GFTT) untuk mendeteksi tepi atau sudut gambar. Output dari program menunjukkan gambar asli, di mana titik akhir yang terdeteksi ditandai dengan lingkaran. Jumlah titik yang terdeteksi juga dicetak. Ini berguna untuk mengidentifikasi sudut atau tepi gambar yang dapat digunakan untuk analisis lebih lanjut. Algoritme deteksi sudut seperti Harris Corner Detector atau FAST (Features of Accelerated Segment Test) dapat digunakan. Kelebihan dari metode ini adalah kemampuan untuk menemukan titik-titik kunci yang dapat digunakan untuk memisahkan objek atau memetakan perubahan sudut yang signifikan. Namun, metode ini kurang efektif dalam mendeskripsikan karakteristik tekstur atau warna citra.

Latiham 5

Program yang digunakan tersebut melakukan pencocokan fitur antara dua citra dengan menggunakan metode SIFT (Scale-Invariant Feature Transform) dan menampilkan pasangan yang paling cocok dari citra yang dihasilkan. Output dari program menunjukkan gambar yang dihasilkan dengan pasangan yang paling cocok ditandai pada gambar. Ini berguna saat Anda ingin melihat bagaimana fitur cocok antara dua gambar dan dapat digunakan dalam aplikasi deteksi objek, penyajian gambar, atau penggabungan gambar.

**Latihan 6**

Dalam Latihan 6 ini, program ini terdiri dari dua bagian. Bagian pertama menampilkan dua gambar secara bersamaan menggunakan matplotlib, sedangkan bagian kedua melakukan pencocokan pola antara kedua gambar tersebut. Output dari program ini menunjukkan hasil pencocokan dan posisi pencocokan pola yang terdeteksi antara dua gambar. Ini berguna untuk aplikasi deteksi objek atau mencari objek dalam gambar berdasarkan pola yang telah ditentukan.

Latihan 7

Pada percobaan ke 7 ini juga terdapat dua bagian pada Program. Dimana bagian pertama menampilkan dua gambar, "apel2.png" dan "apel.jpg", menggunakan matplotlib. Gambar pertama ("apel2.png") ditampilkan di subplot kiri (subplot 121), sedangkan gambar kedua ("apel.jpg") ditampilkan di subplot kanan (subplot 122). Bagian kedua program melakukan template matching antara gambar "apel.jpg" dan "apel2.png" menggunakan metode COEFF-NORMALIZED (cv2.TM\_CCOEFF\_NORMED). Keluaran dari program ini adalah gambar apel.jpg dengan persegi panjang berwarna kuning yang menunjukkan posisi objek yang sesuai dengan template apel2.png. Program ini cocok untuk pengenalan objek berdasarkan pola yang telah ditentukan dalam gambar.