

Nama : Siti Nurul Qamariyah
Prodi : Ti D
Semester : V
Dosen pengampu : Bapak Thahiruddin

JAWABAN

1) Dimana foto asli tersebut menjadi gambar negative



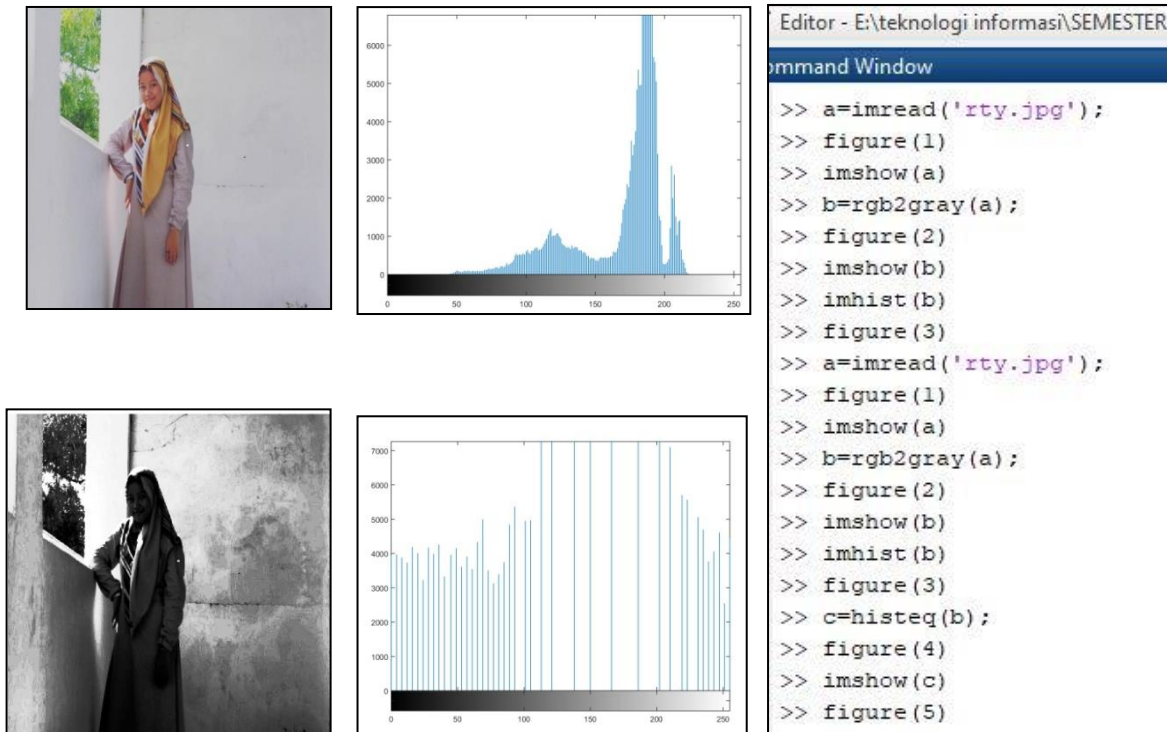
```
Editor - E:\teknologi informasi\SEMESTER 5\pengelo  
Command Window  
>> I = imread('rty.jpg');  
>> J = imcomplement(I);  
>> imshow(I), figure, imshow(J)  
fx >> |
```

2) Foto tersebut menjadi Grayscale menggunakan metode luminosity



```
Editor - E:\teknologi informasi\SEMESTER 5  
Command Window  
>> I = imread('rty.jpg');  
>> J = rgb2gray(I);  
>> figure, imshow(J);  
fx >> |
```

- 3) Gambarkan Histogram dari foto tersebut kemudian lakukan Histogram equalization pada foto tersebut!



- 4) Lakukan konvolusi dengan menggunakan beberapa filter berikut dan berikan penjelasannya:

a. Lowpass Filter

Low Pass Filter atau sering disingkat dengan LPF adalah Filter atau Penyaring yang melewatkan sinyal Frekuensi rendah dan menghambat atau memblokir sinyal Frekuensi tinggi. Dengan kata lain, LPF akan menyaring sinyal frekuensi tinggi dan meneruskan sinyal frekuensi rendah yang diinginkannya.



Koding untuk lowpass filter :

```
Editor - C:\Program Files\MATLAB\MATLAB Product
Command Window
O = B;
for i = 2:k-1
    for j = 2:l-1
        I = double(B(i-1:i+1,j-1:j+1));
        O(i,j) = sum(sum(I.*mask));
    end
end
figure(3)
imshow(O)
>> A = imread('rty.tiff');
figure(1)
imshow(A)
B = rgb2gray(A);
figure(2)
imshow(B)
mask = 1/9*ones(3);
[k,l] = size(B);
O = B;
for i = 2:k-1
    for j = 2:l-1
        I = double(B(i-1:i+1,j-1:j+1));
        O(i,j) = sum(sum(I.*mask));
    end
end
figure(3)
imshow(O)
```

b. High-boost Filter

Bila kita ingin menunjukkan komponen yang menunjukkan detail dari citra tanpa menghilangkan komponen frekuensi rendahnya seperti halnya kita menggunakan highpass filter dalam kasus sharpening atau penajaman. Kita bisa menggunakan high-boost filtering. Dengan menggunakan filter ini kita bisa mempertajam detail dari citra tapi frekuensi rendah tidak dihilangkan.

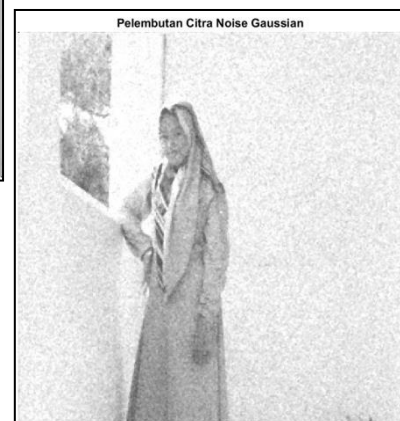
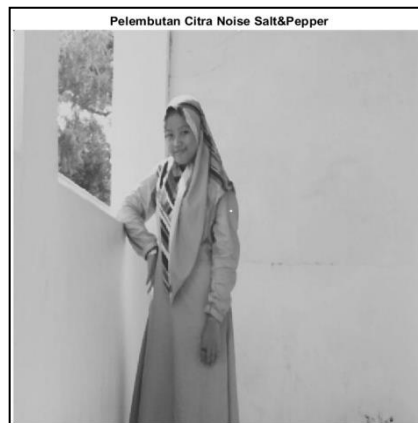


Koding untuk High-boost Filter :

```
Editor - C:\Program Files\MATLAB\MATLAB Production
Command Window
>> A = imread('rty.tiff');
H = [0,-1,0;-1,5,-1;0,-1,0];
B = imfilter(A,H, 'replicate');
imshowpair(A,B, 'montage');
fx >> |
```

c. Median Filter

Filter median adalah teknik pemfilteran digital nonlinear yang biasa dipakai untuk menghilangkan derau pada citra atau sinyal. Pengurangan derau adalah langkah praolah untuk memperbaiki hasil pada pengolahan selanjutnya (misalnya deteksi pinggiran pada citra). Filter median dipakai luas dalam pengolahan citra digital karena dapat menghilangkan derau sekaligus menjaga pinggiran dalam keadaan tertentu. Selain itu, filter ini juga dipakai dalam pengolahan sinyal digital.



Koding untuk median filter :

```
Command Window

>> gambar=imread('rty.tiff');
gray=rgb2gray(gambar);
noise=imnoise(gray,'Salt & Pepper',0.03);
noise2=imnoise(gray,'Gaussian',0.2);
awal=medfilt2(gray);
k=medfilt2(noise);
k2=medfilt2(noise2);
figure,imshow(awal),title('Pelembutan Citra Awal');
figure,imshow(k),title('Pelembutan Citra Noise Salt&Pepper');
figure,imshow(k2),title('Pelembutan Citra Noise Gaussian');
```

5) Morfologi citra

Operasi morfologi citra merupakan suatu proses yang bertujuan untuk mengubah bentuk objek pada citra asli. Proses tersebut dapat dilakukan pada citra grayscale maupun citra biner.

Jenis-jenis operasi morfologi di antaranya adalah dilasi, erosi, closing, dan opening. Secara berurutan, persamaan yang digunakan untuk masing-masing operasi yaitu:

$$\begin{aligned} A \oplus B & \dots\dots (1) \\ A \ominus B & \dots\dots (2) \\ A \bullet B & = (A \oplus B) \ominus B \dots\dots (3) \\ A \circ B & = (A \ominus B) \oplus B \dots\dots (4) \end{aligned}$$

di mana A adalah citra asli dan B adalah structuring element. Structuring element merupakan matriks operator yang dapat berbentuk garis, persegi, disk, diamond, dll.





Koding untuk Morfologi citra :

```
Editor - C:\Program Files\MATLAB\MATLAB
Command Window

>> I = imread('rty.tiff');
YCbCr = rgb2ycbcr(I);
figure, imshow(I);
Y = YCbCr(:,:,1);
Cb = YCbCr(:,:,2);
Cr = YCbCr(:,:,3);
figure, imshow(Y);
figure, imshow(Cb);
figure, imshow(Cr);
```