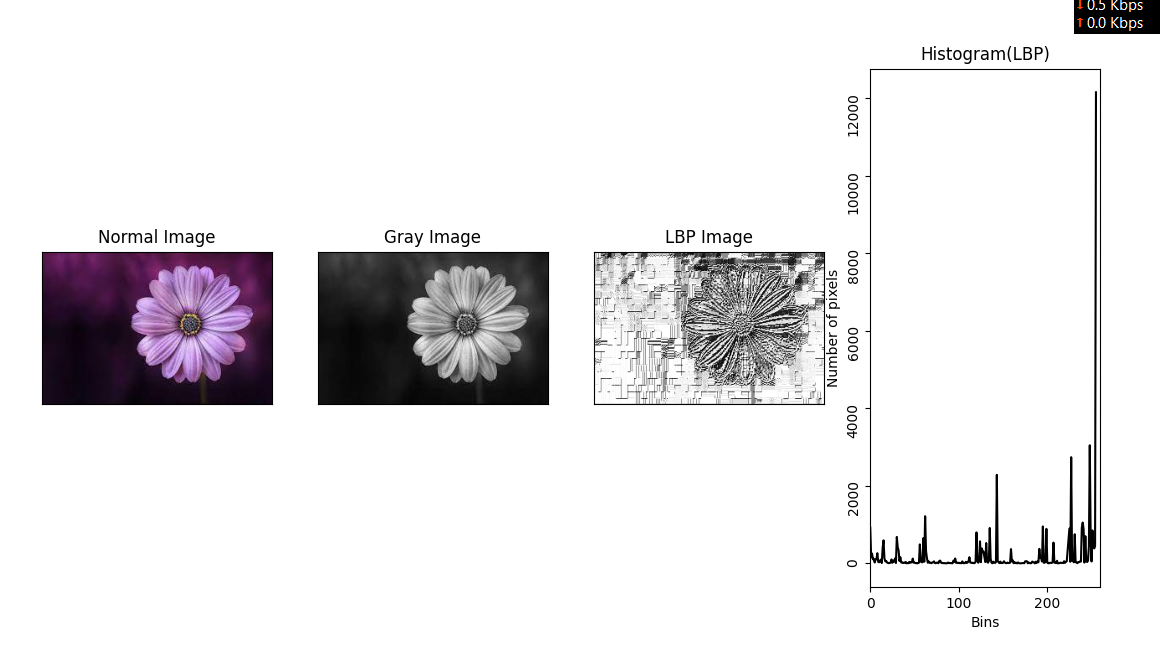
Nama : Hardiansyah Firdaus

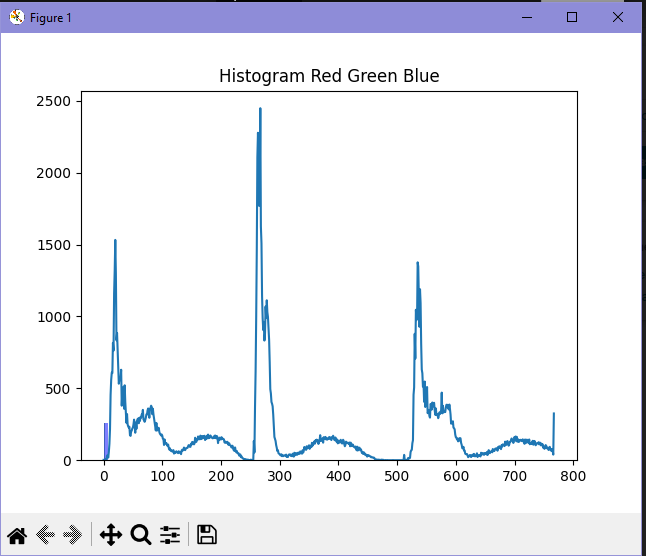
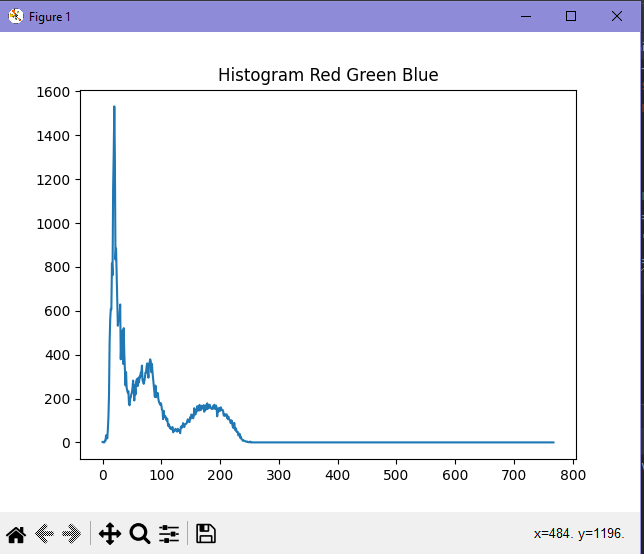
NIM : 1207070052

Latihan 1 ekstraksi Fitur LBP



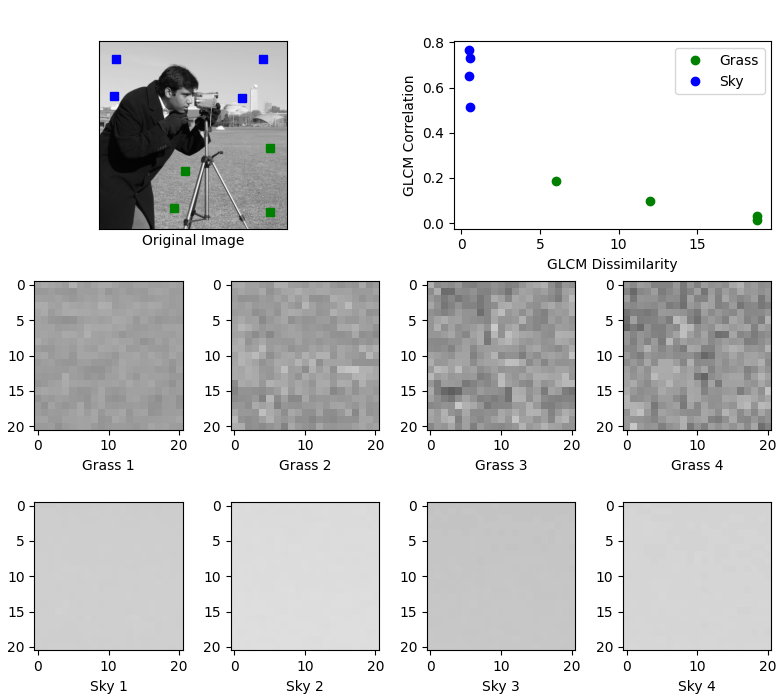
Bisa dilihat pada hasil dari Latihan 1, cukup pekat atau tajam untuk pendeteksian menggunakan metode LBP, bisa dikatakan sangat tampak jelas untuk pendeteksian garis tepi dari bunga pada gambar.

Latihan 2 Ekstraksi Fitur berdasarkan Warna

Dapat dilihat pada gambar diatas, gambar yang kiri terbentuk dari penentuan plot(0,768,100) dan yang kanan linspace(0,256, 128). Program tersebut yang terdiri dari 100 angka yang terdistribusi secara merata dari 0 hingga 768. Ini akan digunakan sebagai batasan bin untuk histogram warna keseluruhan (RGB). Lalu membuat histogram warna antara merah, hijau, dan biru dengan batasan bin yang digunakan serta menentukan transparansi histogram pada nilai 0.5. Pada gambar 1 masing2 RGB dipisah setiap penentuan histogramnya(posisinya), sedangkan gambar kedua itu menggabungkan dan menampilkan hasil merata antar RGB tersebut.

Latihan 3 EKSTRASI FITUR GLCM



Pada percobaan kali ini saya mencoba menganti ke gambar pribadi eror terus menerus, sudah dikonvert ke greyscale juga masih tetap sama, jdi dapat digambarkan dari hasil latihan 3 ini, pendeteksian langit dan rumput sudah sesuai dengan gambar aslinya, hasil pendeteksiannya ditandai dengan titik2 hijau untuk rumput dan biru untuk langit.

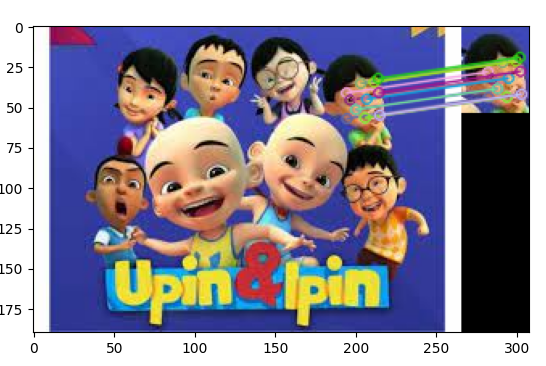
Latihan 4 Corner Detector



Dapat diamati pendeteksian garis tepi menggunakan metode corner detector tidak begitu akurat, masih ada beberapa yang tidak terdeteksi, mungkin ini dikarenakan foto yang digunakan saat kondisi malam hari.

jumlah titik terdeteksi = 1000, berikut jumlah titik yang terdeteksi pada gambar diatas

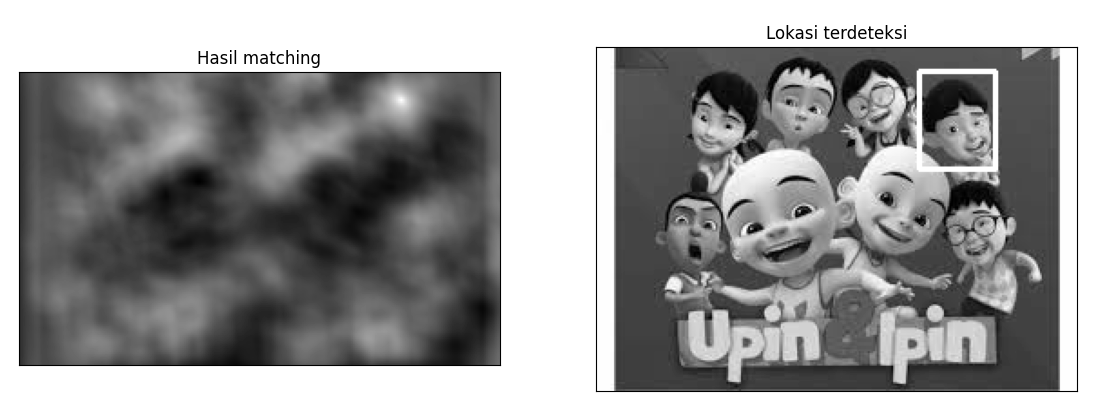
Latihan 5 Feature Detection and Matching



Pada hasil tersebut gambar sebelah kanan untuk pendeteksian gambar mail cocok atau bisa dikatakan sesuai dengan gambar template

Latihan 6 Template Matching pada gambar

hasil metode cv2.TM\_CCOEFF :

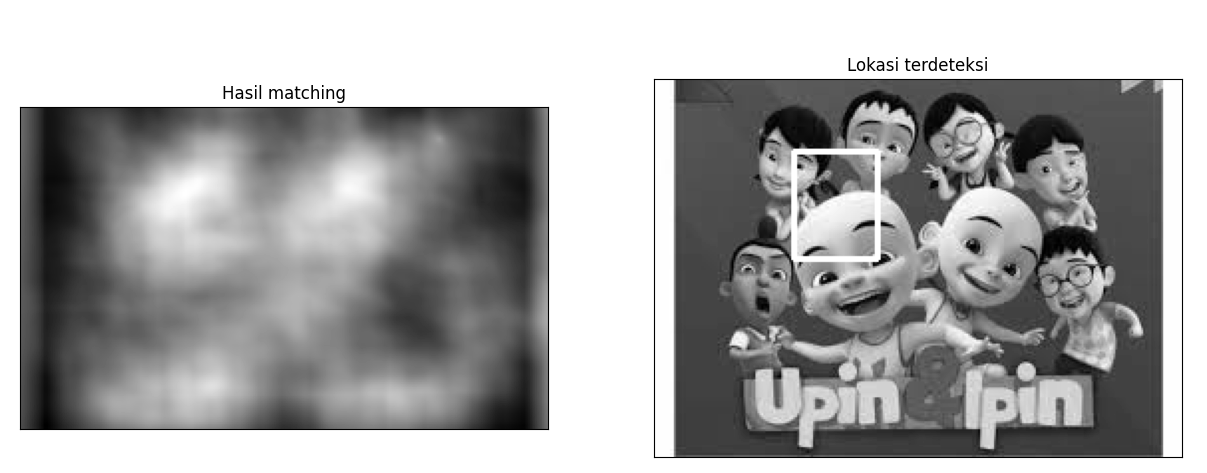


hasil metode cv2.TM\_CCOEFF\_NORMED :

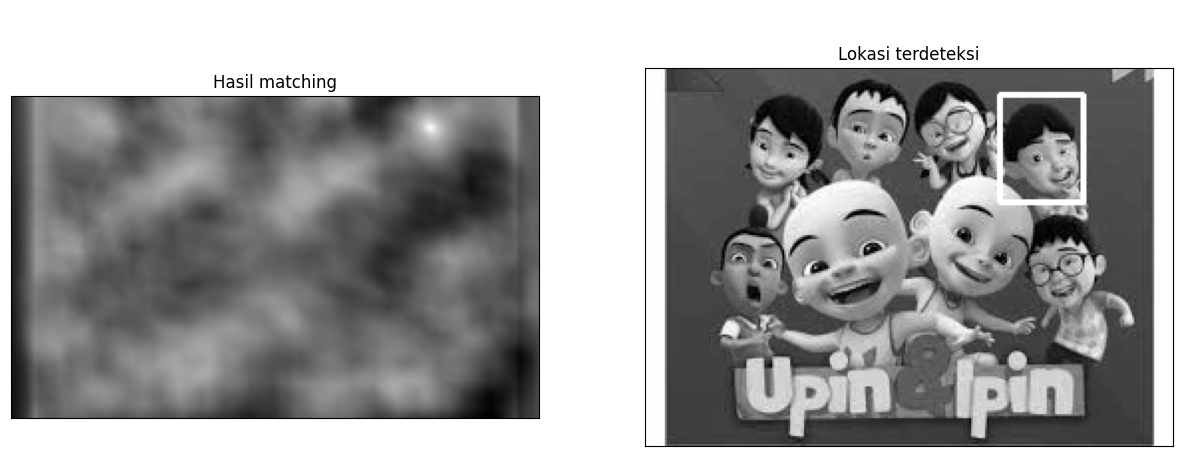




hasil metode cv2.TM\_CCORR :



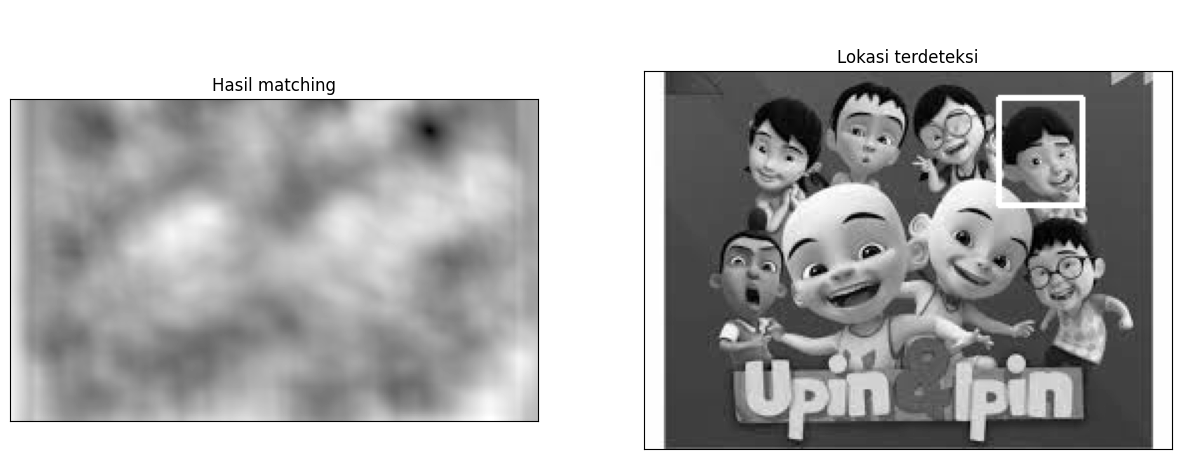
hasil metode cv2.TM\_CCORR\_NORMED :



hasil metode cv2.TM\_SQDIFF :

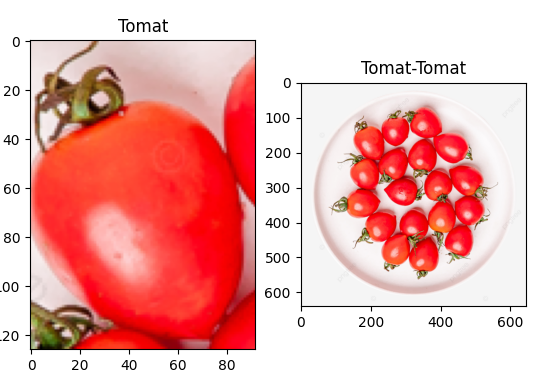
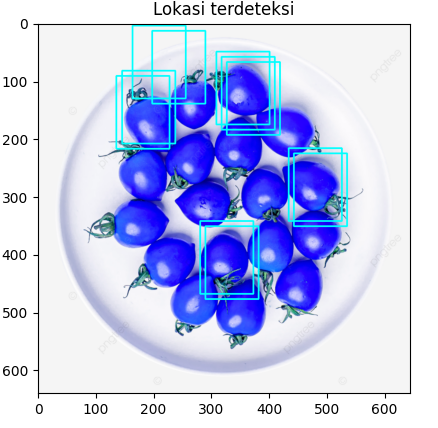


hasil metode cv2.TM\_SQDIFF\_NORMED :



Dari hasil perbandingan keenam gambar hanya 1 dari 6 saja yang menentukan hasil yang tidak akurat, tetapi sisanya menunjukan hasil yang sesuai dengan posisi gambar yang dituju.

Latihan 7 Menghitung Tomat dengan template matching

Gambar sebelah kiri merupakan gambar aslinya, untuk gambar yang kedua merupakan hasil pendeteksian lokasi gambar yang dituju, disini menggunakan 0.35 untuk thresholdnya, karena jika menggunakan threshold 0.15 hasil yang ditemukan terlalu banyak, jdi saya menggunakan thresholdnya yang terdekat saja

Jumlah objek ditemukan 11, brikut data yang ditemukan ketika menggunakan metode Template Matching