LAPORAN

PENGOLAHAN CITRA DIGITAL

Pertemuan : 13

TUGAS

NAMA : SYAHRUL FIKRI

NIM : 1207070121

KELAS : TKK

HARI, TANGGAL : Kamis, 8 Juni 2023

WAKTU : 13.00 – Selesai

DOSEN : RIN RIN NURMALASARI, S.Pd., M.T.



TEKNIK ELEKTRO

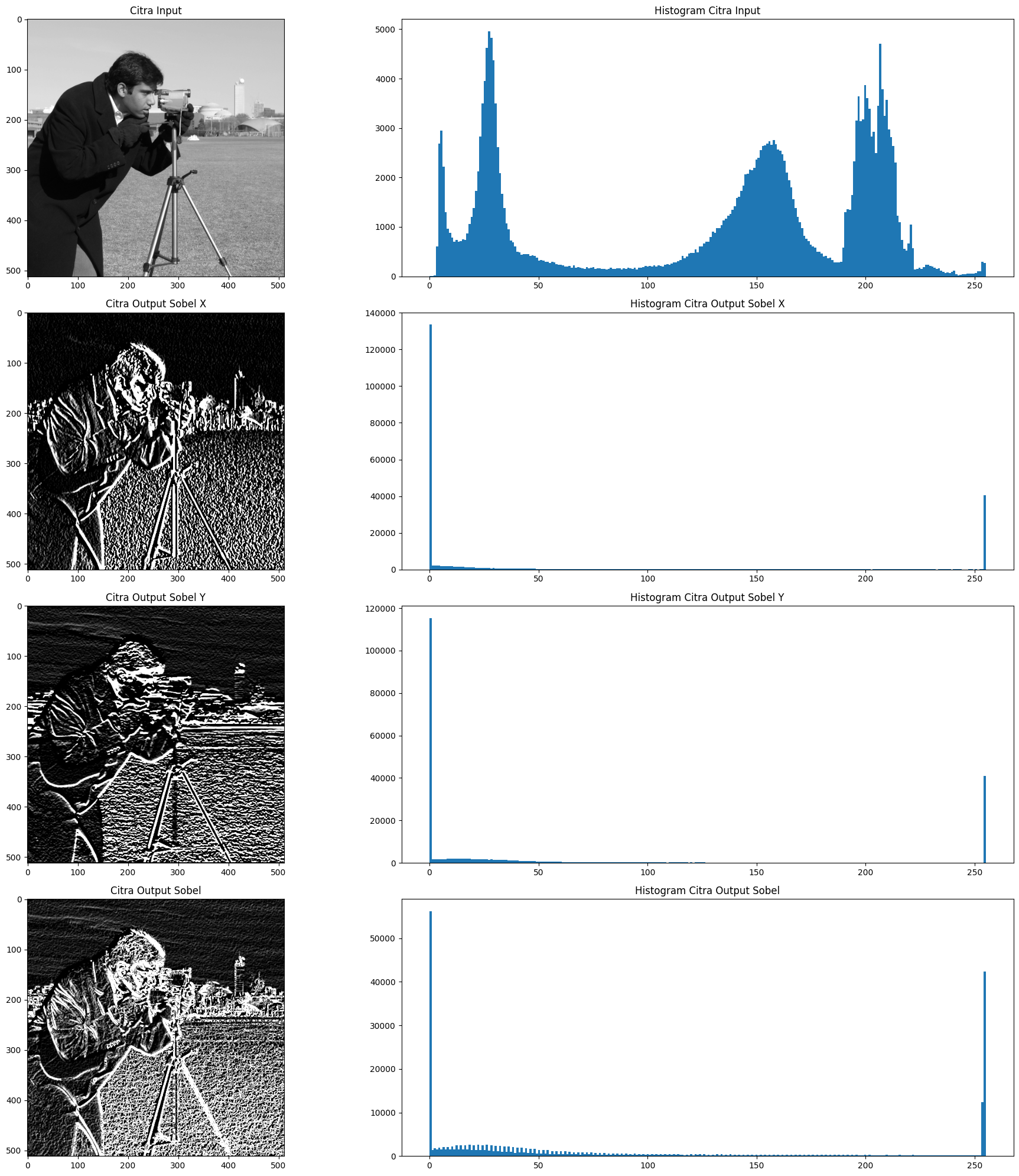
FAKULTAS SAINS DAN TEKNOLOGI

UIN SUNAN GUNUNG DJATI BANDUNG

2023

**PENGOLAHAN CITRA DIGITAL**

1. **FILTER SOBEL**



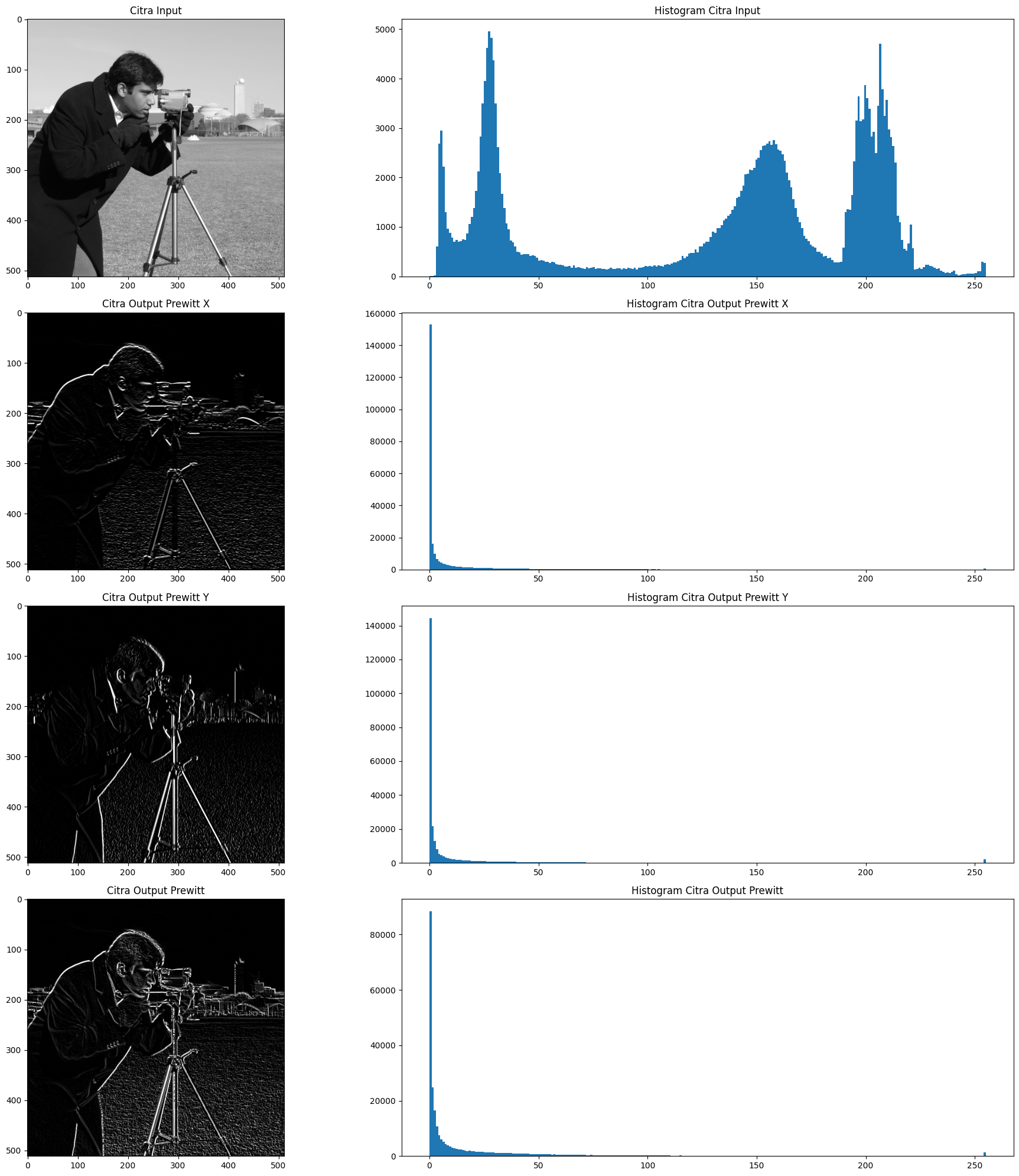
Operasi citra diatas merupakan operasi untuk edge detection menggunakan filter sobel yang dapat merespon secara maksimal tepi terhadap kisi piksel berdasarkan gradient. Pada operasi tersebut menggunakan bahasa pemrograman pyhton untuk menjalankan perintah terhadap pengolahan citra tersebut. Untuk dapat mengolah citra tersebut dengan operasi filter sobel perlu memanfaatkan salah satu *package* dalam *library* Opencv yaitu “cv2.Sobel”. Dengan menggunakan fungsi tersebut maka komputasi gaussian perhitungan konvolusi kernel dapat dilakukan untuk menghasilkan pendeteksian tepinya.

Data citra yang digunakan dalam operasi diatas merupakan data citra yang didapat dengan membaca atau meng-*load* data citra yang terdapat dalam library skimage. Citra tersebut dipilih karena dianggap memiliki berbagai macam objek dalam satu gambar sehingga dapat dilakukan pengujian untuk mendeteksi tepian setiap objek yang ada. Pada pengolahan citra ini ukuran kernel yang dipilih adalah 5(lima) yang dimaksudkan untuk mendapatkan konvolusi 5x5 dari piksel kernel terdekatnya menjadi lebih besar lagi sehingga tepian setiap objek terlihat lebih jelas.

Pada citra pertama atau terletak paling atas merupakan data citra normalnya yang belum dilakukan edge detection dan dengan format citra grayscale untuk dapat memudahkan pendeteksian tepi kemudian untuk persebaran histogram citranya tampak normal sesuai dengan gambarnya. Pada citra kedua atau terletak dibawah citra normal merupakan citra hasil filter sobel dalam gradien x yang dimana citra dideteksi dari sisi horizontal dan dinyatakan dalam format hitam putih sehingga persebaran histogram tidak merata yang tersebar pada sisi kiri dan kanan saja. Pada citra ketiga atau citra dibawah citra filter sobel x merupakan citra hasil filter sobel dalam gradien y yang dimana citra dideteksi dari sisi vertikal dan dinyatakan dalam format hitam putih sehingga persebaran histogram tidak merata yang tersebar pada sisi kiri dan kanan saja. Pada citra keempat atau citra paling bawah merupakan citra filter sobel hasil konvolusi atau penjumlahan dari kedua gradien yaitu gradien x dan gradien y yang dimana citra dideteksi dari sisi horizontal dan vertikal dan dalam format hitam putih sehingga persebaran histogram tidak merata yang tersebar pada sisi kiri dan kanan saja dengan hasil konvolusi histogram gradien x dan gradien y.

Sehingga dari pendeteksian citra tersebut untuk operasi turunan pertama atau filter sobel ini membutuhkan matriks kernel yang akan dideteksi terhadap arah gradien filter baik itu dari arah horizontal atau pada gradien x dan dari arah vertikal atau pada gradien y dengan hasil konvolusi pendeteksian kedua gradien tersebut menghasilkan pendeteksian tepi yang lebih jelas lagi terhadap objeknya dengan pendeteksian lebih baik menggunakan data citra dengan format grayscale supaya pendeteksian lebih mudah dilakukan dan tidak terjadinya kesalahan pendeteksian akibat dari banyaknya warna yang ada.

1. **FILTER PREWITT**

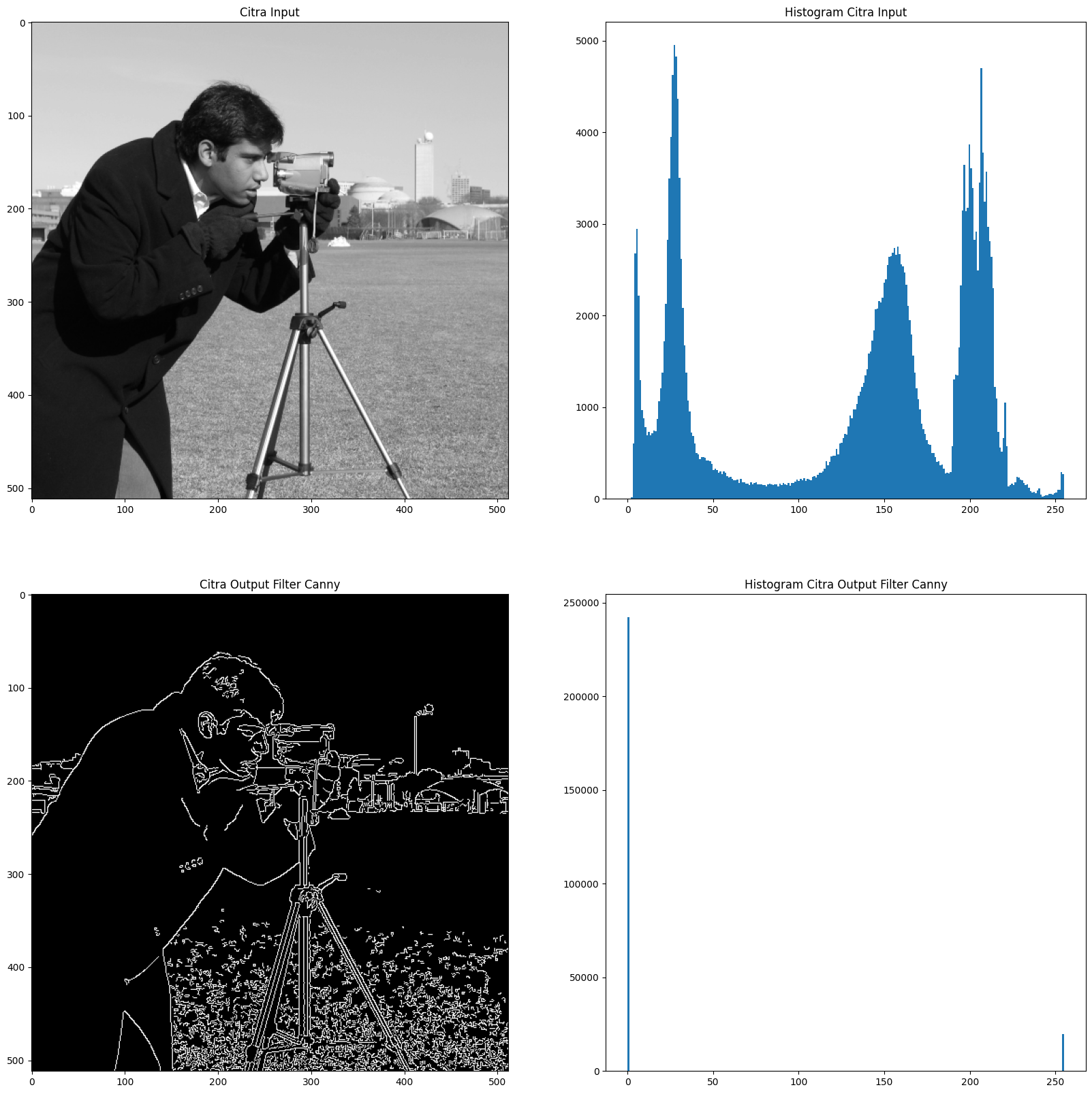


Operasi pengolahan citra tersebut merupakan pengolahan citra dengan menggunakan filter prewitt yang mendeteksi tepi citra terhadap kisi piksel berdasarkan gradien. Pada operasi tersebut menggunakan bahasa pemrograman pyhton untuk menjalankan perintah terhadap pengolahan citra tersebut. Dalam filter prewitt ini tidak terdapat operatur fungsi khusus yang digunakan untuk melakukan komputasi konvolusi matriks pixelnya hanya saja fungsi prewitt perlu didefinisikan terhadap gradiennya dan untuk mengolah data citranya perlu menggunakan salah satu *package* dalam *library* Opencv yaitu “cv2.filter2D” yang dimanfaatkan untuk melakukan pengolahan terhadap citra dua dimensi tersebut.

Data citra yang digunakan dalam operasi diatas merupakan data citra yang didapat dengan membaca atau meng-*load* data citra yang terdapat dalam library skimage. Citra tersebut dipilih karena dianggap memiliki berbagai macam objek dalam satu gambar sehingga dapat dilakukan pengujian untuk mendeteksi tepian setiap objek yang ada. Untuk fungsi kernel yang digunakan dalam filter ini memanfaatkan matriks kernel 3x3 yang telah didefinisikan sebelumnya terhadap gradiennya.

Pada citra pertama atau terletak paling atas merupakan data citra normalnya yang belum dilakukan edge detection dan dengan format citra grayscale untuk dapat memudahkan pendeteksian tepi kemudian untuk persebaran histogram citranya tampak normal sesuai dengan gambarnya. Pada citra kedua atau citra yang terletak dibawah citra normal merupakan citra hasil operasi filter prewitt yang dideteksi menggunakan gradien x sehingga pendeteksian citra dilakukan secara horizontal dan dinyatakan dalam format hitam putih sehingga persebaran histogram tidak merata yang tersebar pada sisi kiri dan kanan saja. Pada citra ketiga atau citra yang terletak dibawah citra filter prewitt gradien x merupakan citra hasil operasi filter prewitt yang dideteksi menggunakan gradien y sehingga pendeteksian citra dilakukan secara vertikal dan dinyatakan dalam format hitam putih sehingga persebaran histogram tidak merata yang tersebar pada sisi kiri dan kanan saja. Pada citra keempat atau citra yang terletak paling bawah merupakan citra hasil filter prewitt dengan penjumlahan dari pendeteksian gradien x dan pendeteksian gradien y dan dinyatakan dalam format hitam putih sehingga persebaran histogram tidak merata yang tersebar pada sisi kiri dan kanan saja dengan hasil konvolusi histogram gradien x dan gradien y.

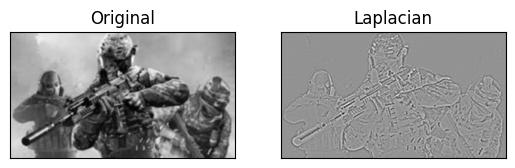
Sehingga dari pendeteksian citra menggunakan operasi filter prewitt merupakan pendeteksian tepi yang mirip dengan operasi filter sobel karena mendeteksi tepian berdasarkan gradien horizontal dan gradien vertikal namun proses pendeteksian lebih cepat karena bagian yang bernilai nol tidak perlu diproses. Operasi filter prewitt ini merupakan komputasi turunan pertama dan filter ini cocok untuk citra yang memiliki kontras tinggi dan derau yang sangat kecil serta lebih baik menggunakan citra dengan format grayscale.

1. **FILTER CANNY**

Operasi pengolahan citra tersebut merupakan pendeteksian tepian citra menggunakan operasi filter canny yang mendeteksi dengan melokalisasi titik-titik tepi. Pada operasi tersebut menggunakan bahasa pemrograman pyhton untuk menjalankan perintah terhadap pengolahan citra tersebut. Dalam pemrosesan dengan operasi filter canny membutuhkan salah satu *package* dari *library* Open CV yaitu “cv2.Canny” yang mendeteksi tepi multi-tahap berdasarkan turunan gaussian untuk menghitung intensitas gradien dan mengurangi efek noise yang ada pada gambar. Kemudian, tepi potensial ditipiskan menjadi kurva 1(satu) piksel dengan menghilangkan piksel non-maksimum dari besaran gradien. Sehingga, piksel tepi disimpan atau dihapus menggunakan ambang batas histeresis pada besarnya gradien.

Pada citra pertama atau terletak paling atas merupakan data citra normalnya yang belum dilakukan edge detection dan dengan format citra grayscale untuk dapat memudahkan pendeteksian tepi kemudian untuk persebaran histogram citranya tampak normal sesuai dengan gambarnya. Pada kedua atau terletak paling bawah merupakan data citra hasil operasi filter canny yang mendeteksi tepian citra secara optimal karena hasil pengoperasian berbagai macam tahap diantara parameternya adalah lebar gaussian, batas ambang, dan batas histeresis pada citra tersebut dan hasil histogram yang ditampilkan merupakan data citra dengan banyaknya piksel tepi yang terdeteksi dari pengurangan berbagai macam noise dengan format gambar hitam putih sehingga persebaran hanya pada sisi kiri dan kanan saja.

1. **FILTER TURUNAN KEDUA**

****

Operasi pengolahan citra tersebut merupakan pendeteksian citra menggunakan operasi filter laplacian yang merupakan operasi dengan turunan kedua. Pada operasi tersebut menggunakan bahasa pemrograman pyhton untuk menjalankan perintah terhadap pengolahan citra tersebut. Dalam pemrosesan pendeteksian citra ini memanfaatkan beberapa *package* dari *library* Opencv yaitu “cv2.cvtColor” untuk melakukan modifikasi warna citra, “cv2.GaussianBlur” untuk menghilangkan noise yang dimiliki gambar, dan “cv2.Laplacian” yang package untuk dapat melakukan operasi filter laplacian dengan komputasi turunan kedua.

Pada gambar pertama atau sebelah kiri merupakan data citra awal sebelum dilakukan operasi pendeteksian tepi filter laplacian dengan format citra yang telah diubah menjadi grayscale untuk dapat diproses oleh filter. Pada gambar kedua atau sebelah kanan merupakan data citra hasil operasi pendeteksian citra dengan operasi filter laplacian yang telah dilakukan penghilangan noise sebelumnya.

Sehingga pada Kernel filter Laplacian biasanya berisi nilai negatif dalam pola silang, berpusat di dalam larik. Sudutnya adalah nilai nol atau positif. Nilai pusat dapat berupa negatif atau positif. Perhitungan derivatif ini akan menemukan tempat-tempat digambar dimana intensitas piksel berubah secara drastis. Kemudian, untuk turunan transisi hitam ke putih adalah positif dan sebaliknya untuk turunan transisi putih ke hitam adalah negatif sehingga memerlukan tipe data keluaran tinggi supaya setiap data hasil konversi tidak ada yang hilang.

1. **Github Link Program**

<https://github.com/syahrulfikri/Pengolahan-Citra-Digital/tree/2c55c2ffdb2d5777bc2de6ed244f24a783c9349a/1207070121_Syahrul%20Fikri_PCD_Prak.9>