دانشگاه
خواجه نصیرالدین طوسی
K. N. Toosi University
of Technology

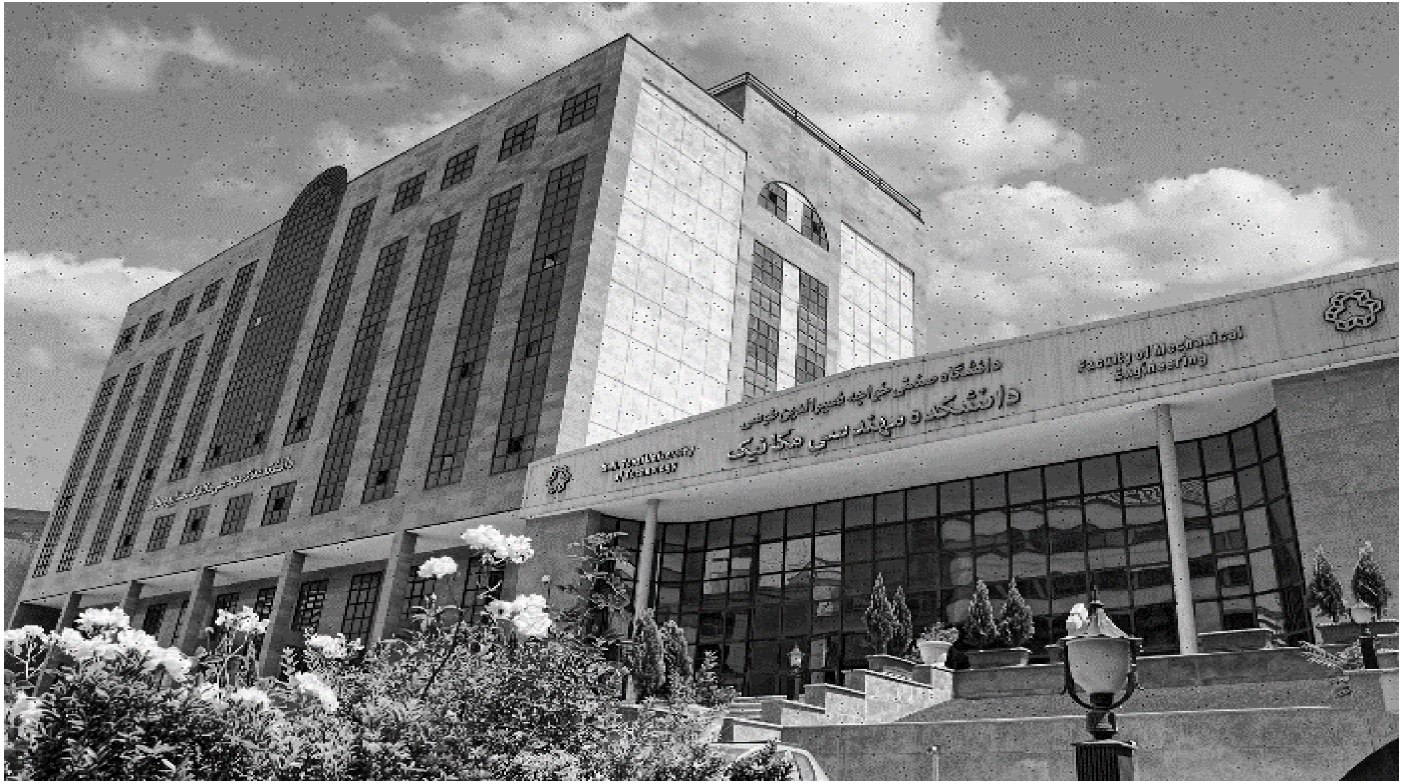
Computer Vision

Lecture 4: Image Filtering

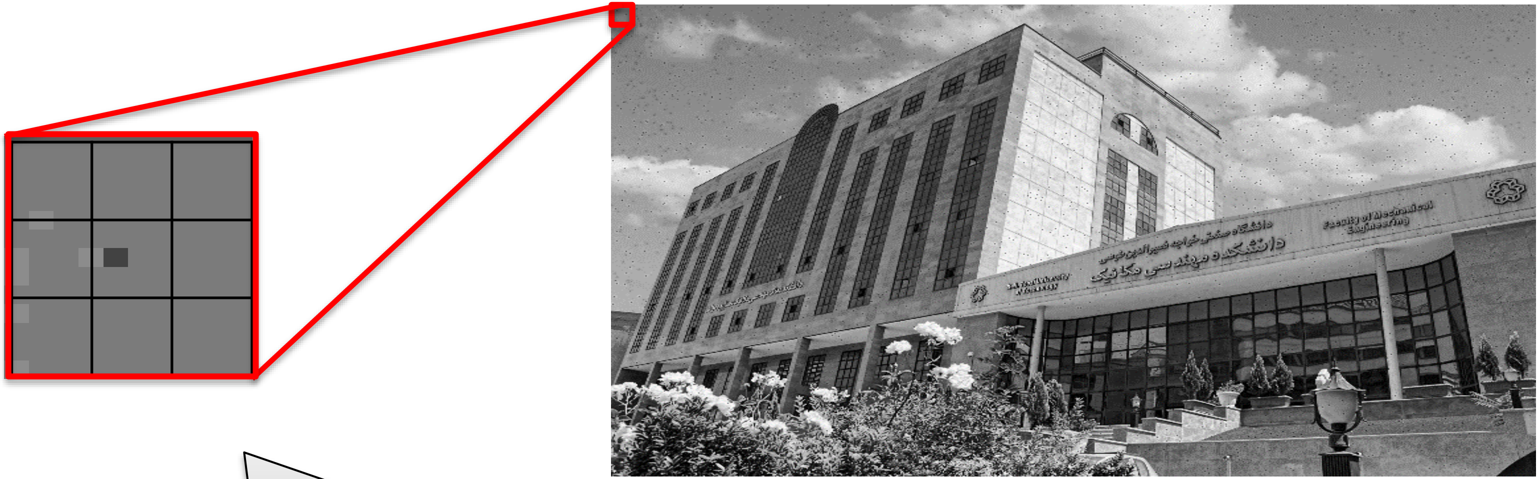
Dr. Esmaeil Najafi

MSc. Javad Khoramdel

How can we reduce effect of noise on an image?


دانشگاه صنعتی خواجه نصیرالدین طوسی
دانشکده مهندسی مکانیک
Faculty of Mechanical Engineering

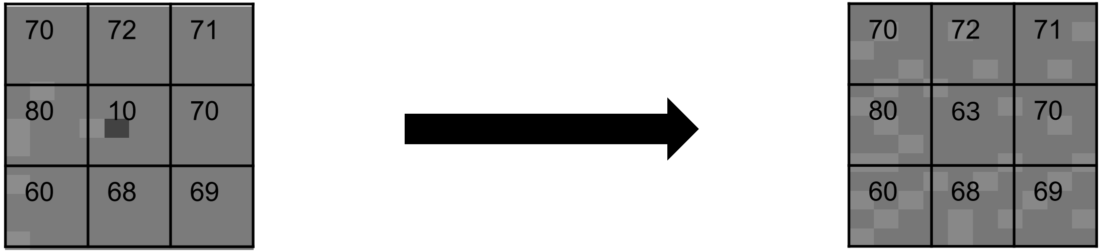
Let’s have a closer look



The pixel at the center of this small area is noisy, but the neighbors are not. So, what if we use the information from neighbors to reduce the effect of the noise?

How do we do that?

- By averaging!



70	72	71
80	10	70
60	68	69

70	72	71
80	63	70
60	68	69

$$middle\ pixel = 10$$

$$middle\ pixel = \frac{(70 + 72 + 71 + 80 + 10 + 70 + 60 + 68 + 69)}{9} = 63$$

Convolution

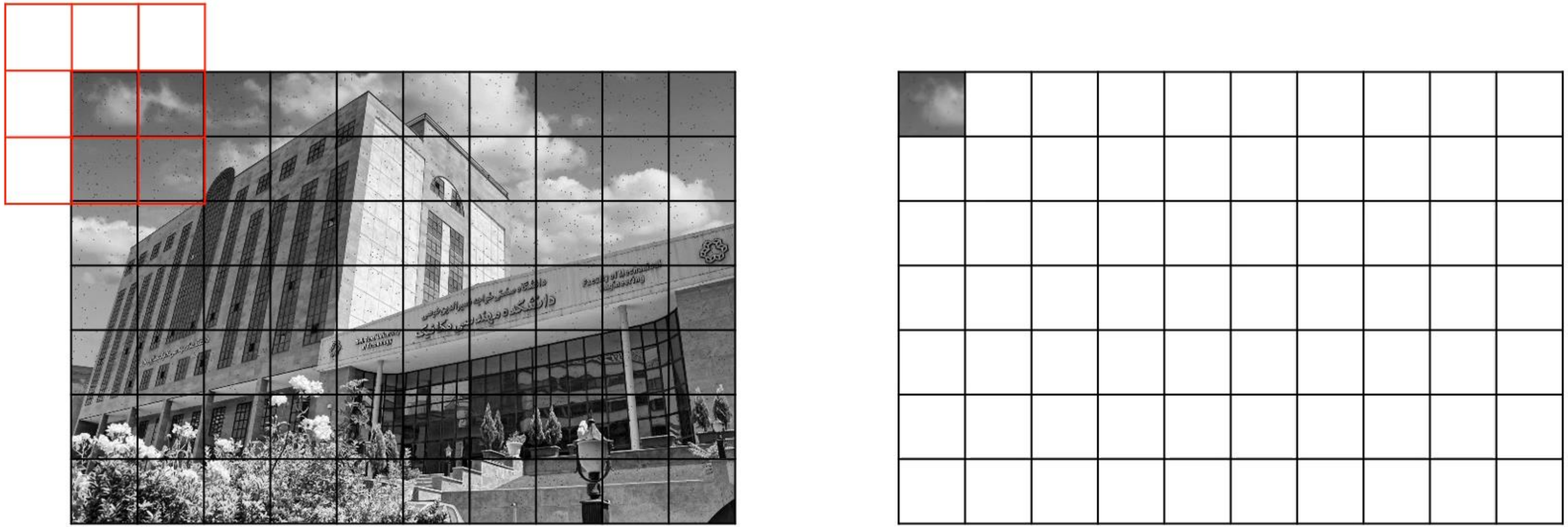
$$\underbrace{\overbrace{\frac{1}{3\times 3}\begin{array}{|c|c|c|}\hline 1\text{x} & 1\text{x} & 1\text{x}\\ \hline 1\text{x} & 1\text{x} & 1\text{x}\\ \hline 1\text{x} & 1\text{x} & 1\text{x}\\ \hline\end{array}}^{\text{Filter (also called kernel)}} * \begin{array}{|c|c|c|}\hline 70 & 72 & 71\\ \hline 80 & 10 & 70\\ \hline 60 & 68 & 69\\ \hline\end{array}}_{\text{Convolution}}$$

This is a linear operation.

$$= \frac{1}{9}(1\times 70 + 1\times 72 + 1\times 71 \\ +1\times 80 + 1\times 10 + 1\times 70 \\ +1\times 60 + 1\times 68 + 1\times 69) = 63$$

The output of the convolution operation is a single number. This single number will be the new value for the middle pixel.

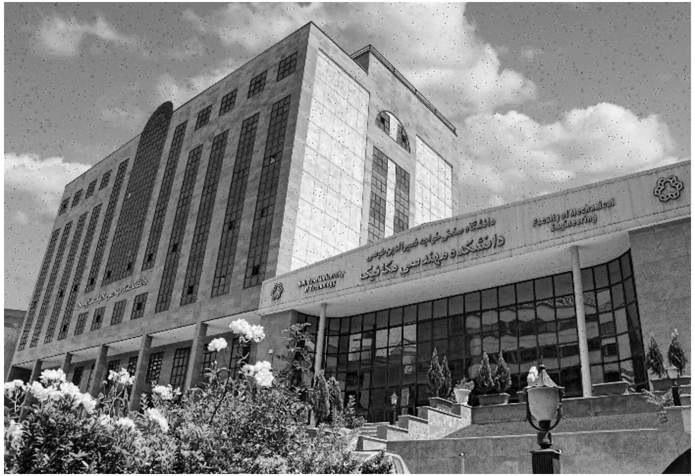
Filter slides across the image

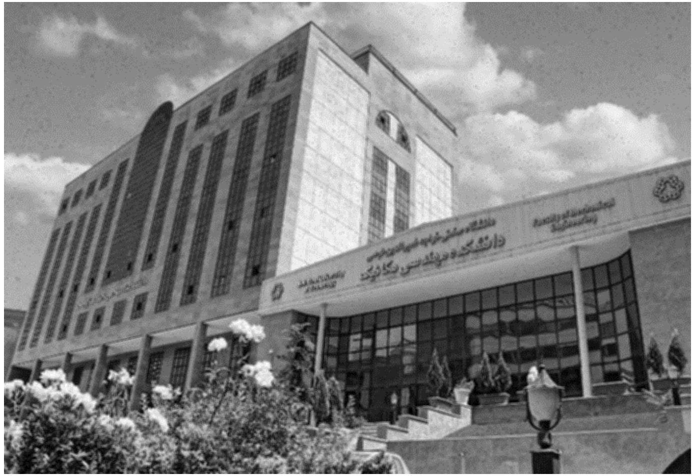


When the filter is at the borders, elements outside the image can be assumed to be zero.

Convolution on the entire image

$$\frac{1}{3 \times 3}\begin{array}{|c|c|c|}\hline 1x & 1x & 1x \\ \hline 1x & 1x & 1x \\ \hline 1x & 1x & 1x \\ \hline \end{array} \quad *$$





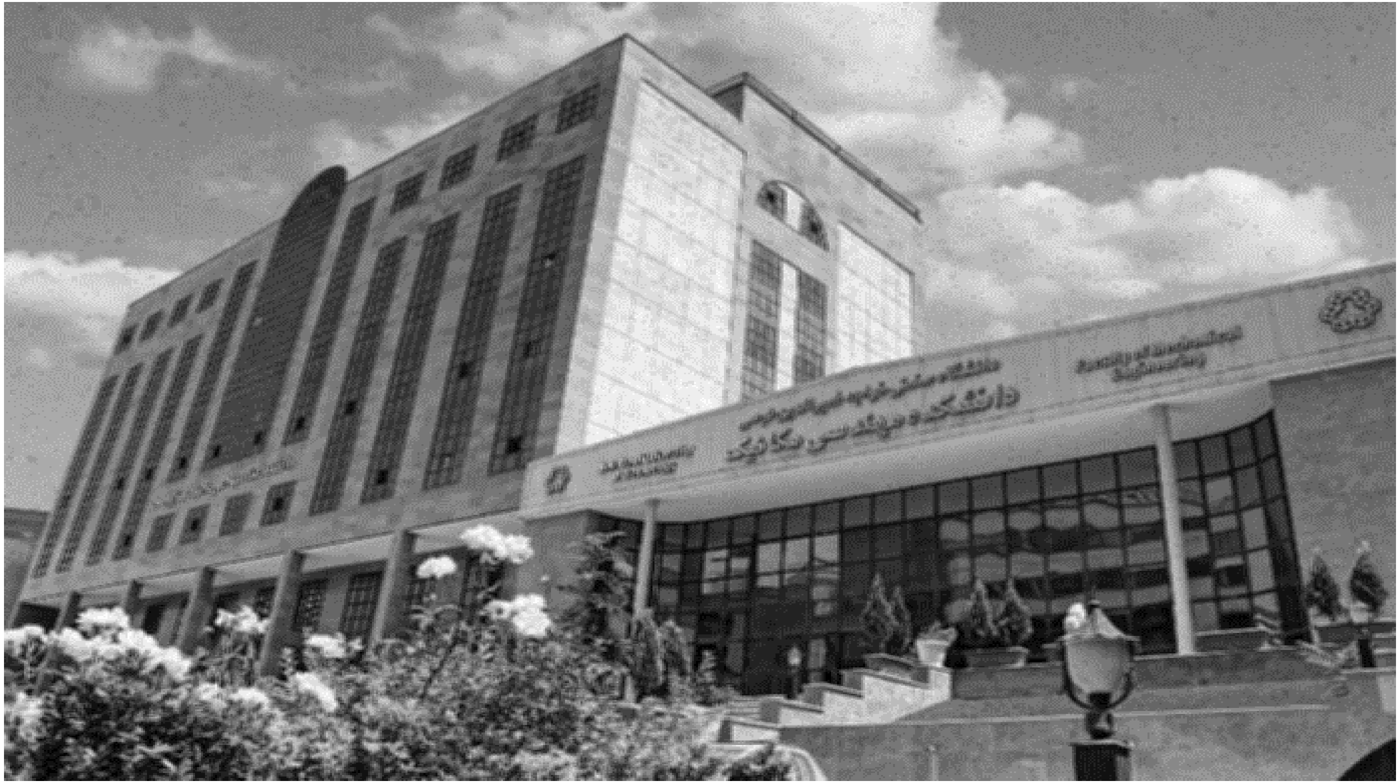
Filter Convolution Raw image

Filtered image
(Convolved with filter)

Filtered image

- Better, but still noisy!

- Let’s try bigger filters



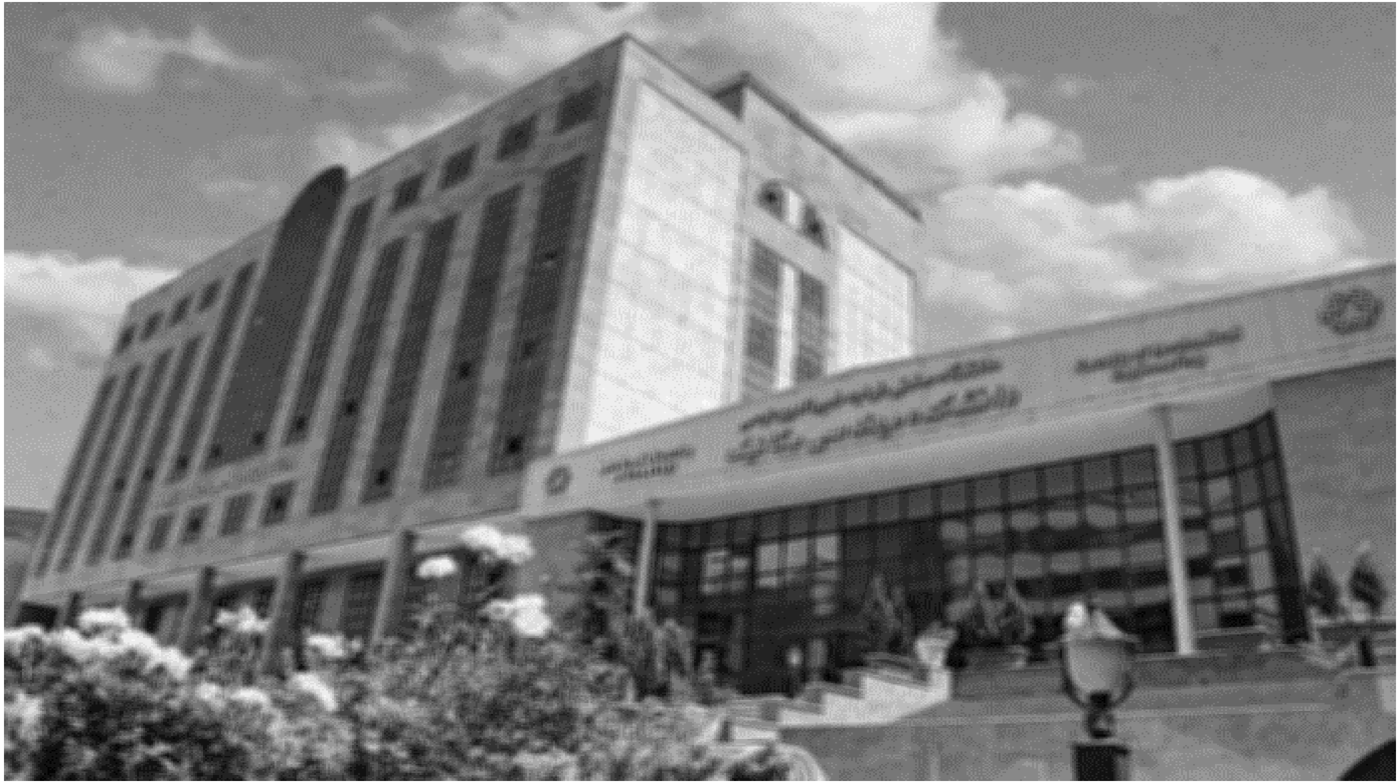
Filtered image

- 5x5 filter

$$\frac{1}{5 \times 5}$$

1x	1x	1x	1x	1x
1x	1x	1x	1x	1x
1x	1x	1x	1x	1x
1x	1x	1x	1x	1x
1x	1x	1x	1x	1x

- Much better!



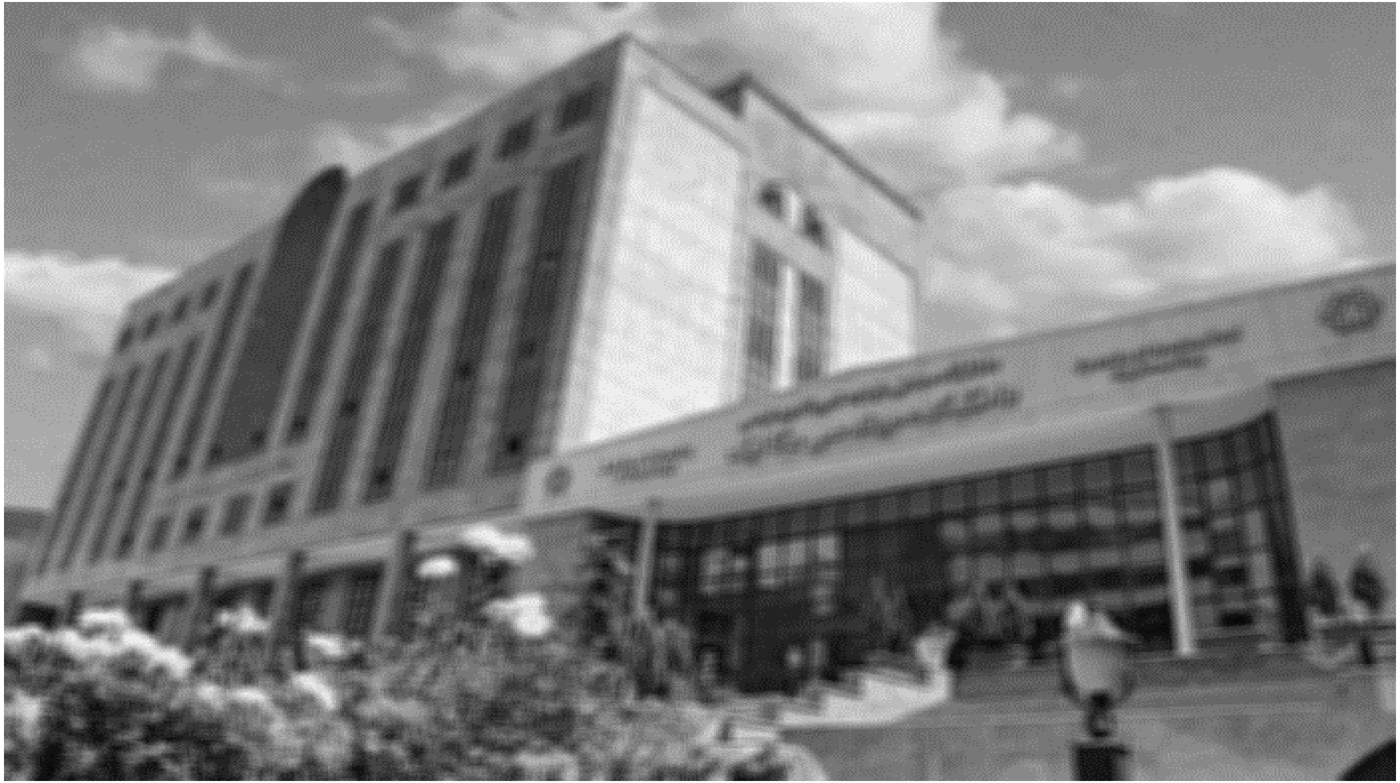
Filtered image

- 7x7 filter

$\frac{1}{7 \times 7}$

1x	1x	1x	1x	1x	1x	1x
1x	1x	1x	1x	1x	1x	1x
1x	1x	1x	1x	1x	1x	1x
1x	1x	1x	1x	1x	1x	1x
1x	1x	1x	1x	1x	1x	1x
1x	1x	1x	1x	1x	1x	1x
1x	1x	1x	1x	1x	1x	1x

- There is no noise, but the image looks very blurred.



This is called “box filter”

- In box filter, all elements are equal to: $\frac{1}{No.of\ elements\ in\ filter}$

$\frac{1}{3 \times 3}$

1x	1x	1x
1x	1x	1x
1x	1x	1x

3x3 box filter

$\frac{1}{5 \times 5}$

1x	1x	1x	1x	1x
1x	1x	1x	1x	1x
1x	1x	1x	1x	1x
1x	1x	1x	1x	1x
1x	1x	1x	1x	1x

5x5 box filter

$\frac{1}{m \times n}$

1x	1x	1x	1x	1x	1x	1x		1x
1x	1x	1x	1x	1x	1x	1x		1x
1x	1x	1x	1x	1x	1x	1x		1x
1x	1x	1x	1x	1x	1x	1x		1x
1x	1x	1x	1x	1x	1x	1x		1x
1x	1x	1x	1x	1x	1x	1x		1x
1x	1x	1x	1x	1x	1x	1x		1x
								1x
1x	1x	1x	1x	1x	1x	1x	1x	1x

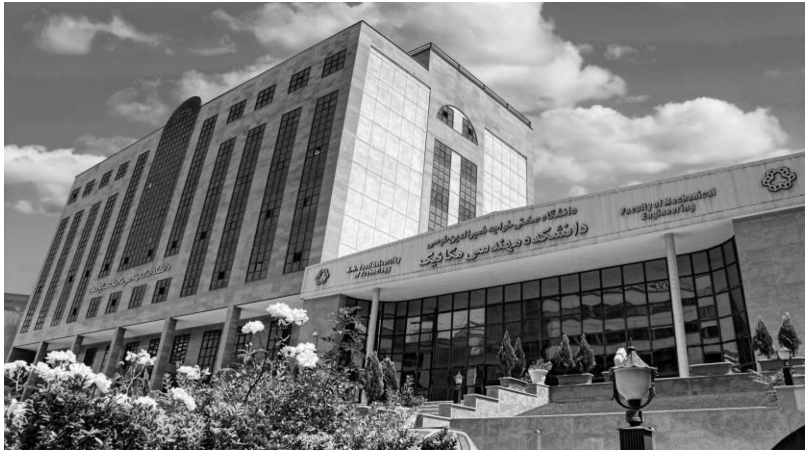
“m” rows

“n” columns

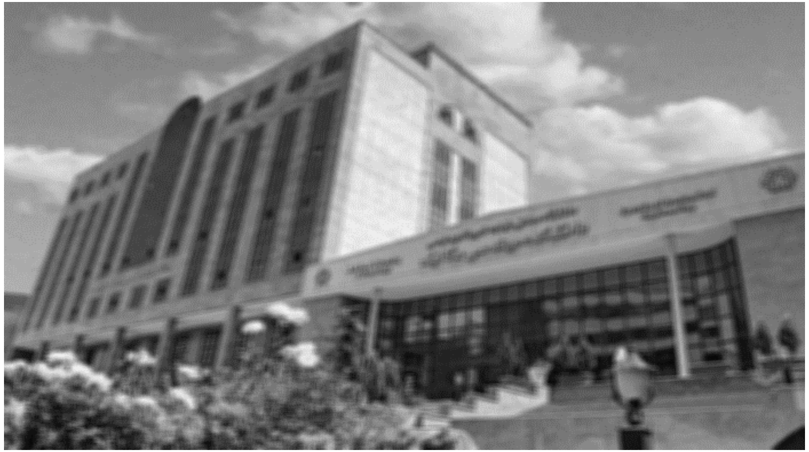
m x n box filter

Box filter is also known as “Mean filter” and “Averaging”. The filter size is a design choice.

Box filters smooth image


دانشگاه صنعتی خواجه نصیرالدین طوسی
دانشکده مهندسی مکانیک
K.N. Toosi University of Technology
Faculty of Mechanical Engineering

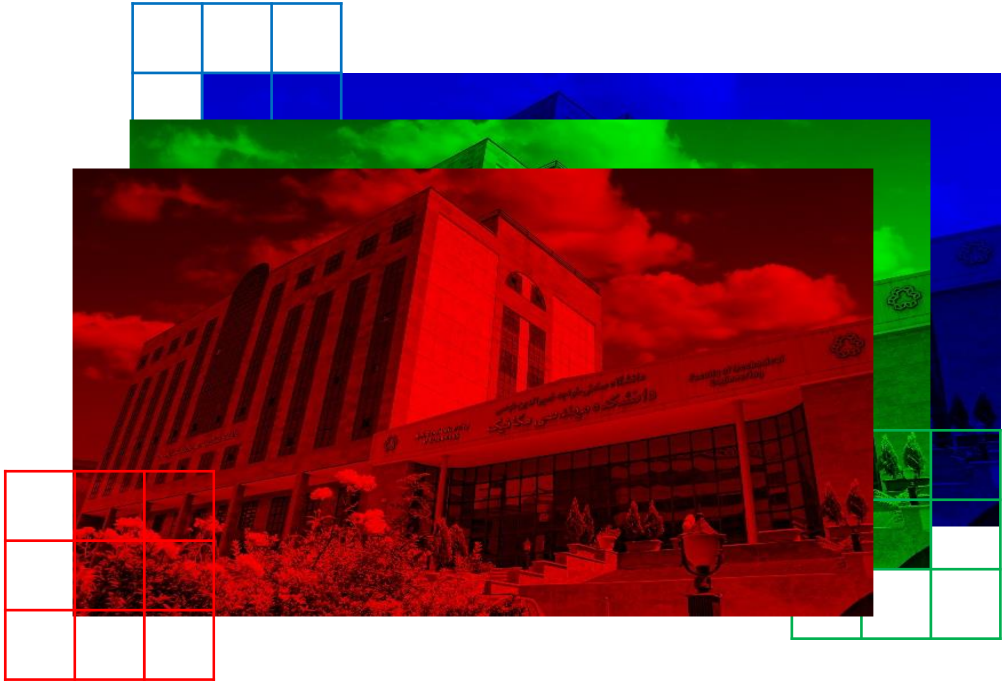
Raw image



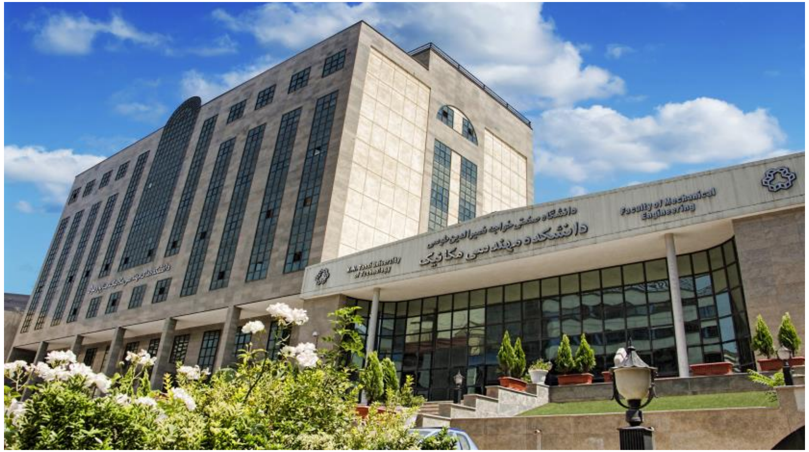
Filtered image

How about a color image?

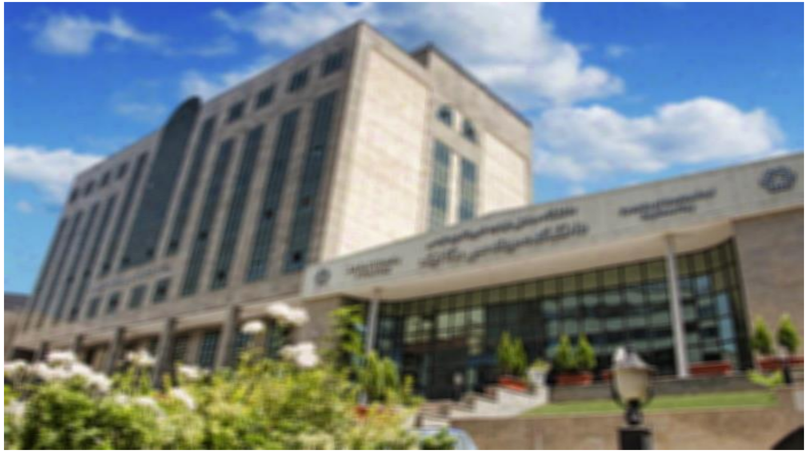
- For color images, we can filter each channel separately, then merge them to get the final result



How about a color image?



Raw image

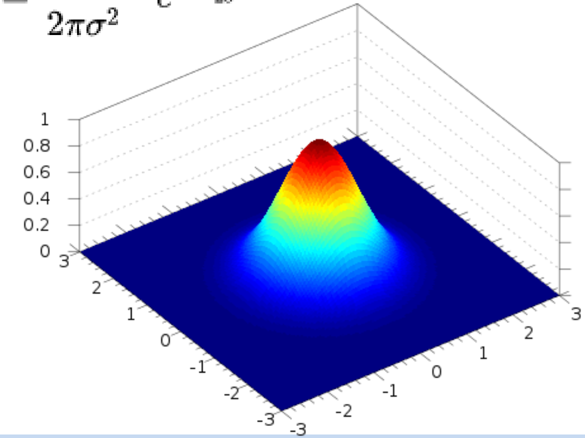


Filtered image

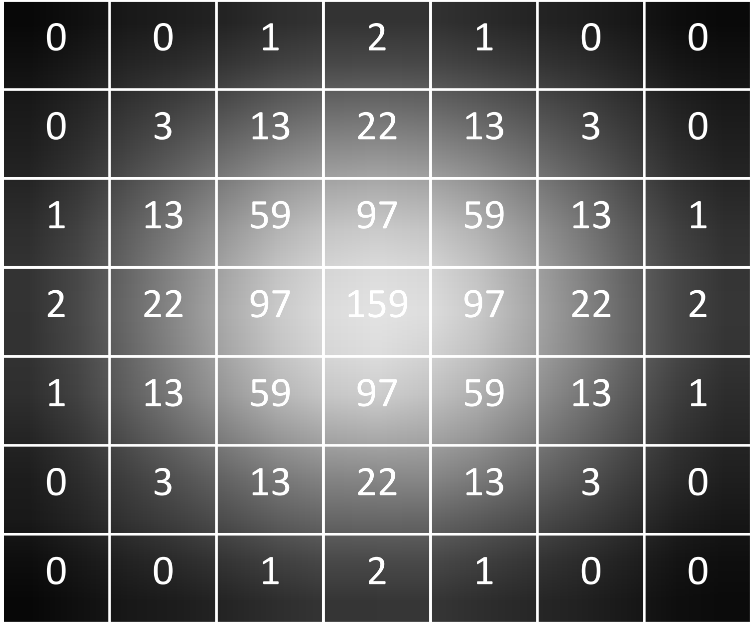
Gaussian filter, a smoothly weighted filter

- 2d Gaussian filter
  - For gaussian filter, we have a formula based on a given variance.
  - Filter size is also a design choice.

$$g(x,y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$



$\frac{1}{1023}$



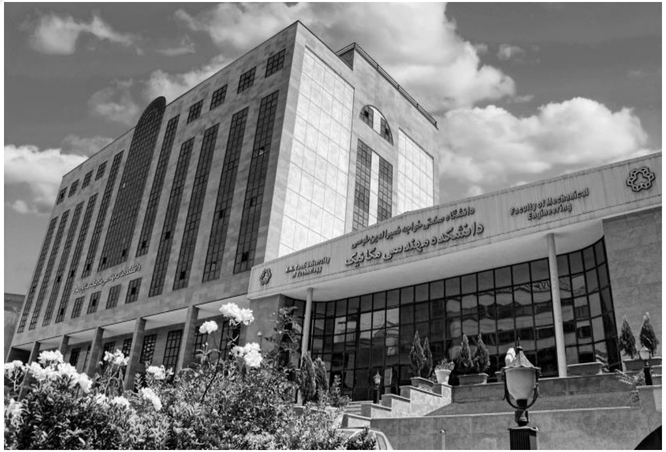
0	0	1	2	1	0	0
0	3	13	22	13	3	0
1	13	59	97	59	13	1
2	22	97	159	97	22	2
1	13	59	97	59	13	1
0	3	13	22	13	3	0
0	0	1	2	1	0	0

Visualized gaussian kernel. The elements in the middle have more weight than the corners; thus, we see them brighter.

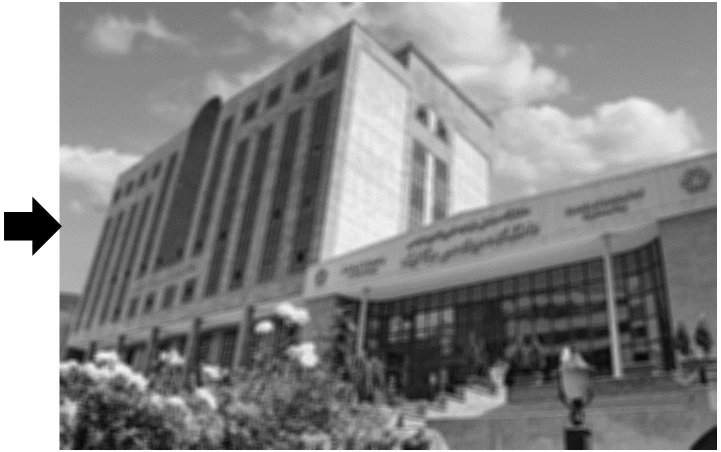
2D Gaussian filter

The convolution is the same as before. This is a linear operation between the kernel and the image.

$$\frac{1}{1023}\begin{bmatrix} 0 & 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 3 & 13 & 22 & 13 & 3 & 0 \\ 1 & 13 & 59 & 97 & 59 & 13 & 1 \\ 2 & 22 & 97 & 159 & 97 & 22 & 2 \\ 1 & 13 & 59 & 97 & 59 & 13 & 1 \\ 0 & 3 & 13 & 22 & 13 & 3 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 & 0 \end{bmatrix} *$$



➡



Filter Convolution Raw image

Filtered image
(Convolved with Gaussian filter)

Gaussian filter also smoothens the image.

Guess the filter!

- What happens if we convolve the image with the given filter?

0	0	0
0	1	0
0	0	0

*

➡

?

Filter Convolution Raw image Filtered image

Identity filter: Does nothing!

- What happens if we convolve the image with the given filter?

0	0	0
0	1	0
0	0	0

Filter

*

Convolution

Raw image

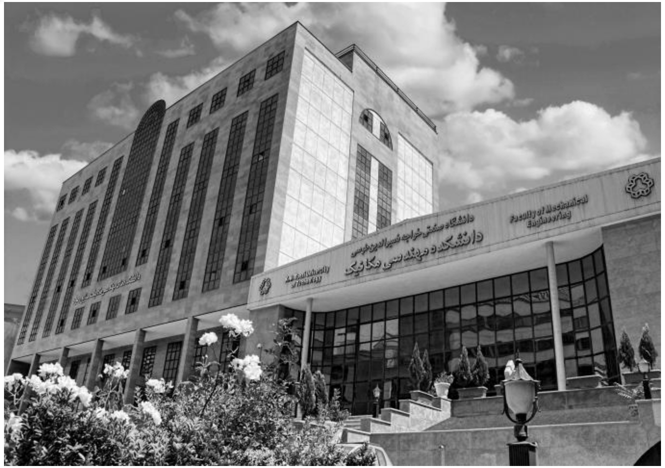
Filtered image

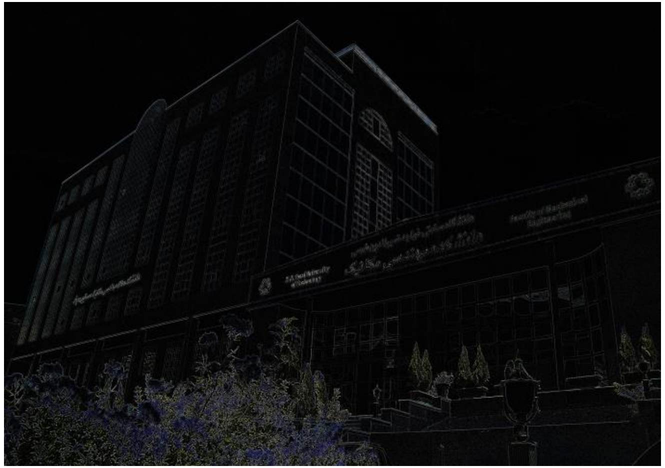
Highpass filter: finds edges

This filter is called 2D Laplacian. It can be shown that this filter calculates the second derivative of the image.

0	-1	0
-1	4	-1
0	-1	0

*





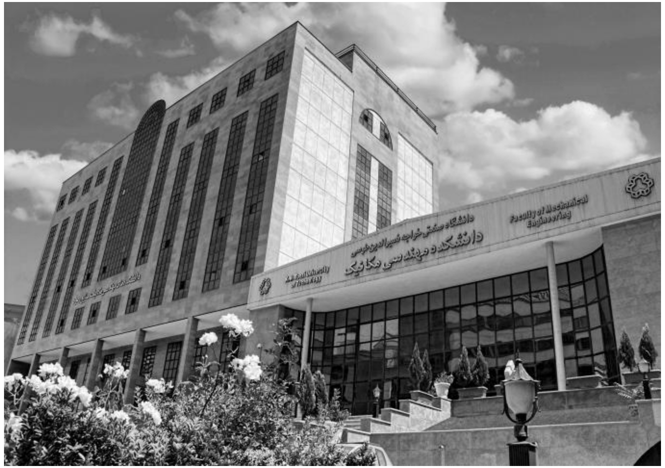
Filter Convolution Raw image Filtered image

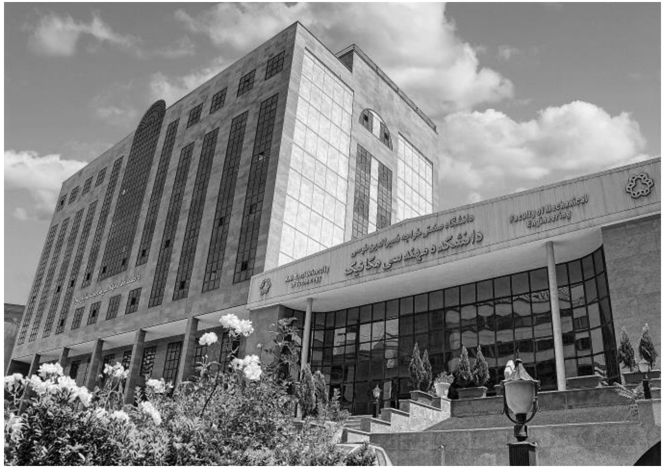
Sharpen filter: sharpens!

This filter is a combination of Laplacian and Identity filters.

0	-1	0
-1	5	-1
0	-1	0

*





Filter Convolution Raw image Filtered image

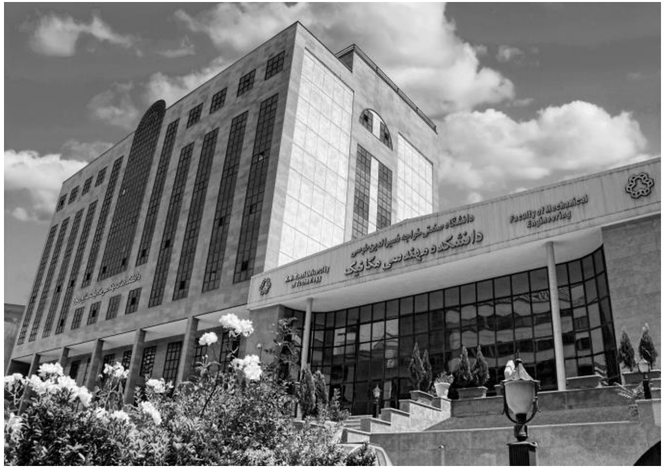
Emboss filter: stylin’

-2	-1	0
-1	1	1
0	1	2

Filter

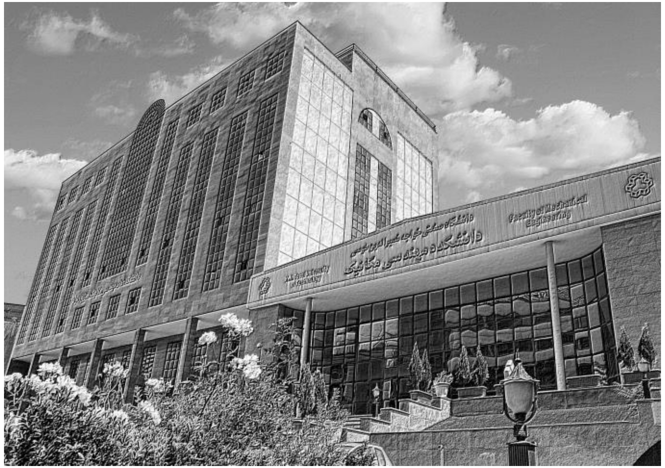
*

Convolution



Raw image

The artistic effect on the image is obvious!



Filtered image

Sobel filters: edges and…

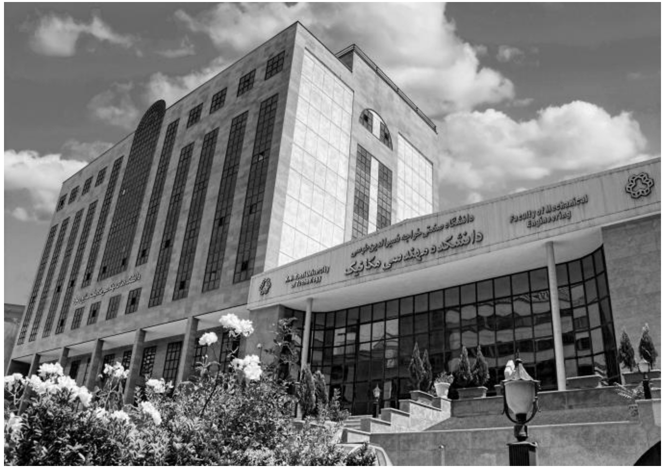
This is Sobel-Y filter. This filter calculates the gradient of the image in Y-direction.

-1	-2	-1
0	0	0
1	2	1

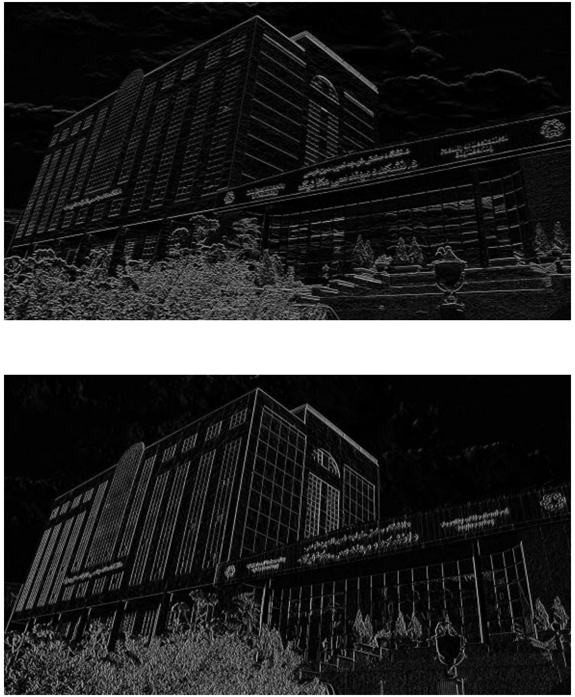
*

-1	0	1
-2	0	2
-1	0	1

This is Sobel-X filter. This filter calculates the gradient of the image in X-direction.



The Sobel-Y mostly finds the vertical edges.



The Sobel-X mostly finds the horizontal edges.

Filters Convolution Raw image Filtered images

Sobel filters: edges and gradients

- With Sobel-X and Sobel-Y, we have the X & Y components of the gradient vector.

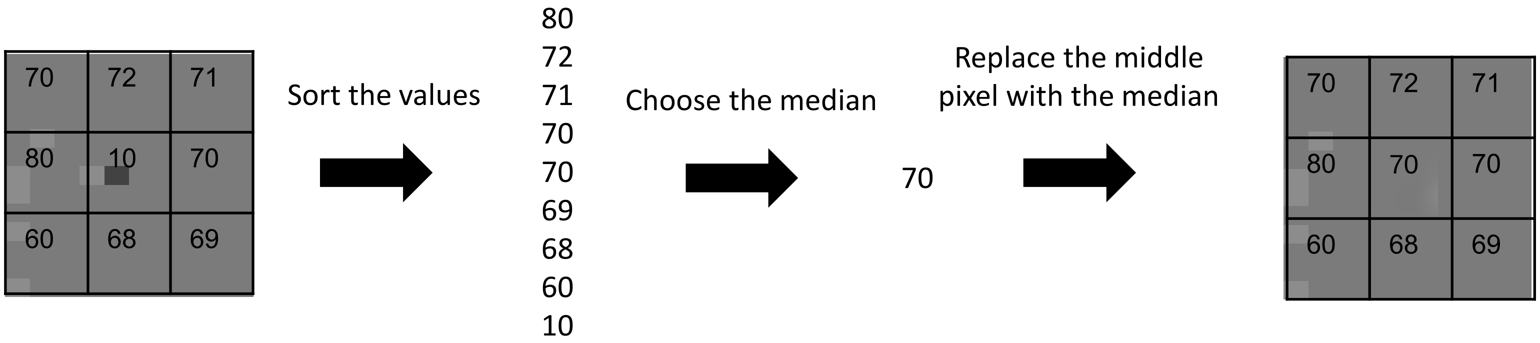
- Gradient Vector Magnitude:
  - $\sqrt{Grad_x^2 + Grad_y^2}$
  - Gradient magnitude shows location of edges in the image

- Gradient Vector Angle:
  - $Arctan2(Grad_y, Grad_x)$
  - Gradient angle shows direction of edge.

Nonlinear filters

- We talked about filters and convolution operation.
- The operations done for filtering the image were linear.
- Image filtering can be non-linear.

Median filter



70	72	71
80	10	70
60	68	69

Sort the values

80
72
71
70
70
70
69
68
60
10

Choose the median

70

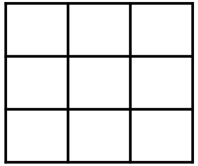
Replace the middle pixel with the median

70	72	71
80	70	70
60	68	69

This is a non-linear operation

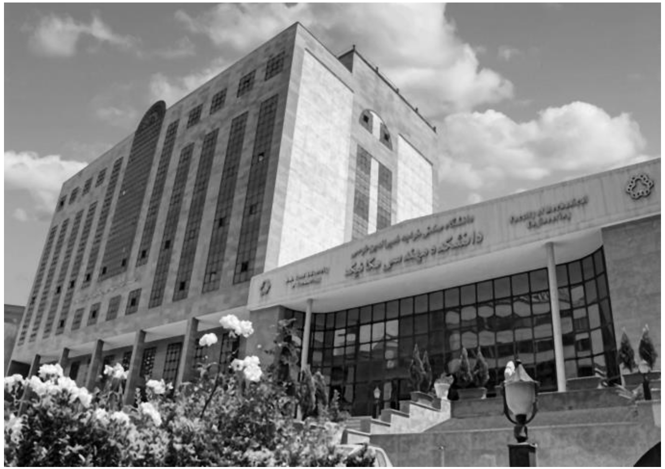
Median filter

- Just like linear filters, we slide the median filter on the entire image.
- This filter doesn't have coefficients, instead we calculate the median.



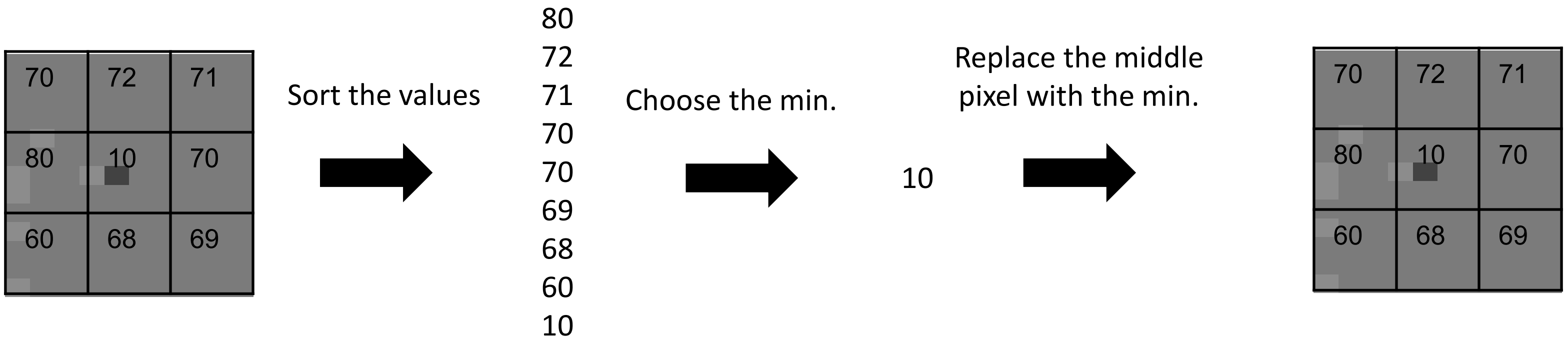
Filter

Raw image



Filtered image

Erosion



70	72	71
80	10	70
60	68	69

Sort the values

80
72
71
70
70
70
69
68
60
10

Choose the min.

10

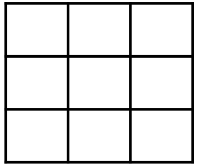
Replace the middle pixel with the min.

70	72	71
80	10	70
60	68	69

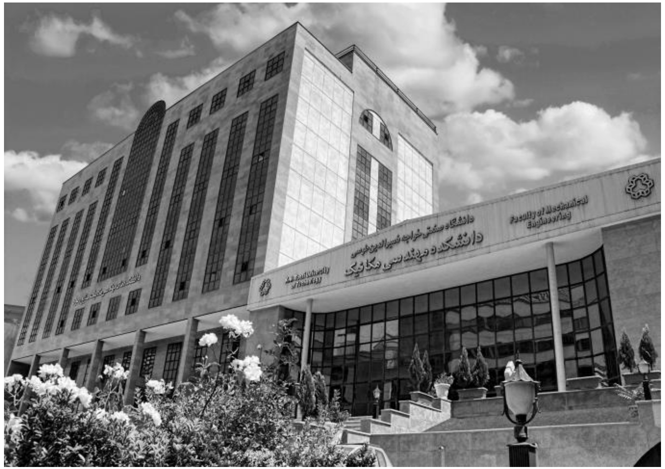
This is a non-linear operation

Erosion

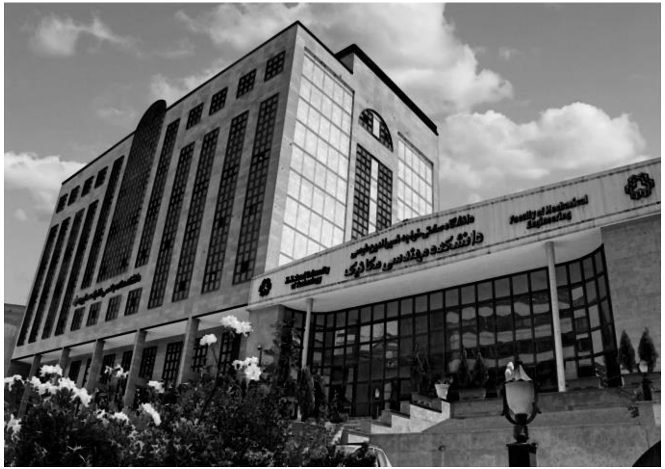
- Erosion uses the minimum value.
- Minimum values are closer to 0 (black).



Filter

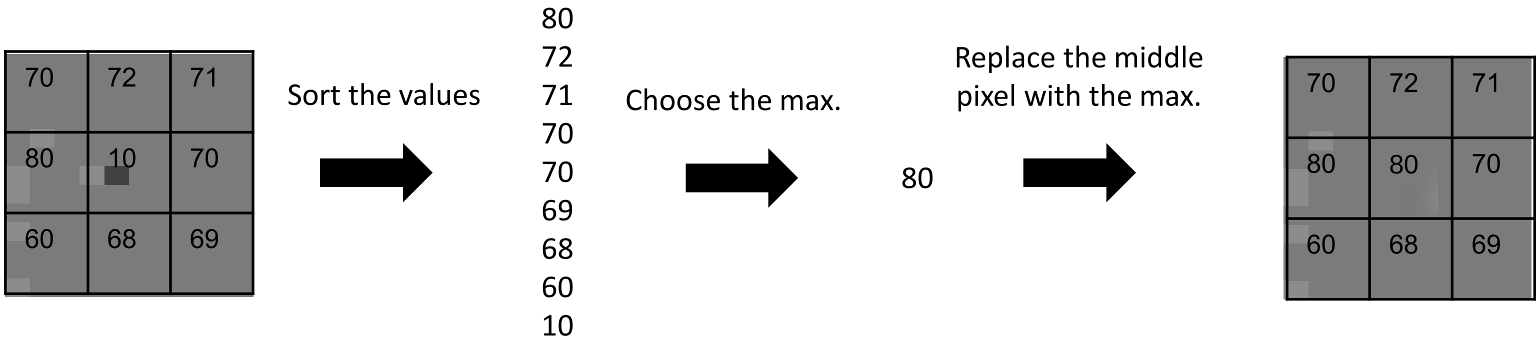


Raw image



Filtered image

Dilation



70	72	71
80	10	70
60	68	69

Sort the values

80
72
71
70
70
69
68
60
10

Choose the max.

80

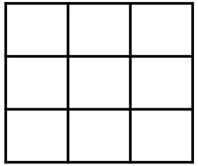
Replace the middle pixel with the max.

70	72	71
80	80	70
60	68	69

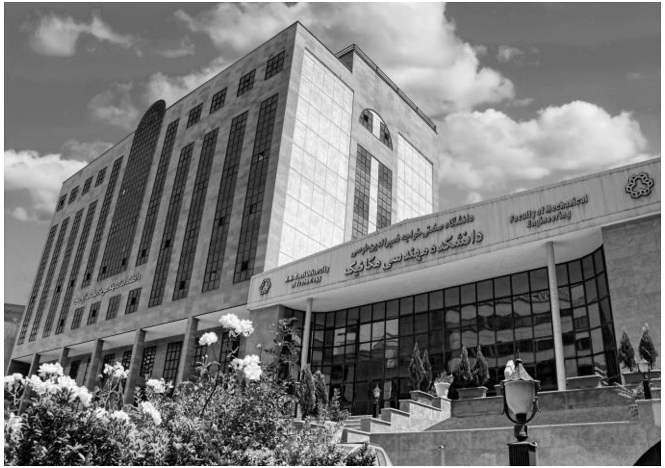
This is a non-linear operation

Dilation

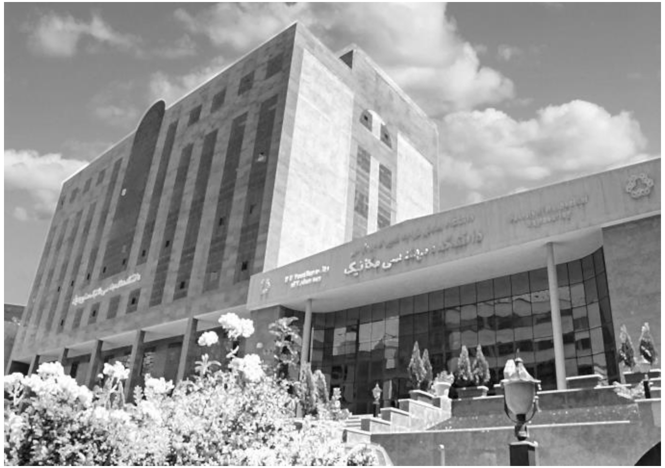
- Dilation uses the maximum value.
- Maximum values are closer to 255 (white).



Filter



Raw image



Filtered image