E/16/134
Herath H.M.Y.B

CO543 – Image Processing Lab03

1. Use the following inputs to write your own 2D convolution function. Verify your implementation with OpenCV cv2.filter2D() function.

Inputs

X=

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10

Y=

-1	-1	-1
0	0	0
1	1	1

Output from Own function

In this minus(-) values from final output are replaced with zeros. Because Images don't have (-) pixel values

Function: con2D(X, Y, padding=1)

```
[[ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [20. 30. 30. 30. 30. 30. 30. 20.]]
```

Output from **cv2.filter2D()** function

In this minus values are also in the output

```
[[-20. -30. -30. -30. -30. -30. -30. -20.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [ 20.  30.  30.  30.  30.  30.  30.  20.]]
```

2. Bilateral, Gaussian and Median Filtering

- **A bilateral filter with mask size 5 × 5 with appropriate values of sigma, set through experimentation.**

Input image

Output image from own funtion

Function: bilateral_fil(img, 200.0, 200.0)

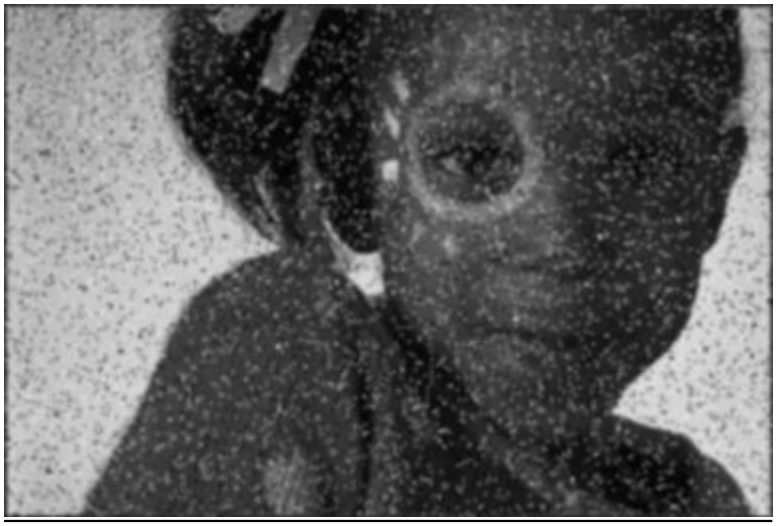
Output image from **cv2.bilateralFilter()** funtion

- **A Gaussian filter with mask size 5 × 5 appropriate values of σ .**

Input image

Output image from own funtion

Function: Gaussian_fil(img, 1)



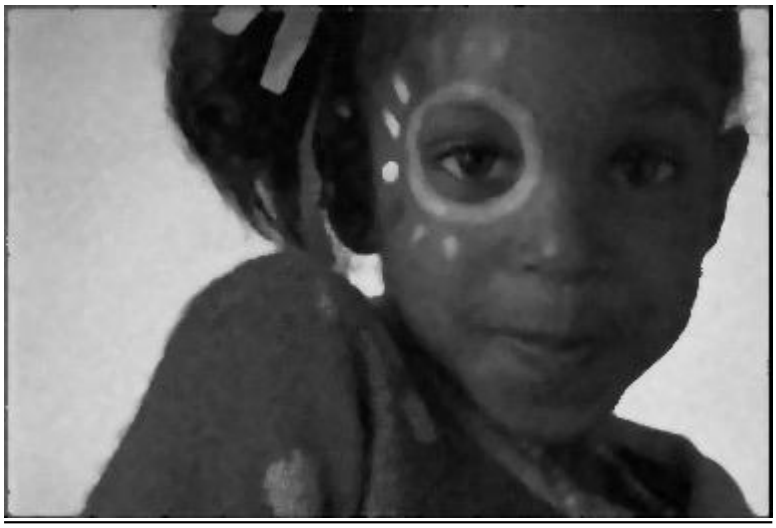
Output image from **cv2.GaussianBlur()** funtion

- **A median filter of appropriate window size.**

Input image

Output image from own funtion

Function: Median_fil(img, 5)



Output image from **cv2.medianBlur()** funtion