**LAPORAN PRAKTIKUM**

**PENGOLAHAN CITRA DIGITAL**



Disusun Oleh :

**Muhammad Awan Ardy Firmansyah**

**F55120041**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TADULAKO**

**TAHUN 2024**

1. **TUJUAN**
2. Mahasiswa mampu memahami apa itu Transformasi Negatif
3. Mahasiswa mampu membuat Transformasi Negatif dengan *Python*
4. **ALAT DAN BAHAN**
5. Laptop
6. *Visual Studio Code*
7. *Python*
8. *Library OpenCV*
9. **TEORI DASAR**

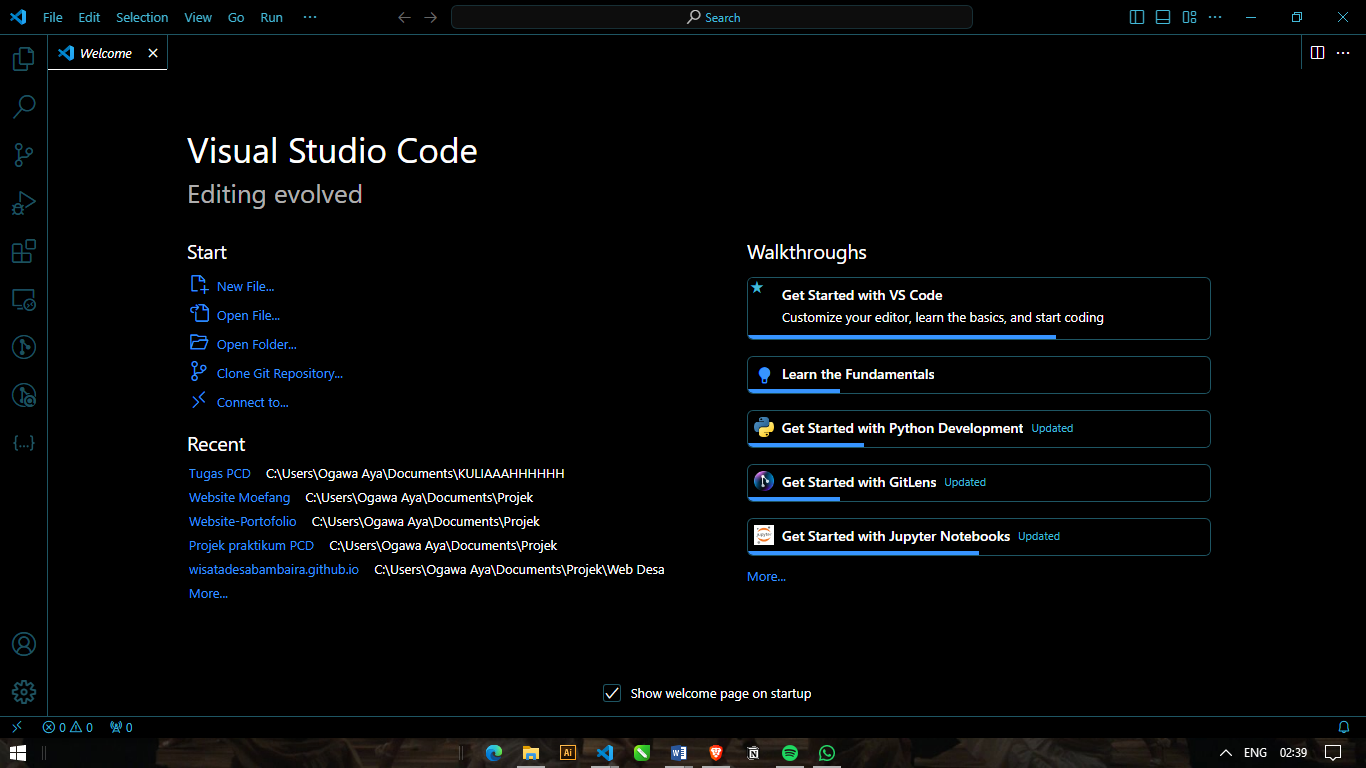
Negatif dari suatu citra dengan tingkat keabuan antara [0, L-1] dapat dihitung mengunakan transformasi negative dengan rumus berikut :

s = L – 1 – r

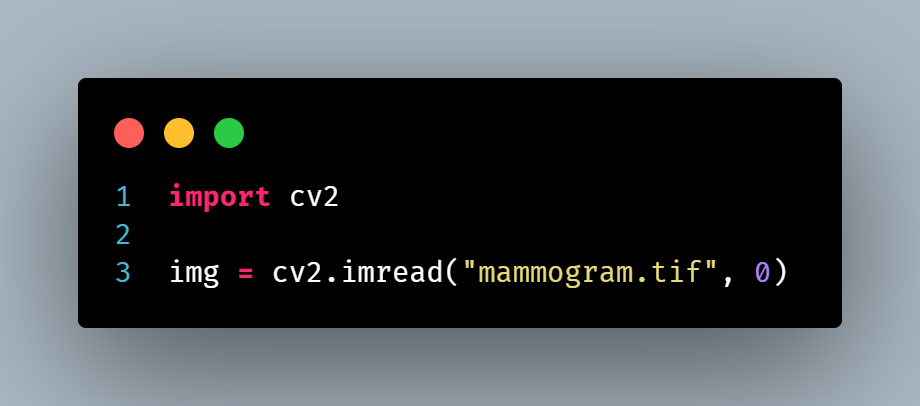
Membalik intensitas citra dengan rumus seperti di atas akan menghasilkan negatif dari foto. Pencarian negatif dari suatu citra cocok untuk memperbaiki gambar yang memiliki rincian sub citra terang pada area yang gelap, khususnya jika ukuran dari area gelap cukup dominan.

Transformasi negatif merupakan salah satu teknik dalam pemrosesan citra yang digunakan untuk mengubah citra menjadi negatifnya. Dalm konteks citra digital, citra tersebut biasanya direpresentasikan sebagai array dua dimensi untuk citra grayscale atau tiga dimensi untuk citra RGB. Teknik ini melibatkan inversi nilai piksel dari nilai maksimal yang mungkin dalam citra tersebut. Misalnya, dalam citra *grayscale* 8-bit, nilai piksel dapat berkisar dari 0 (hitam) hingga 255 (putih). Transformasi negative akan mengubah setiap piksel dengan nilai V menjadi 255 – V sehingga menghasilkan efek negatif pada citra.

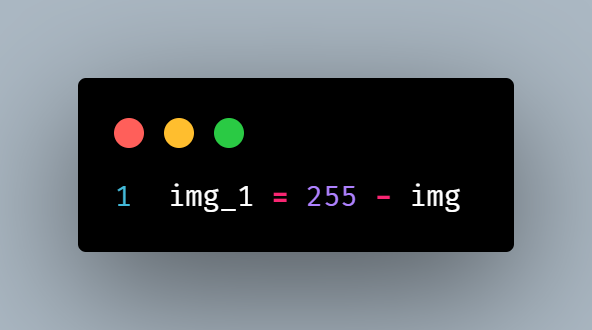
1. **LANGKAH KERJA**
2. Masuk ke aplikasi *Visual Studio Code* dan klik “*New File*” untuk membuat file baru.



1. Masukkan kode berikut untuk memasukkan *library* dan memasukkan gambar.



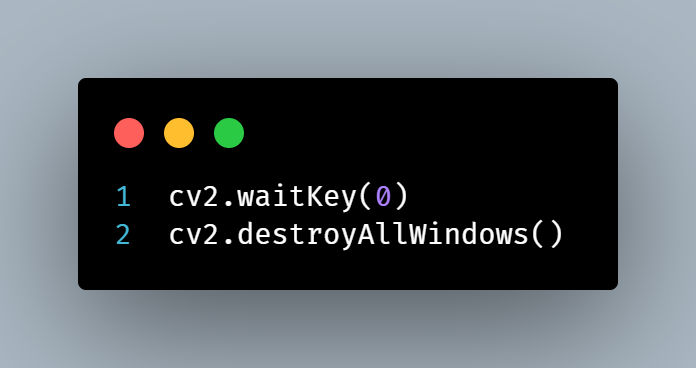
1. Masukkan kode berikut untuk menjalankan fungsi Transformasi Negatif



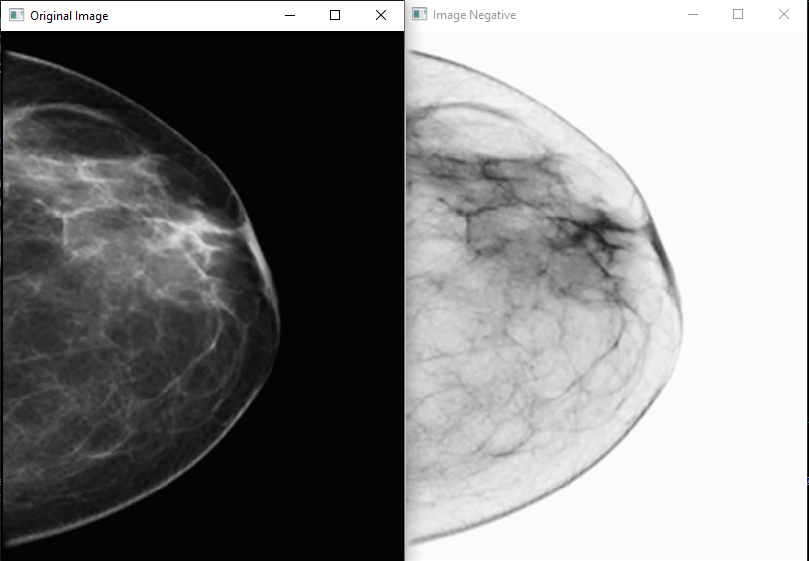
1. Masukkan kode berikut untuk menampilkan gambar original dan hasil dari Transformasi Negatif



1. Masukkan kode berikut



1. **HASIL PERCOBAAN**

****

1. **ANALISIS**

Kode ini awalnya mengimpor *library OpenCV* untuk menggunakan fungsi manipulasi gambar. Kemudian, kode tersebut membaca gambar dalam skala abu-abu. Ini dicapai menggunakan fungsi ‘*cv2.imread*’ dengan parameter kedua disetel ke 0, yang menandakan mode pembacaan gambar dalam skala abu-abu.

Setelah gambar dibaca, kode tersebiut menghasilkan negatif dari gambar tersebut. Proses ini dilakukan dengan mengurangkan nilai setiap piksel dari gambar dari 255, yang efektif mengubah area terang menjadi gelap dan sebaliknya. Ini adalah transformasi sederhana namun efektif yang sering digunakan dalam pengolahan citra.

Kemudian, kode tersebut menggunakan ‘*cv2.imshow*’ untuk menampilkan dua jendela, satu menunjukkan gambar asli dan yang lain menunjukkan negatif dari gambar tersebut. Ini memungkinkan pengguna untuk membandingkan gambar asli dengan versi negatifnya secara visual.

Untuk memastikan bahwa program tidak berakhir menutup jendela secara langsung setelah gambar ditampilkan, ‘*cv2.waitKey(0)*’ digunakan. Ini membuat program menunggu hingga pengguna menekan sembarang tombol sebelum melanjutkan. Setelah input diterima, semua jendela yang dibuka oleh *OpenCV* akan ditutup menggunakan ‘*cv2.destroyAllWindows()*’.

Secara keseluruhan, kode ini adalah contoh yang bagus dari beberapa operasi dasar pengolahan citra dengan OpenCV, yaitu membaca, memodifikasi nilai piksel, dan menampilkan gambar. Ini bisa dijadikan dasar untuk eksplorasi lebih lanjut dalam pengolahan citra, termasuk tetapi tidak terbatas pada filtrasi, deteksi tepi dan segmentasi gambar.

1. **KESIMPULAN**

Dapat disimpulkan bahwa kode ini merupakan contoh dasar namun penting dalam dunia pengolahan citra menggunakan OpenCV. Kode ini menunjukkan bagaimana sebuah gambar dapat dibaca, diolah dan ditampilkan dengan mudah menggunakan fungsi-fungsi yang tersedia dalam *library OpenCV.* Proses pembuatan negatif dari gambar adalah salah satu transformasi citra dasar yang efektif untuk memahami konsep manipulasi nilai piksel.