Math 3360: Mathematical Imaging

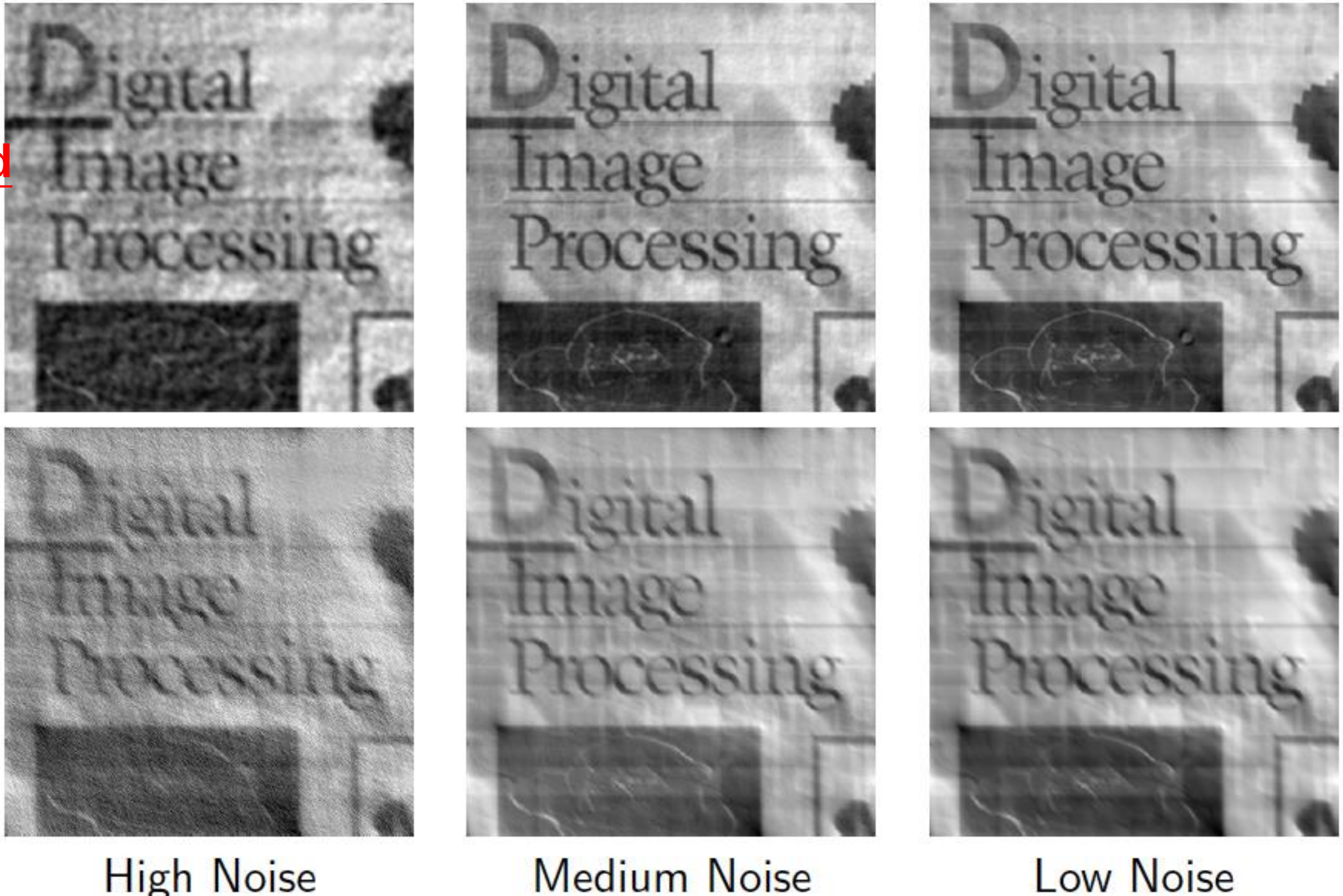
Lecture 15:
Constrained least square filtering

Prof. Ronald Lok Ming Lui

Department of Mathematics,
The Chinese University of Hong Kong

Constrained least square filtering

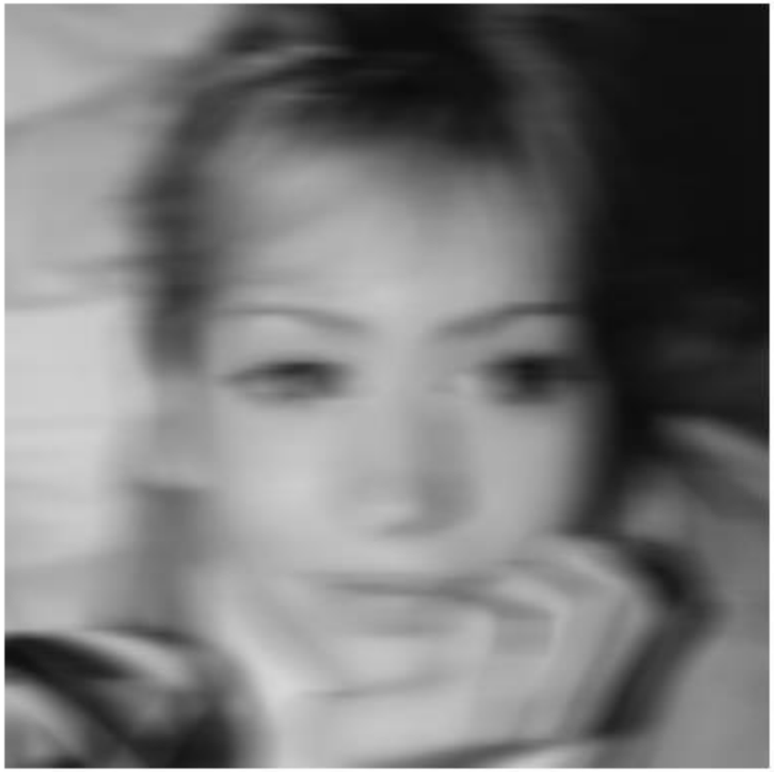
$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma |P(u,v)|^2} \right] G(u,v)$$



Constrained Least Square

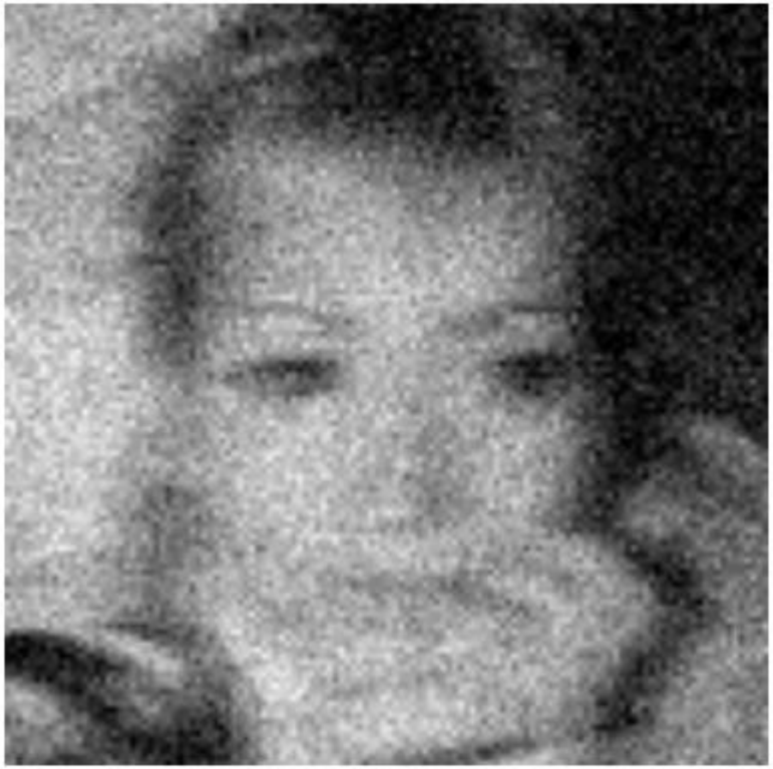
Wiener filter

Constrained least square filtering



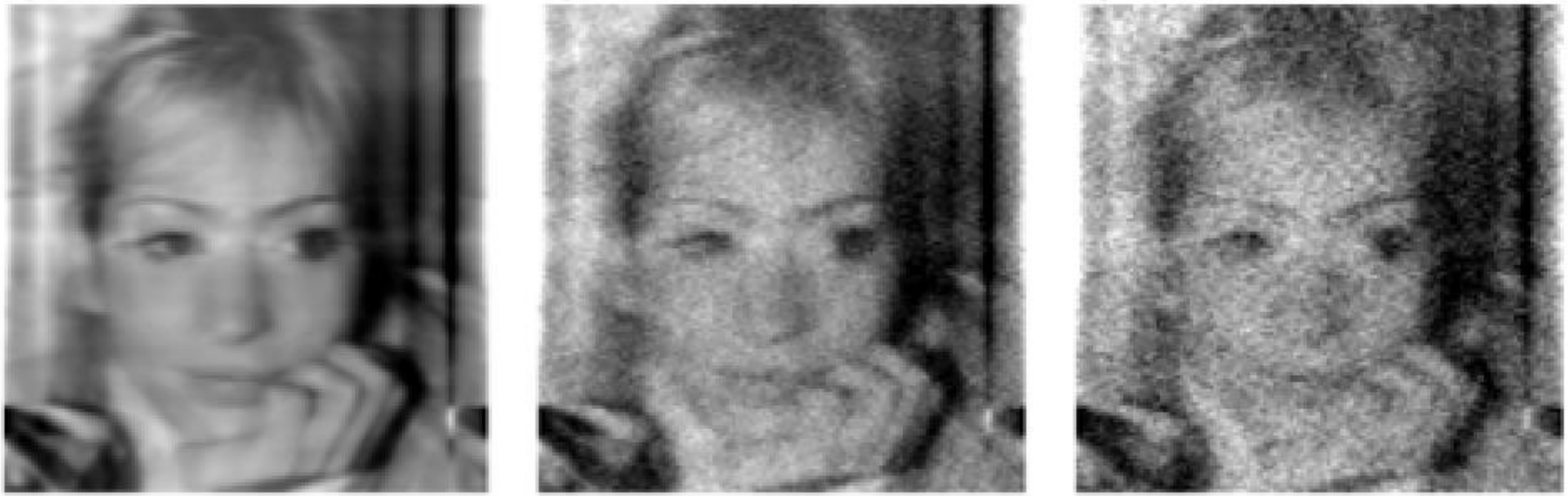
Blurry image without noise

Constrained least square filtering

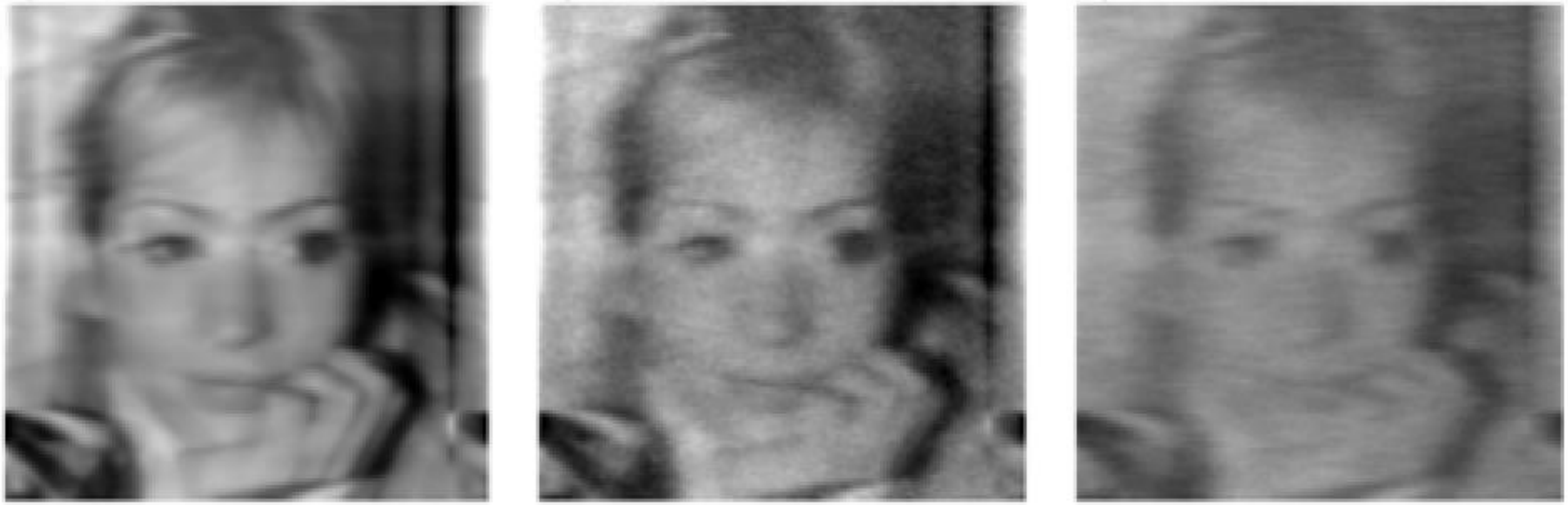


Blurry image with noise

Constrained least square filtering

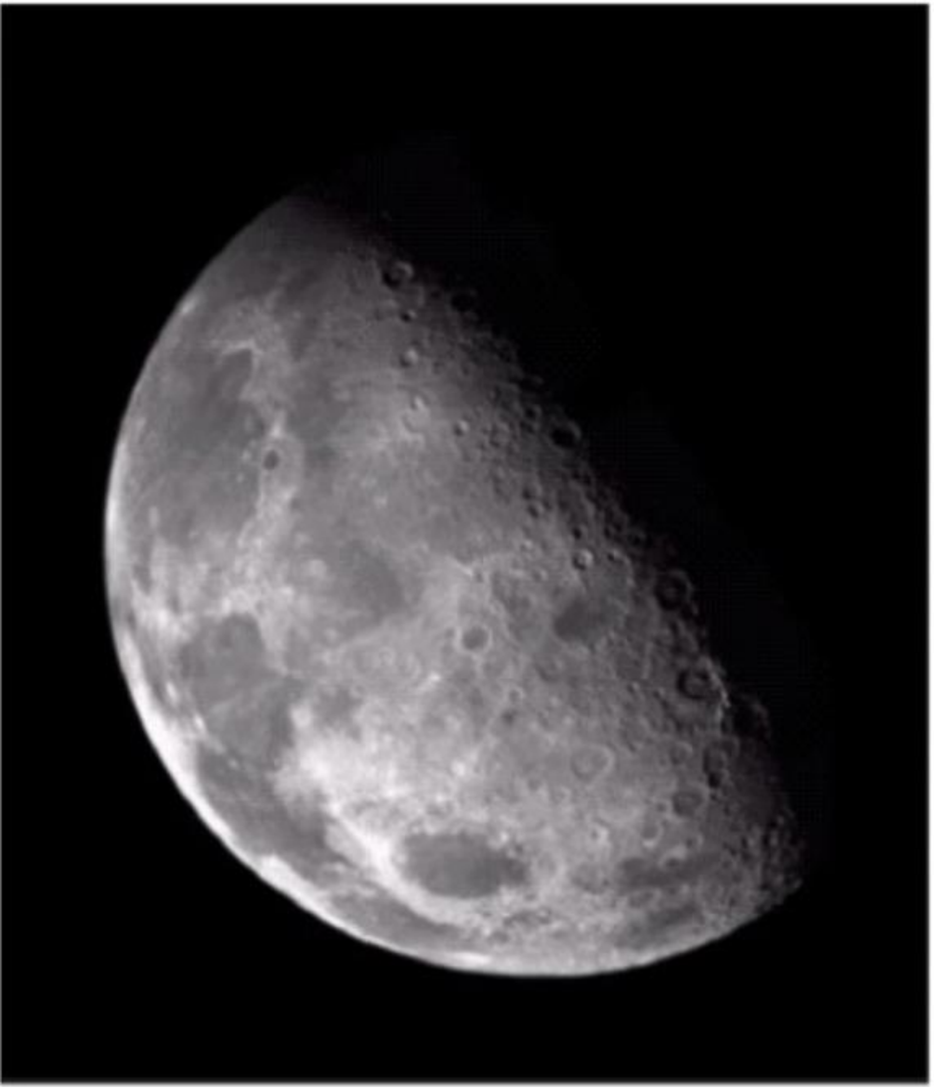


Blurry images

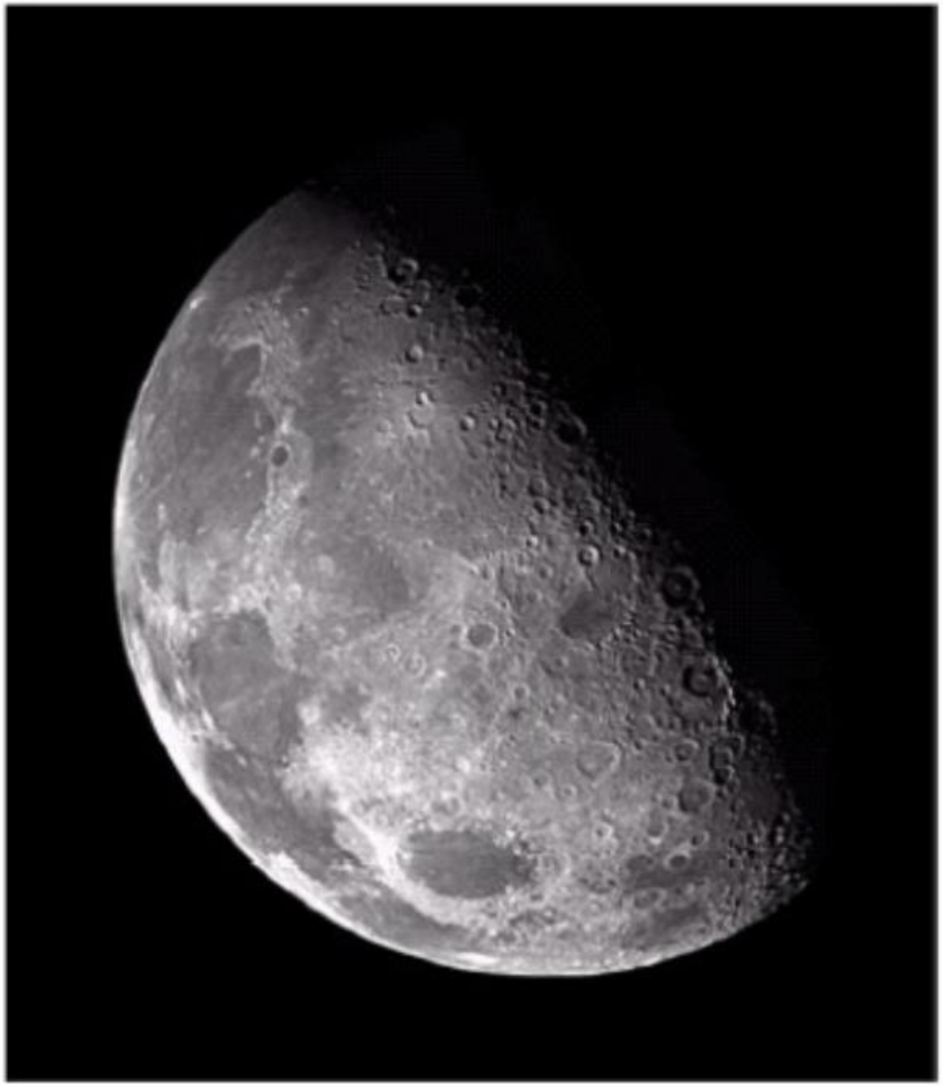


Deblurred images

Laplacian mask

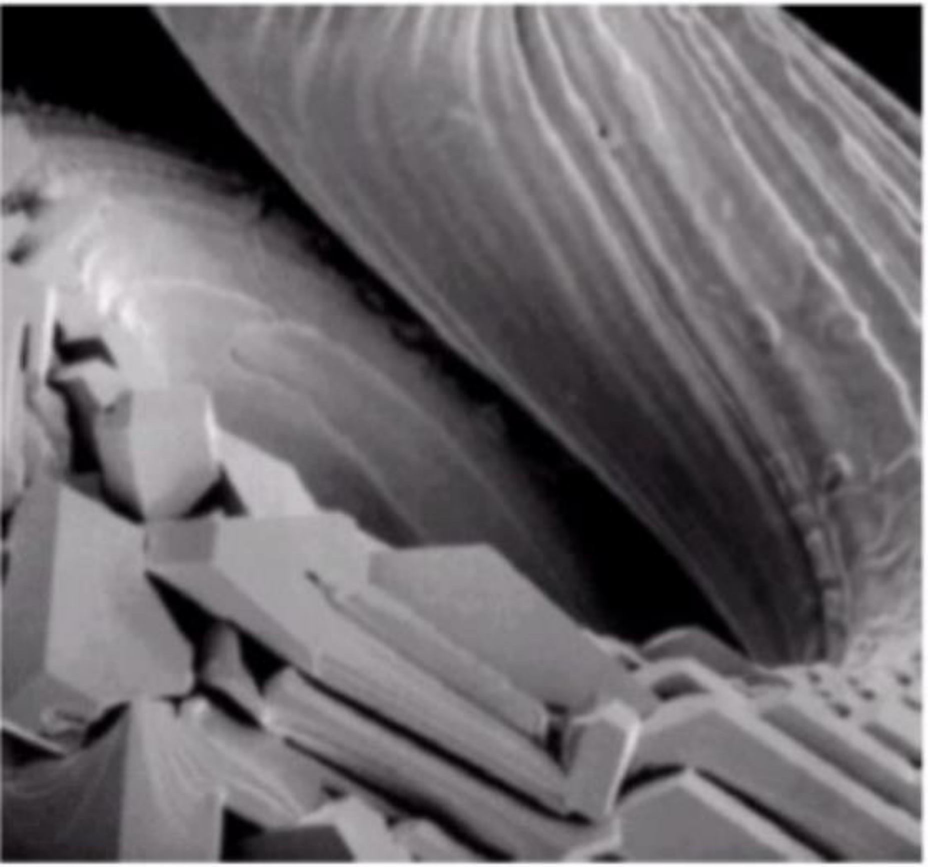


Original image

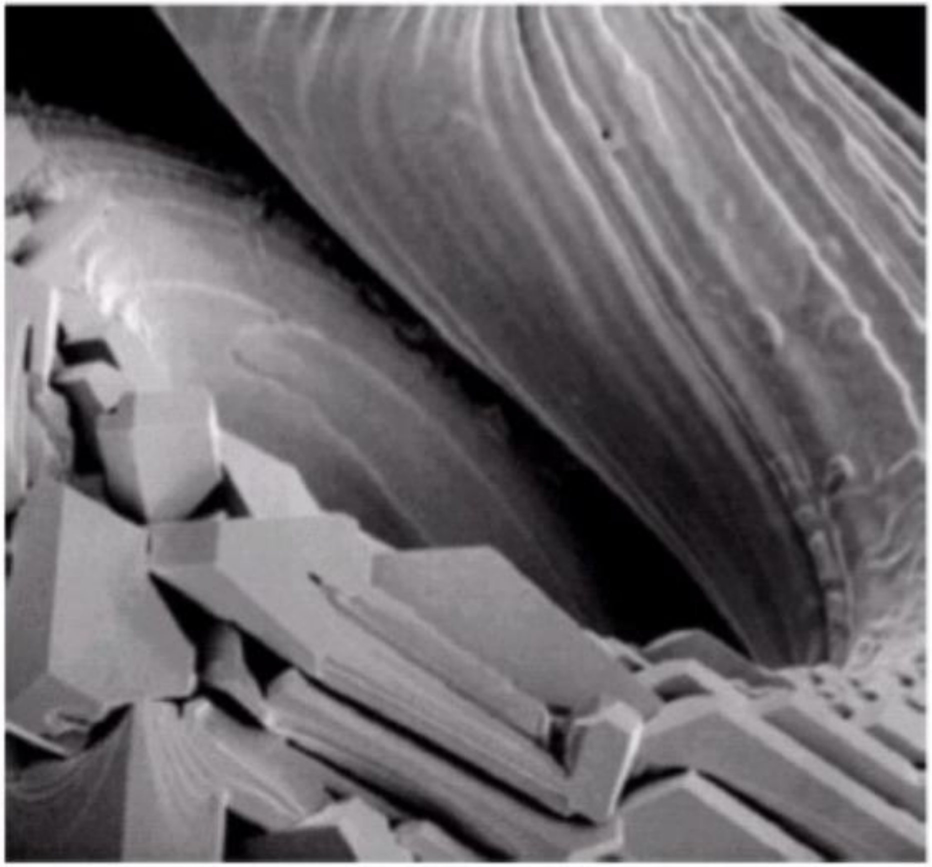


Laplacian mask

Laplacian mask

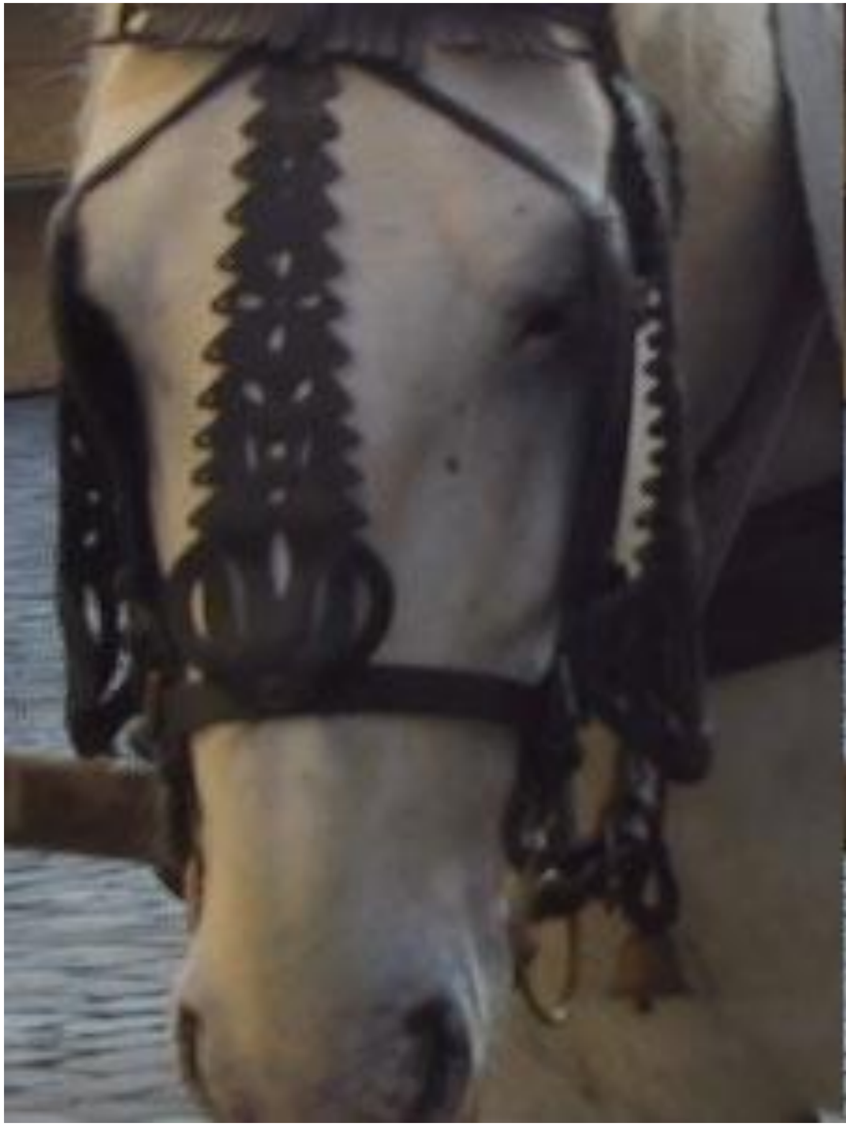


Original image

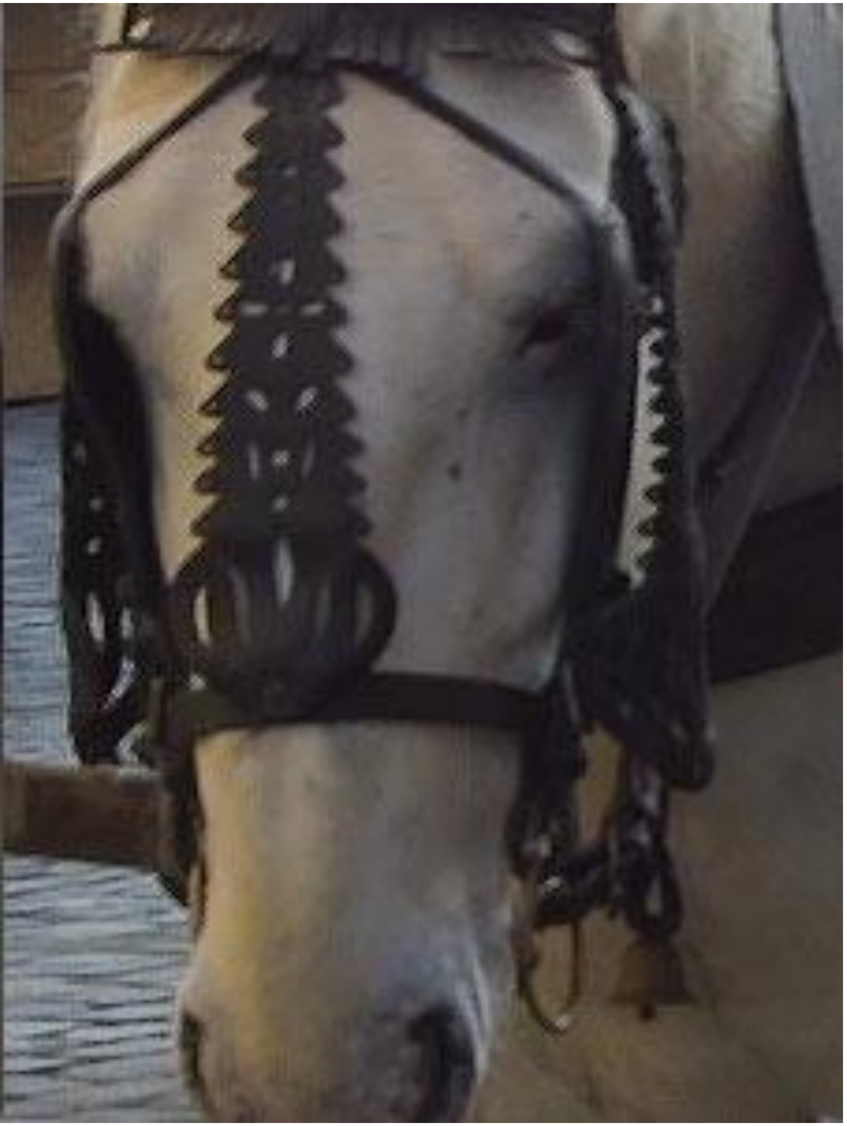


Laplacian mask

Laplacian mask

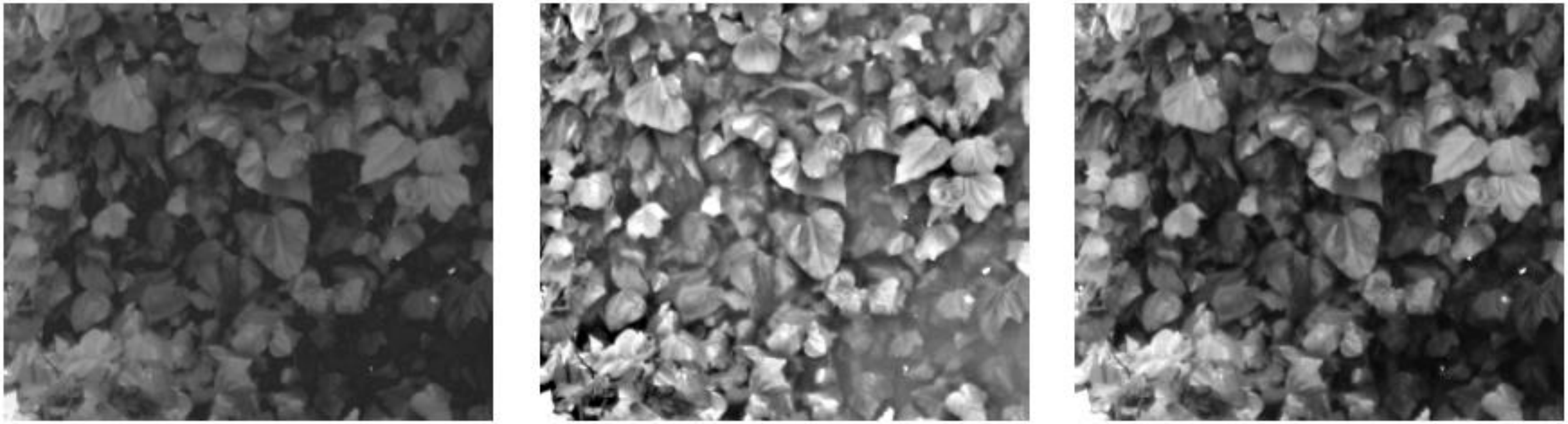


Original image



Laplacian mask

Unsharp masking

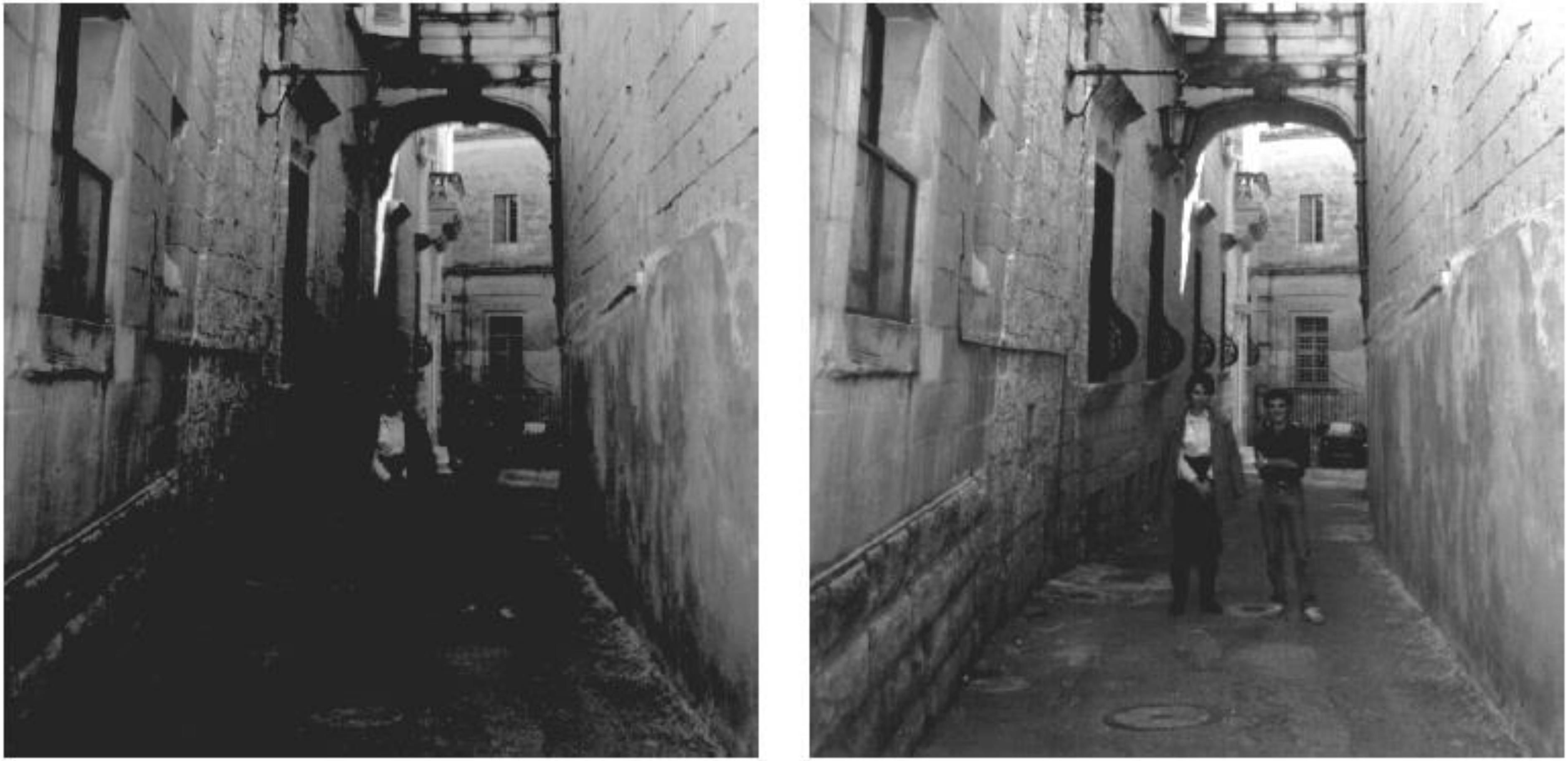


(a) Original

(b) Gaussian

(c) Mean

Unsharp masking



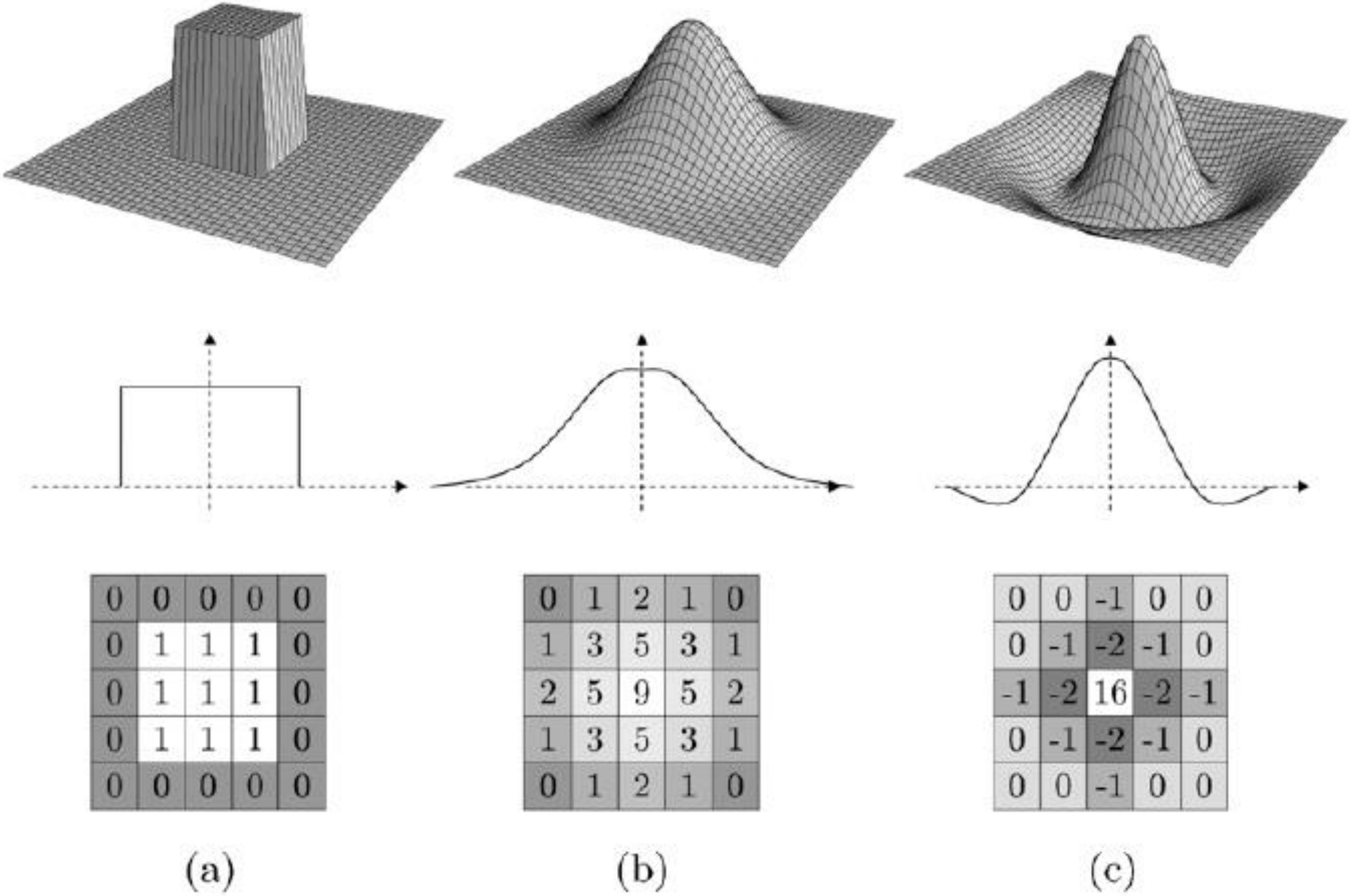
(a) Original

(b) Global, Gaussian 121 × 121

Linear filter = Convolution

- Linear filtering of a (2M+1)x(2N+1) image I (defined on [-M,M]x[-N,N]) = CONVOLUTION OF I and H
- H is called the filter.
- Different filter can be used:
  - Mean filter
  - Gaussian filter
- Variation of these filters (Non-linear)
  - Median filter
  - Edge preserving mean filter

Type of filter



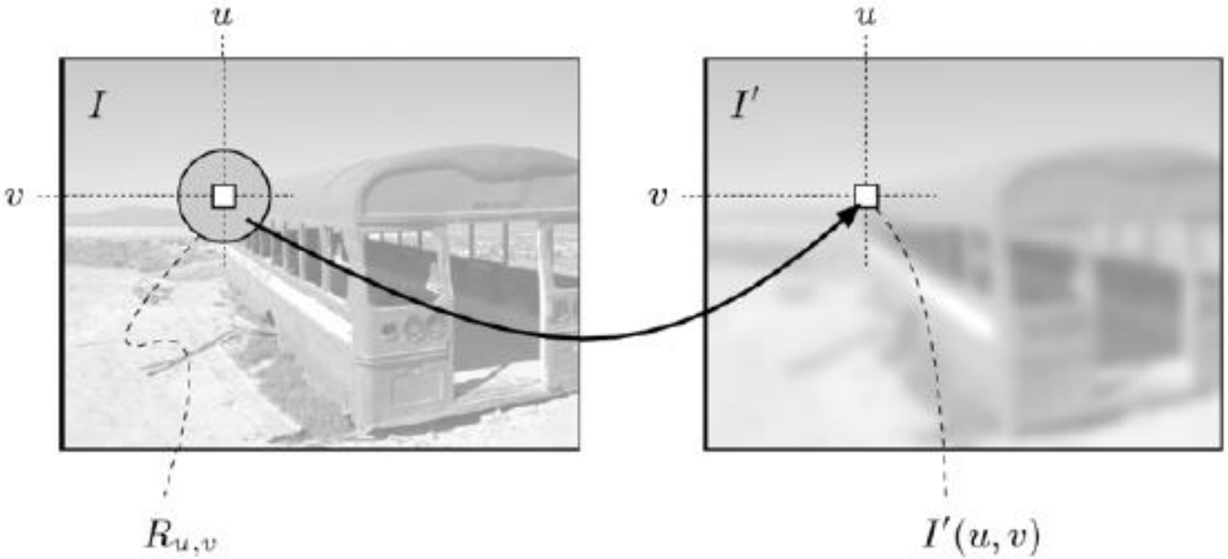
0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

0	1	2	1	0
1	3	5	3	1
2	5	9	5	2
1	3	5	3	1
0	1	2	1	0

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

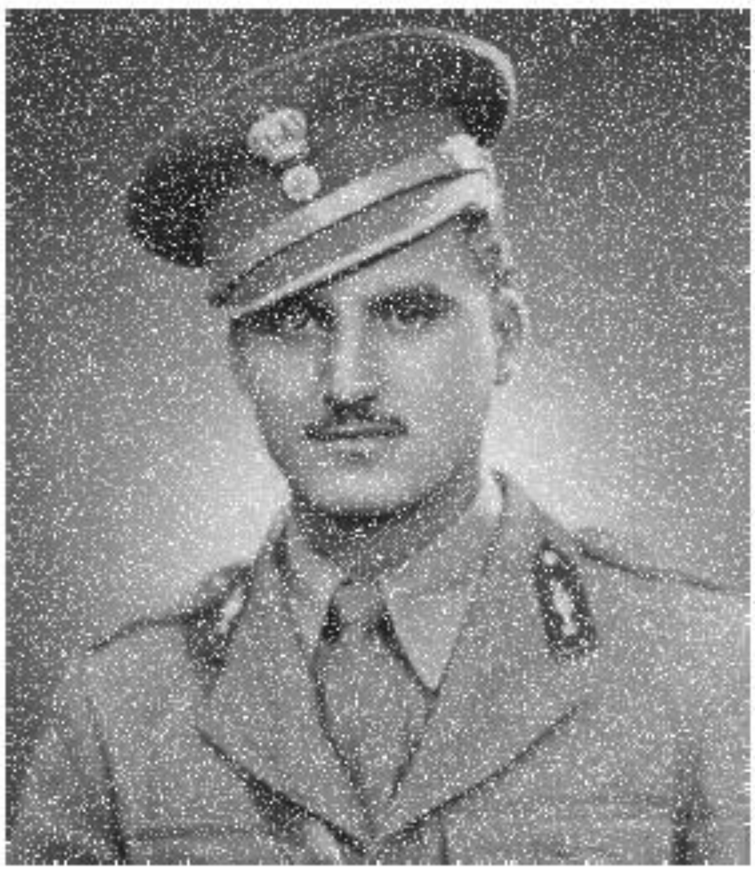
Mean filter

$$H(i,j) = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix} = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

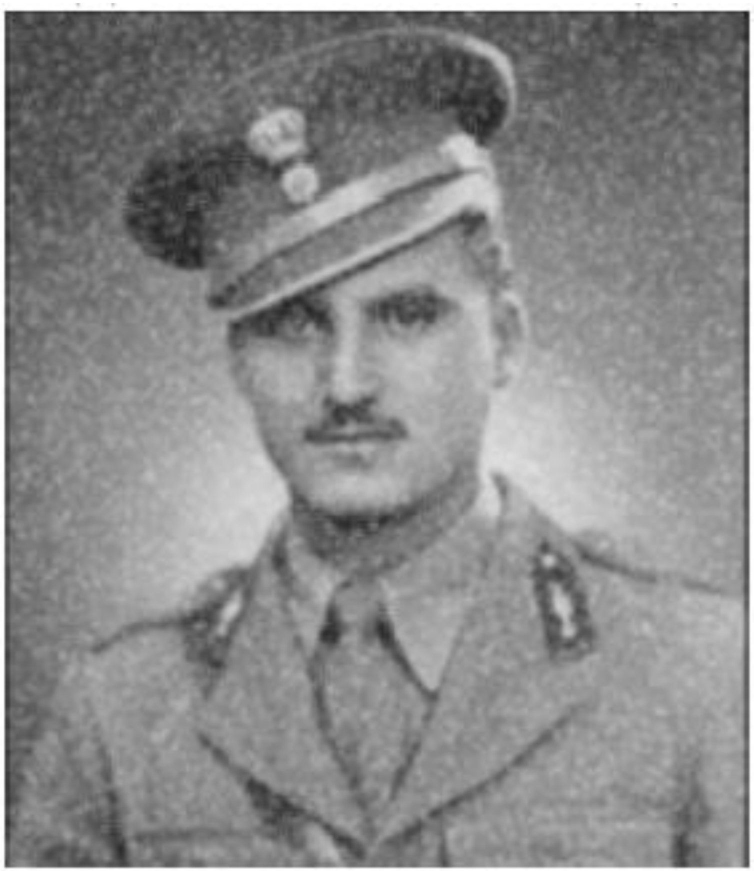


$$I'(u,v) \leftarrow \frac{p_0 + p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8}{9}$$

Mean filter

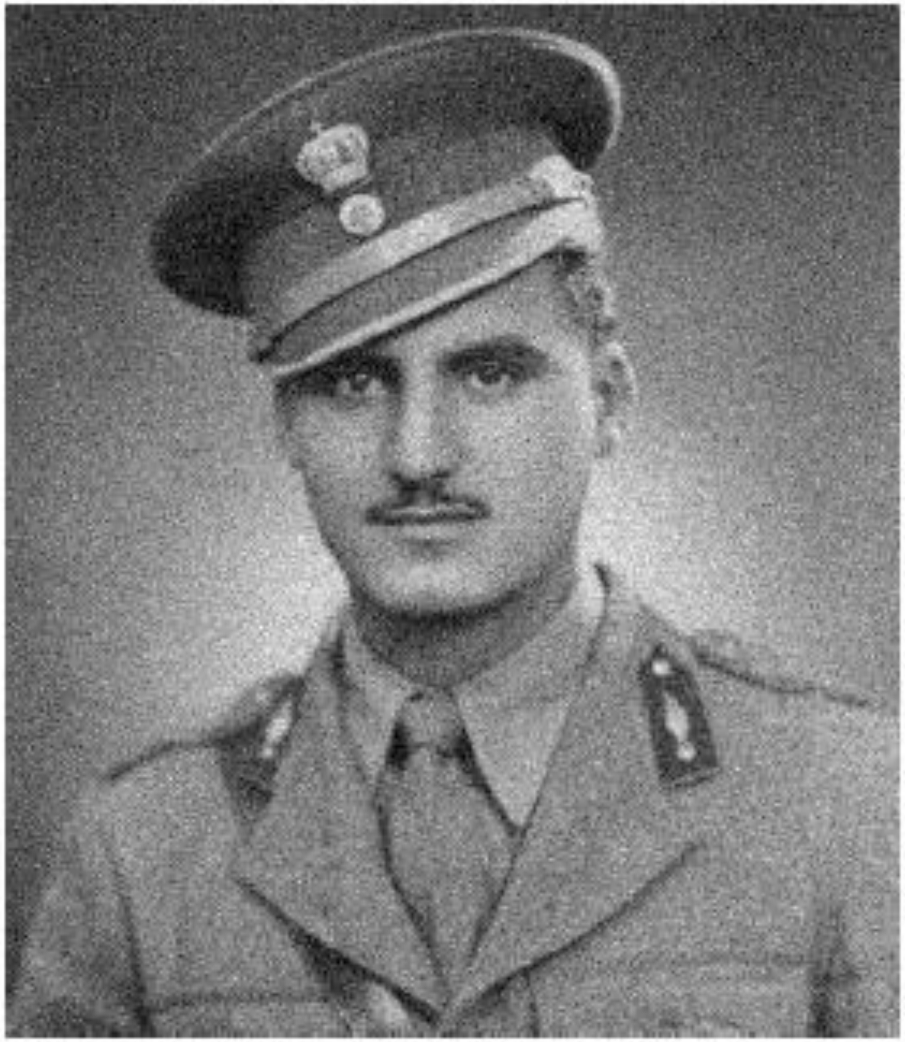


Impulse noise

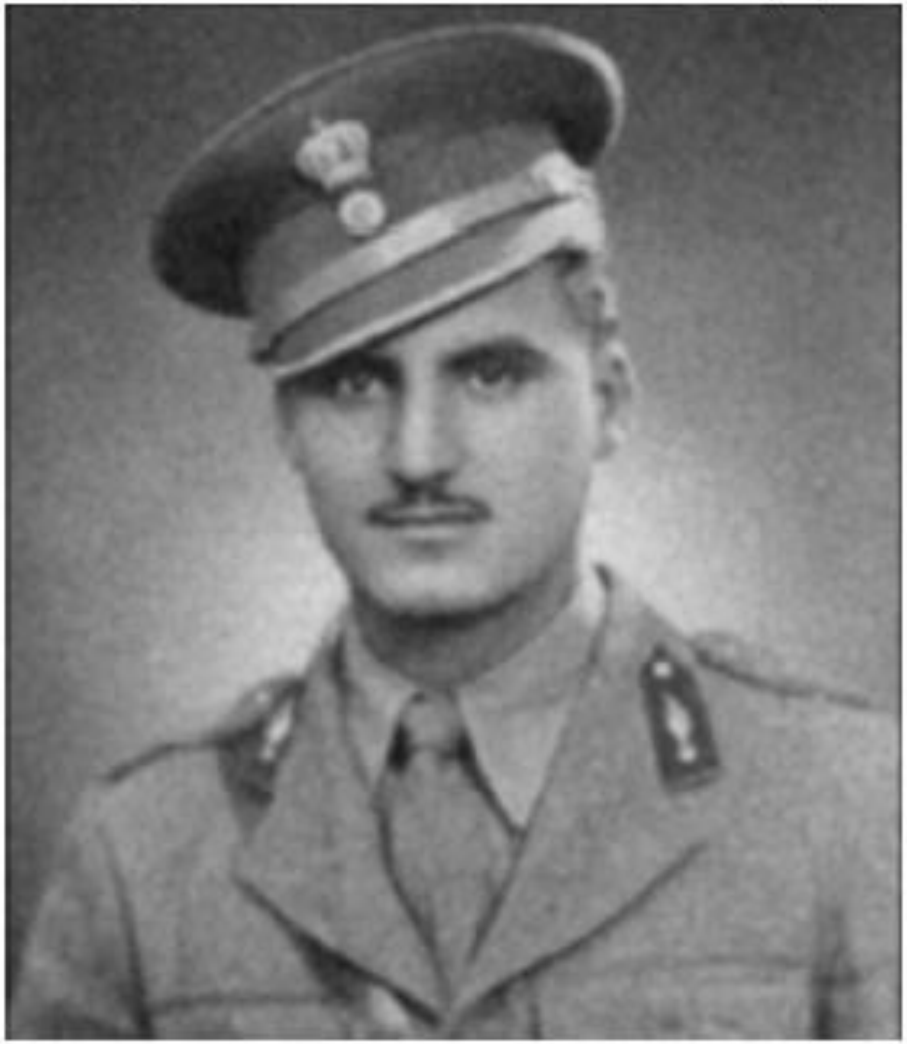


After mean filter

Mean filter

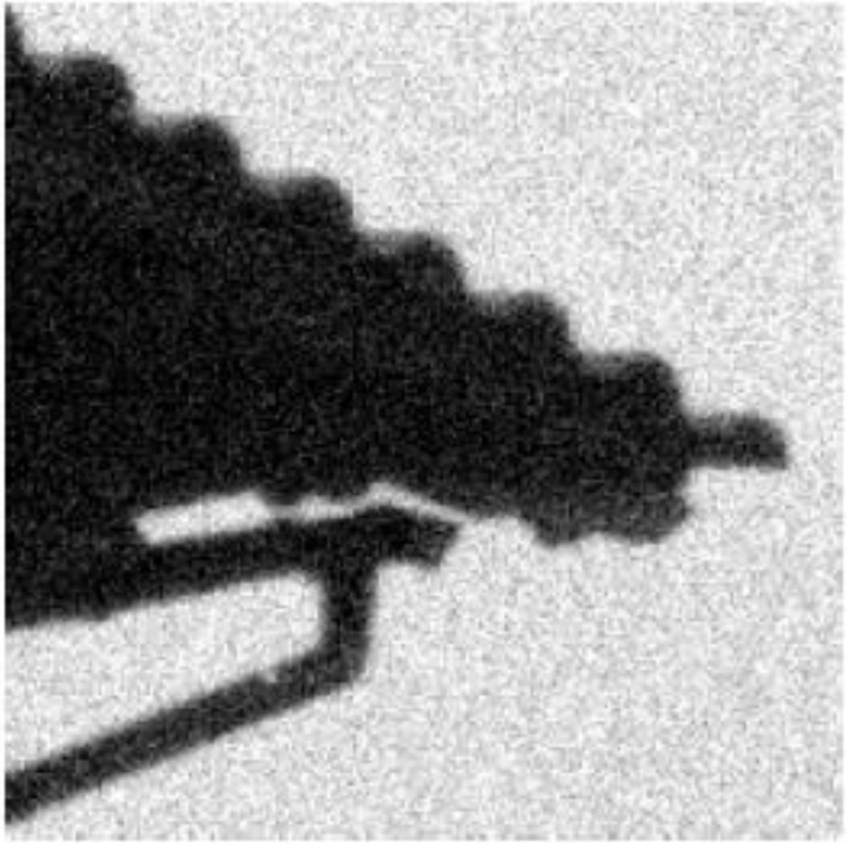


Gaussian noise

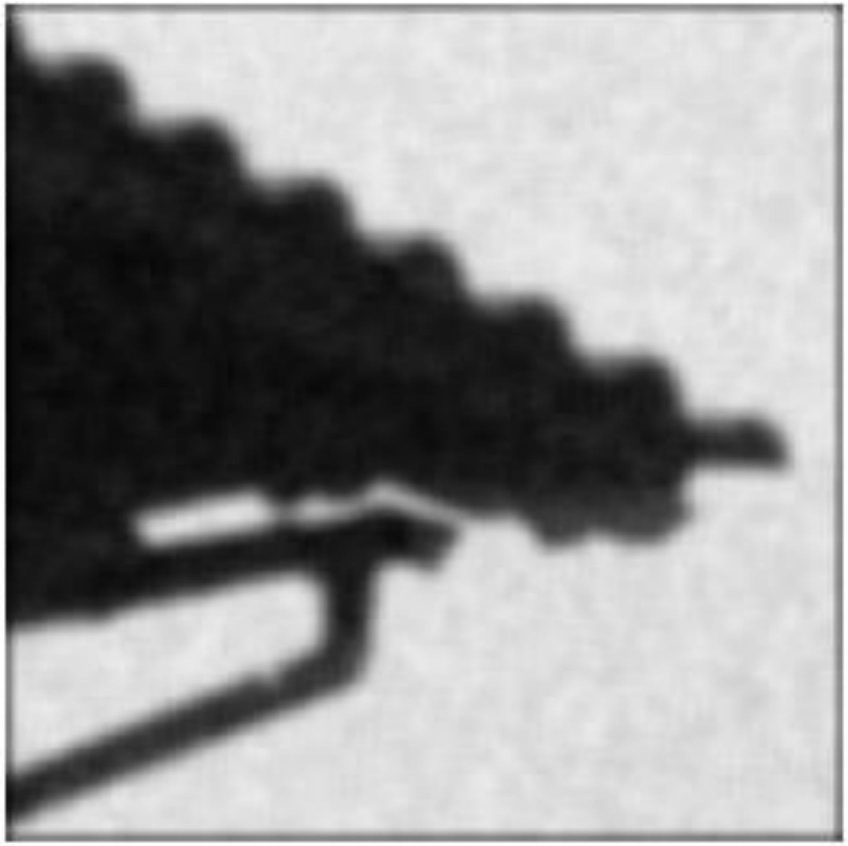


After mean filter

Mean filter

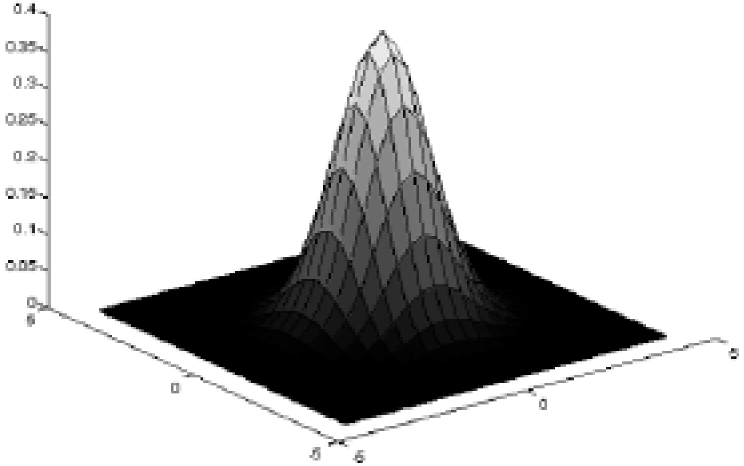


Real image

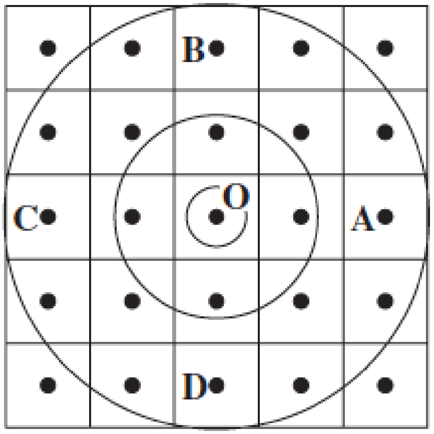


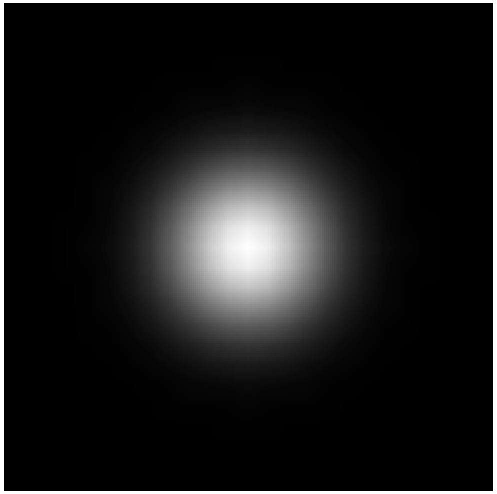
After mean filter

Gaussian filter



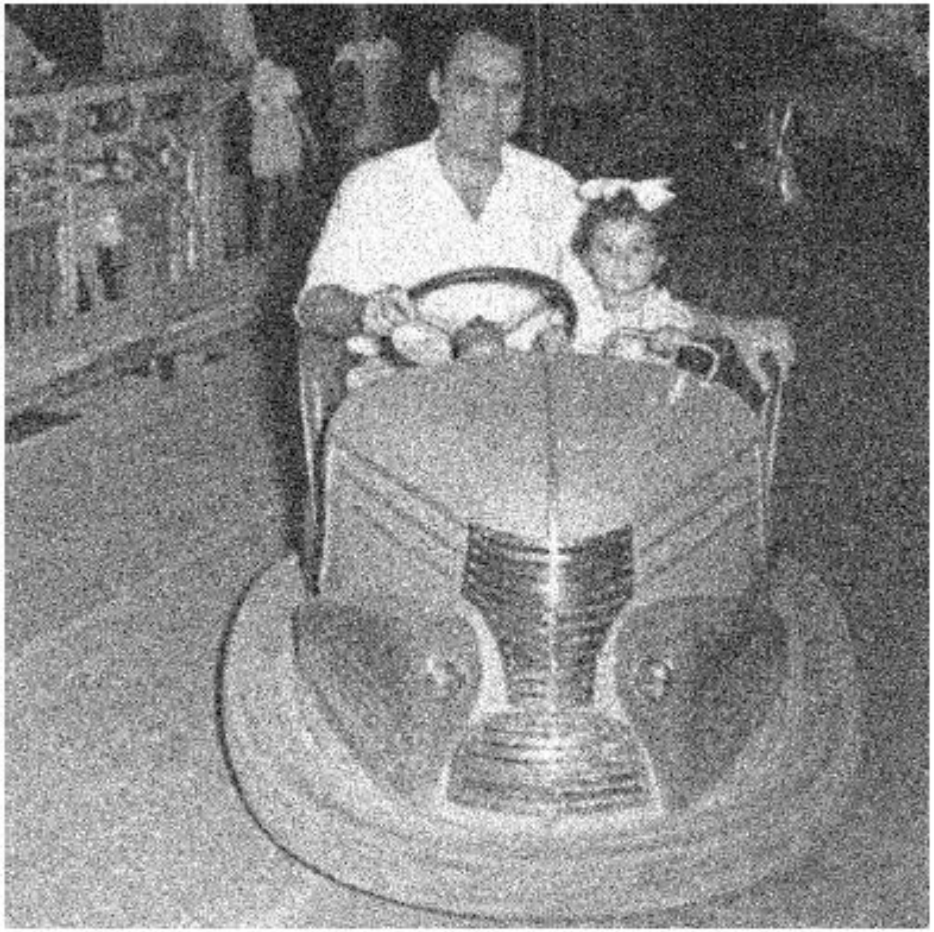
Define a function using Gaussian function



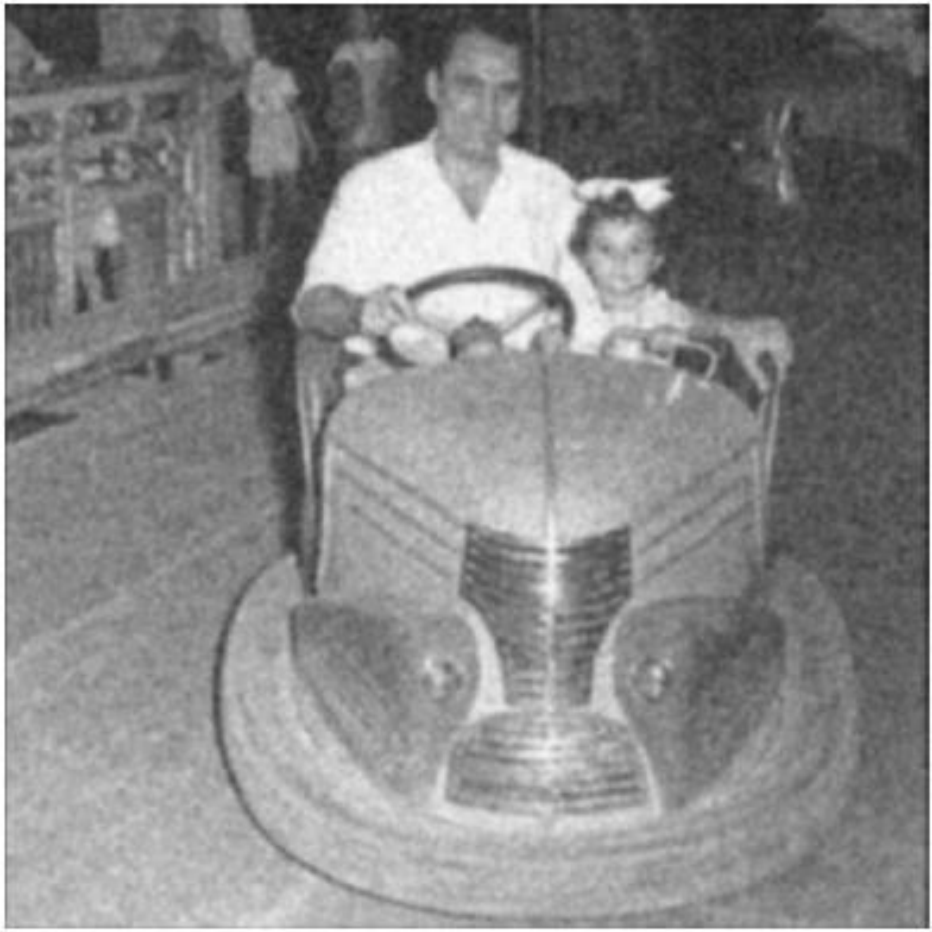


Definition of H

Gaussian filter



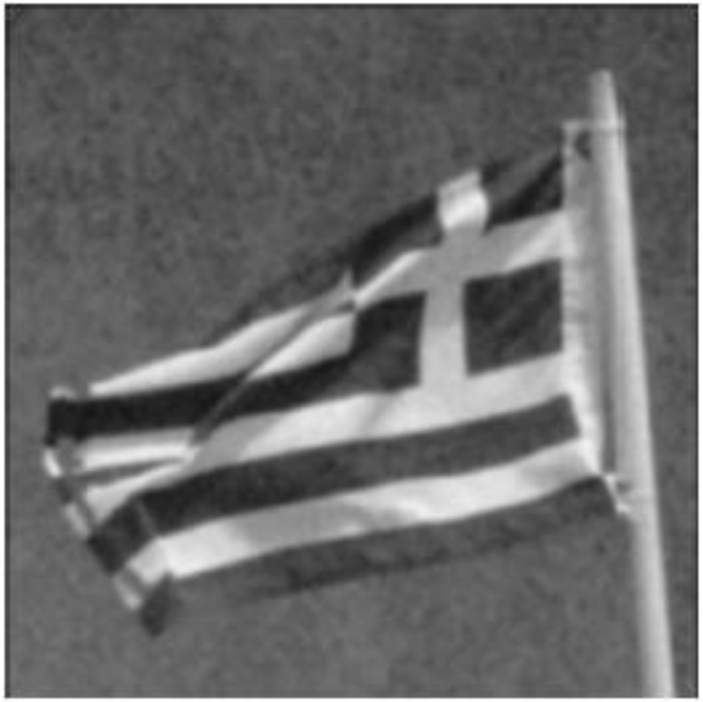
Real image



After Gaussian filter

Gaussian filter

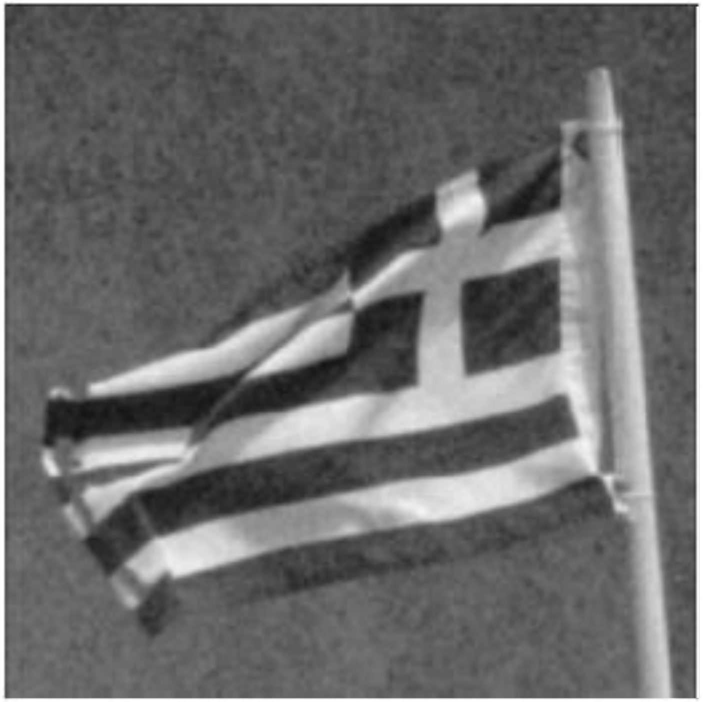
Real image



After mean filter

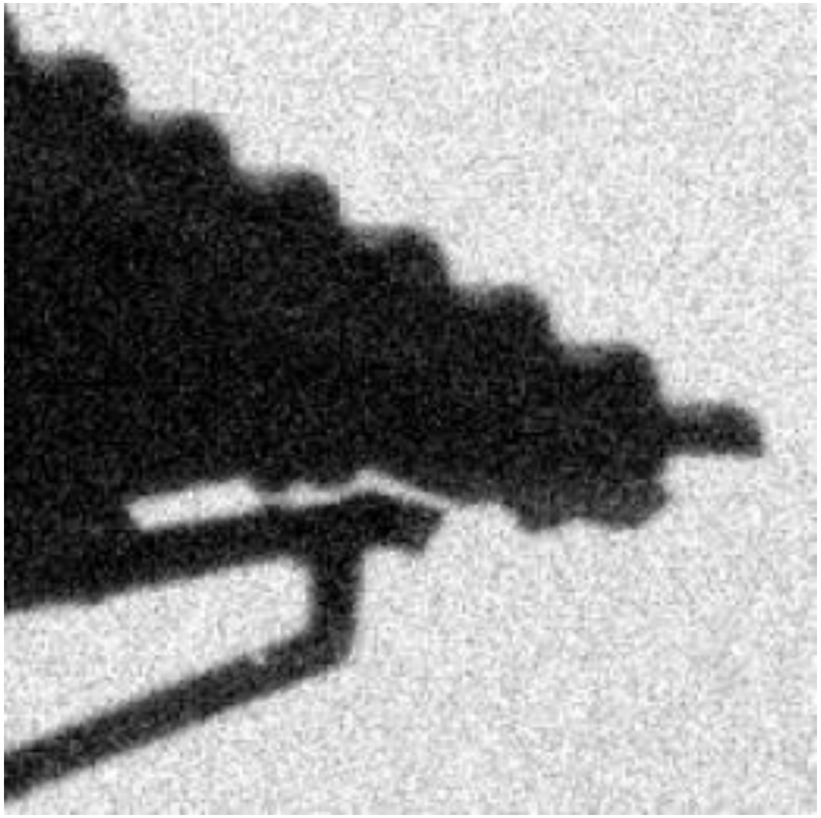
Gaussian filter

Real image

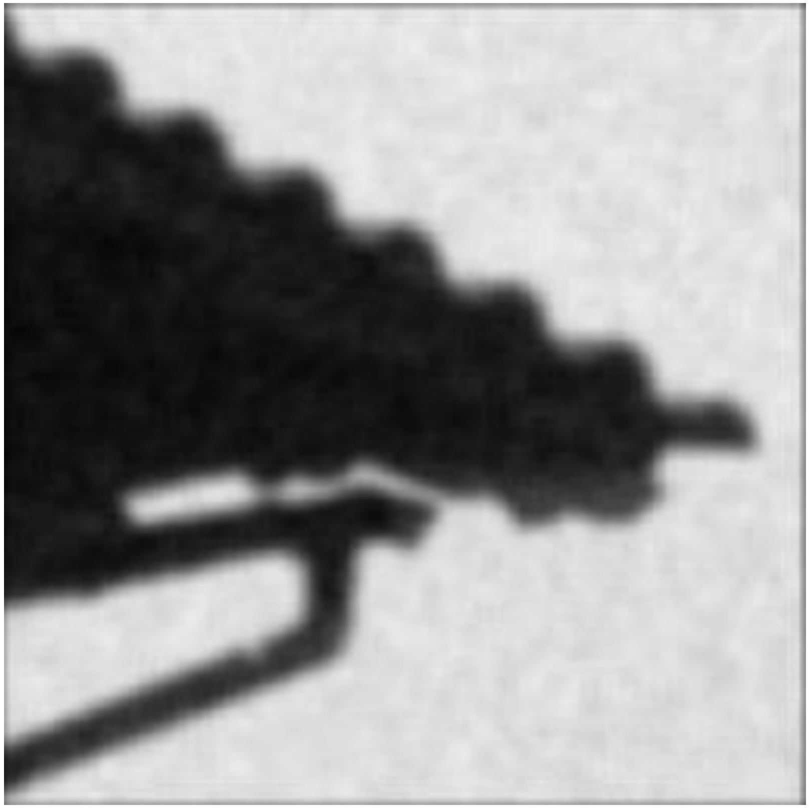


After Gaussian filter

Gaussian filter

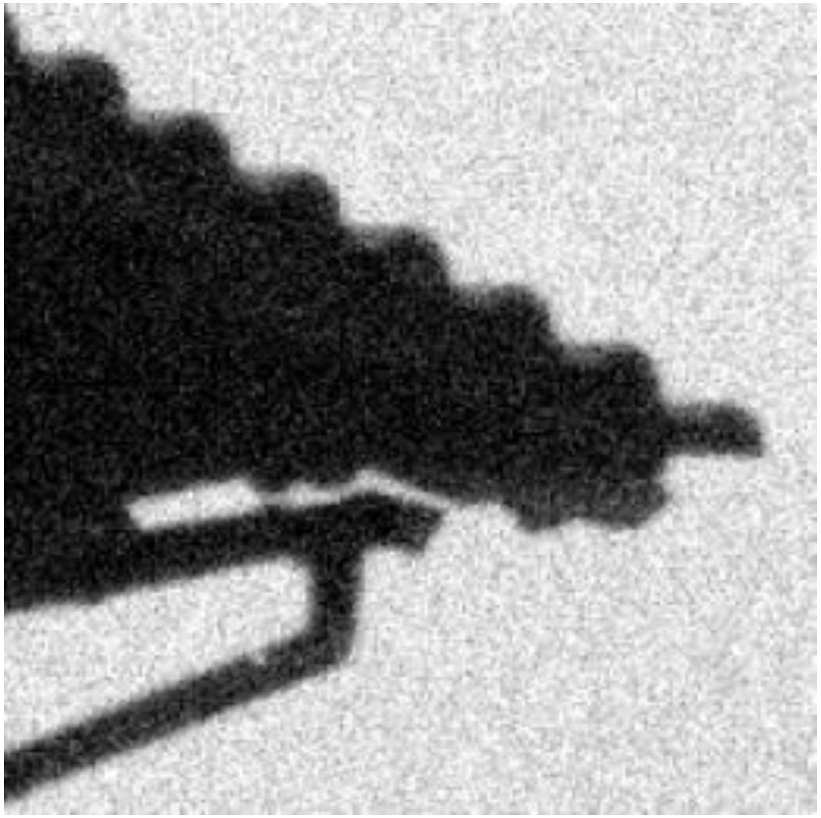


Real image

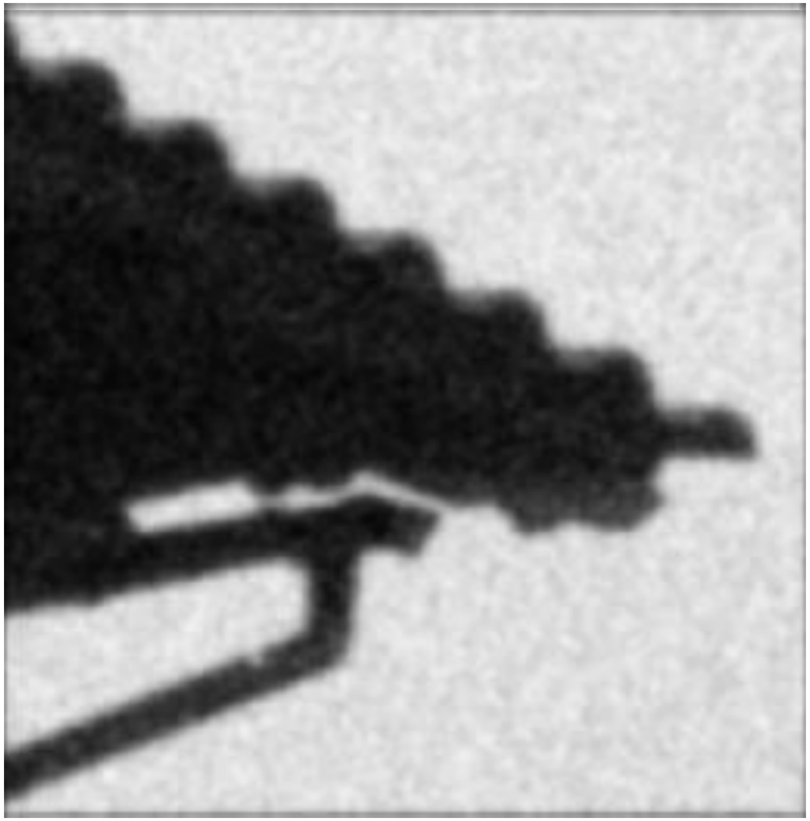


After mean filter

Gaussian filter



Real image

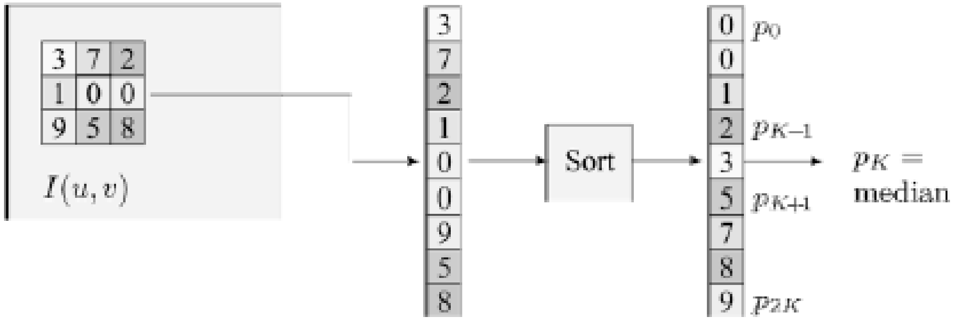


After Gaussian filter

Median filter

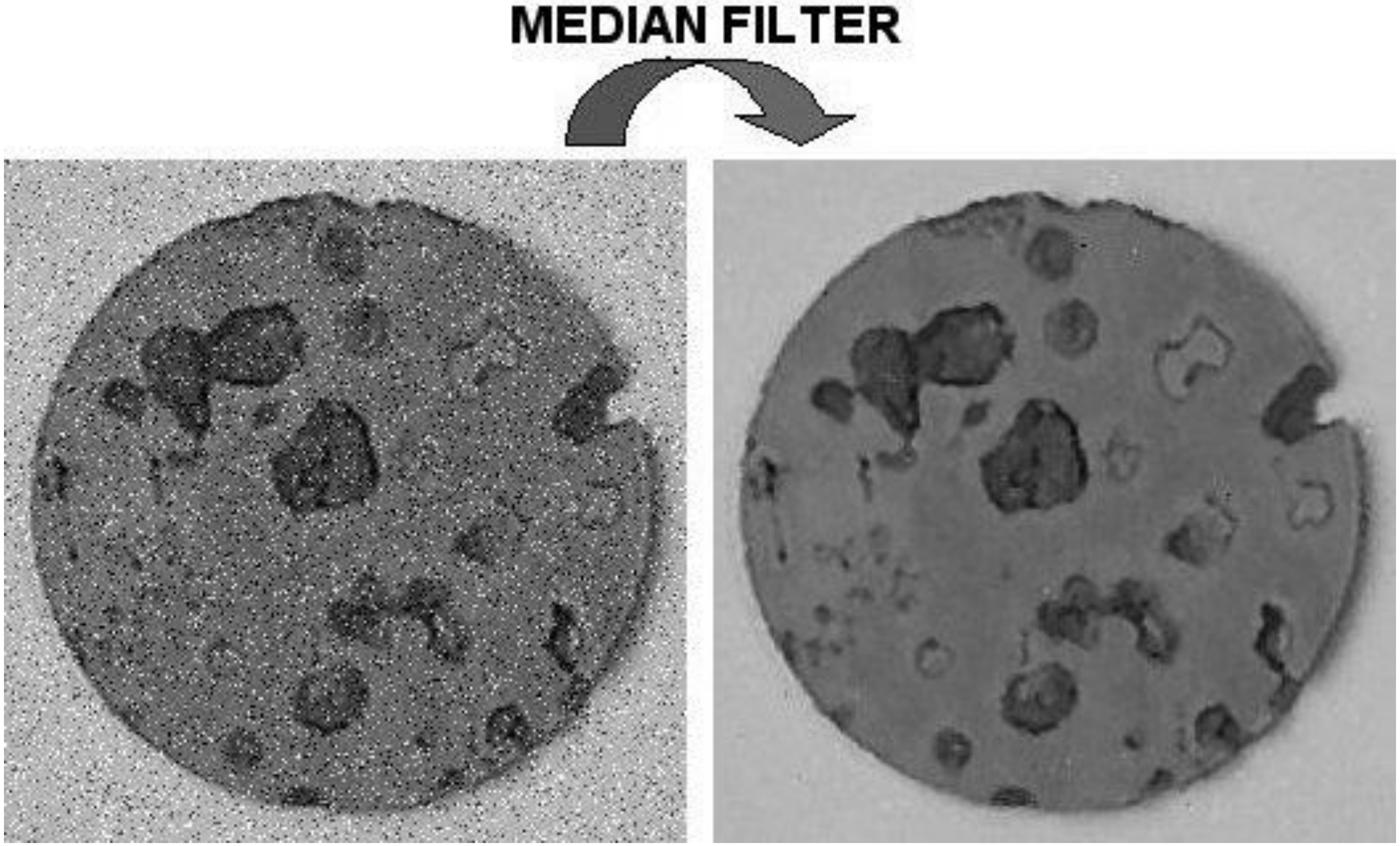
- Median
  - Nonlinear filter
  - Take median within a local window

$$I'(u,v) \leftarrow \text{median}\,\{I(u+i, v+j) \mid (i,j) \in R\}$$



3	7	2
1	0	0
9	5	8

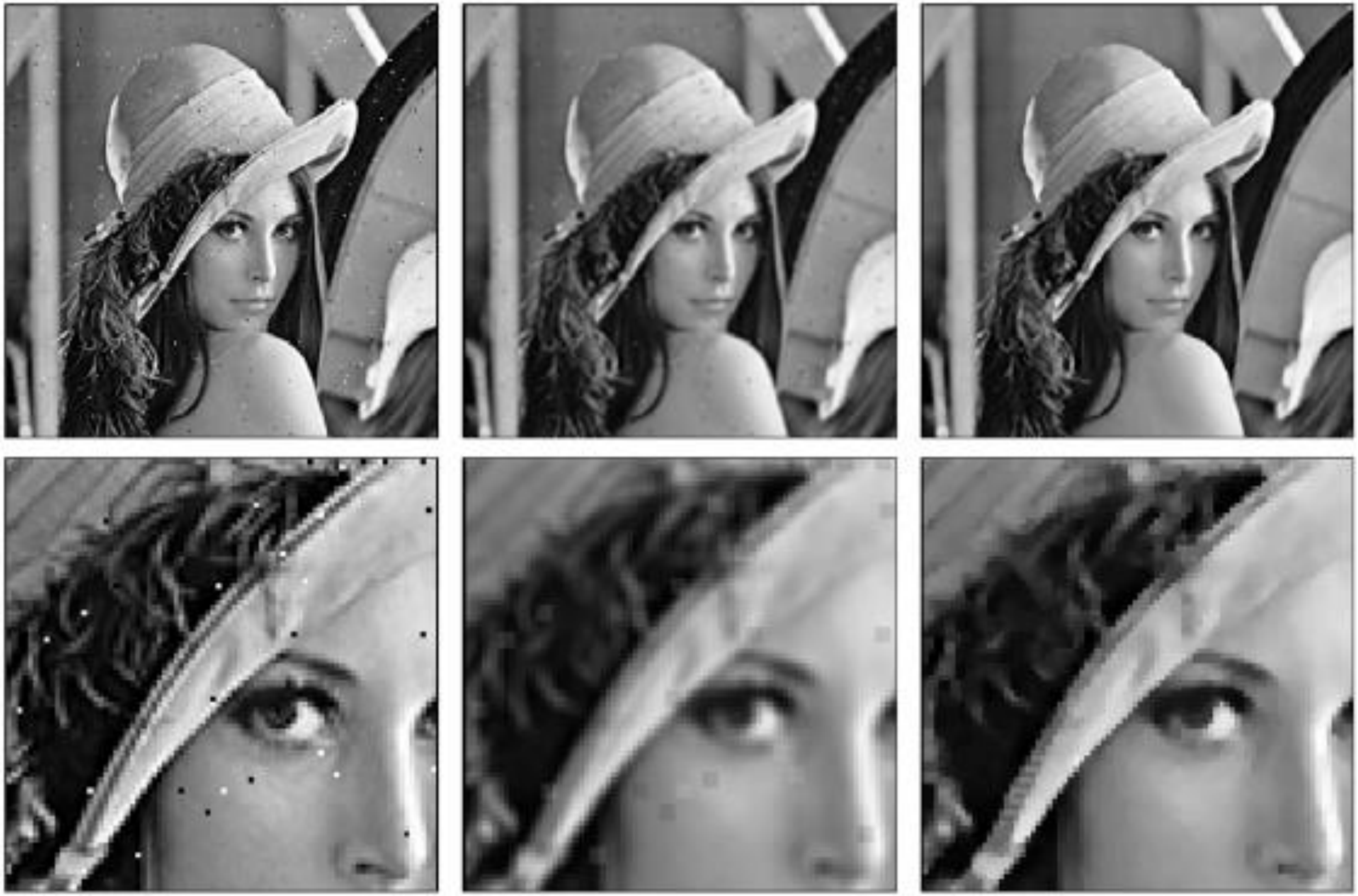
Median filter



Real image

After median filter

Median filter

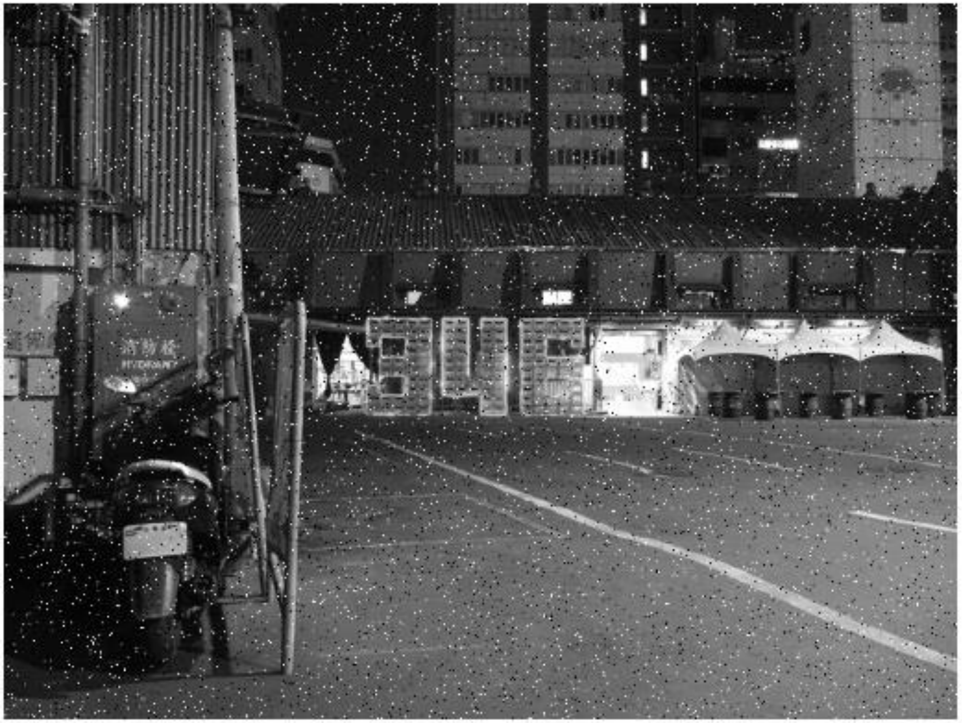


Salt & Pepper

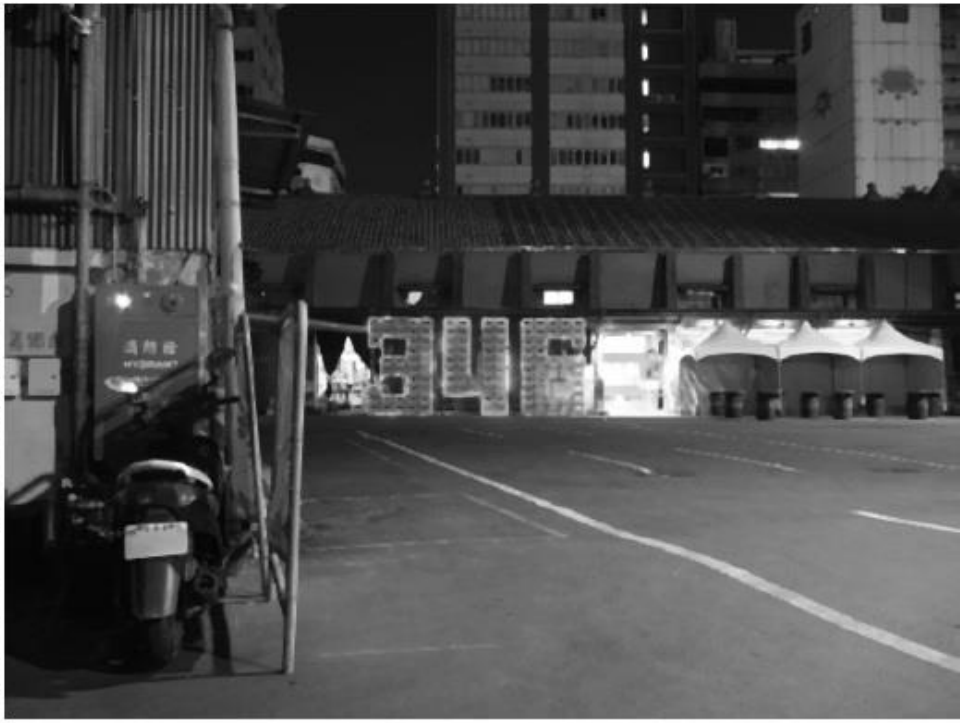
Mean filter

Median filter

Median filter



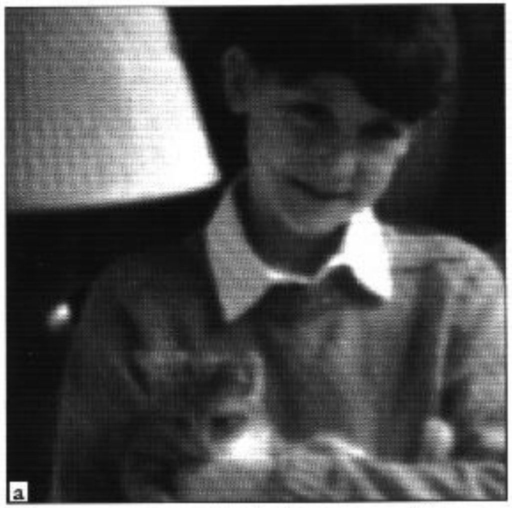
Noisy image



Median filter

Median filter

Original image



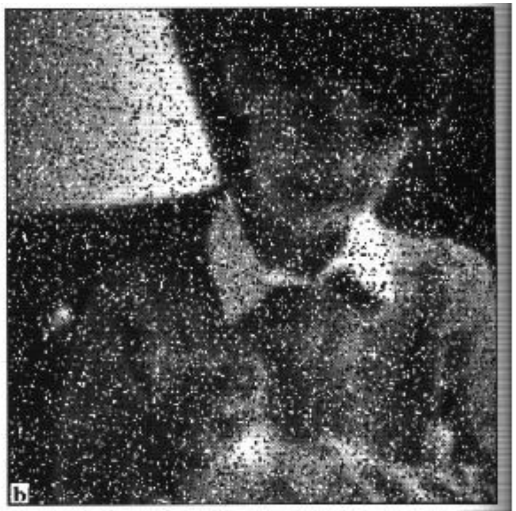
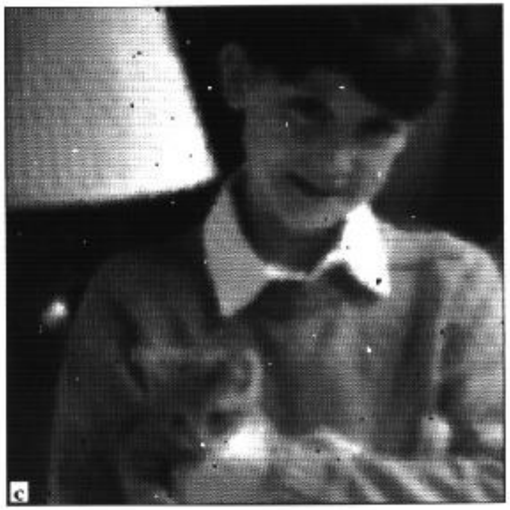
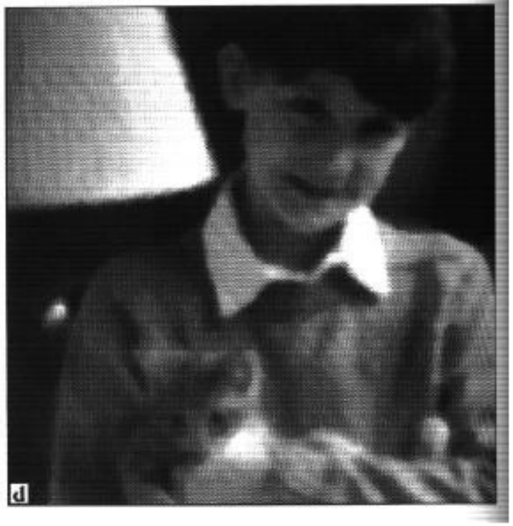


Image a with 10% of the pixels randomly selected and set to black, and another 10% randomly selected and set to white

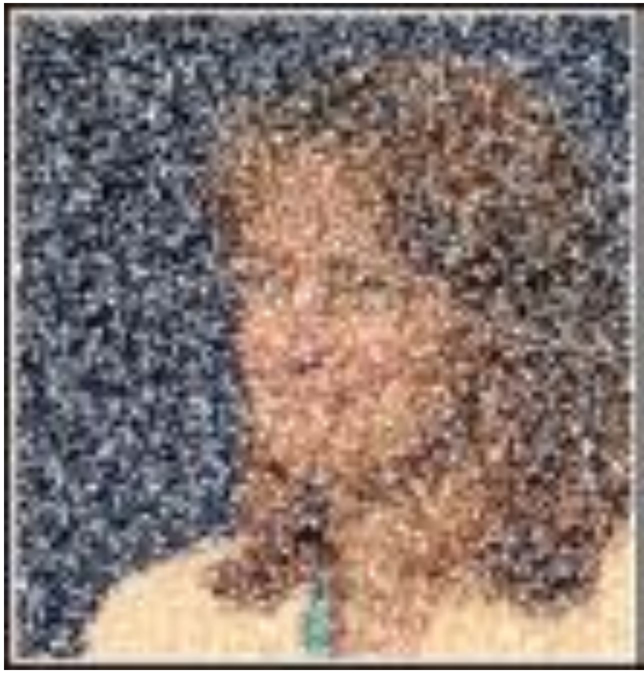
Application of median filtering to image b using a 3x3 square region



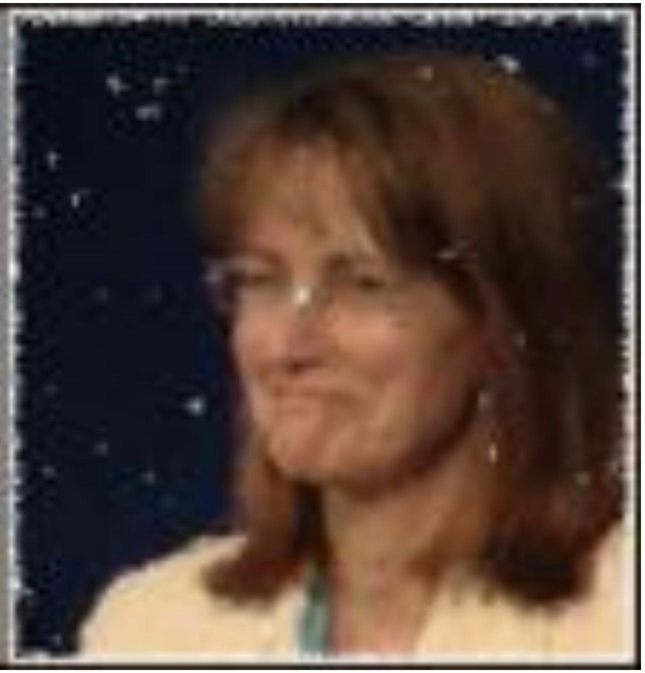


Application of median filtering to image b using a 5x5 square region

Median filter

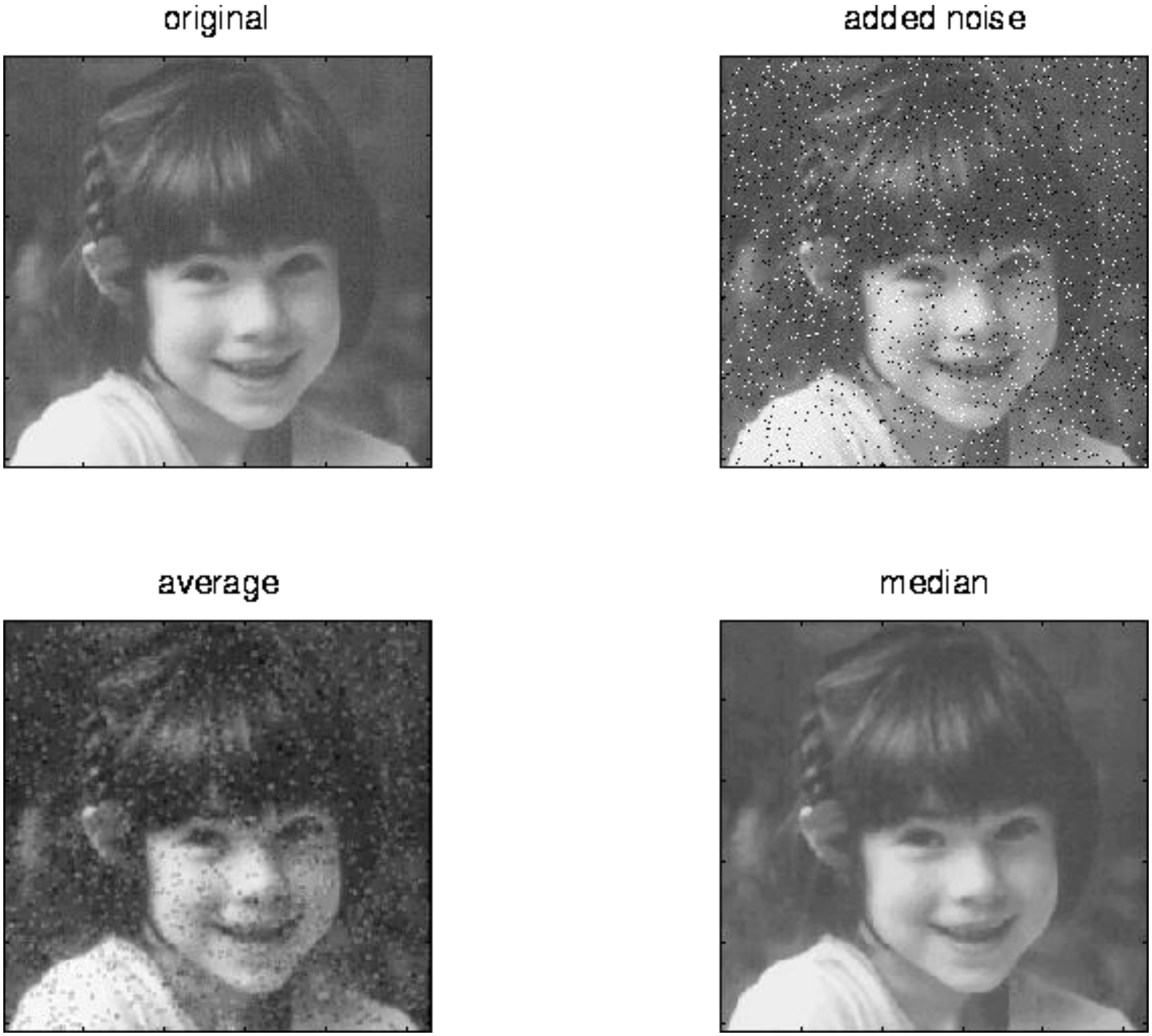


Noisy image



Median filter

Median filter



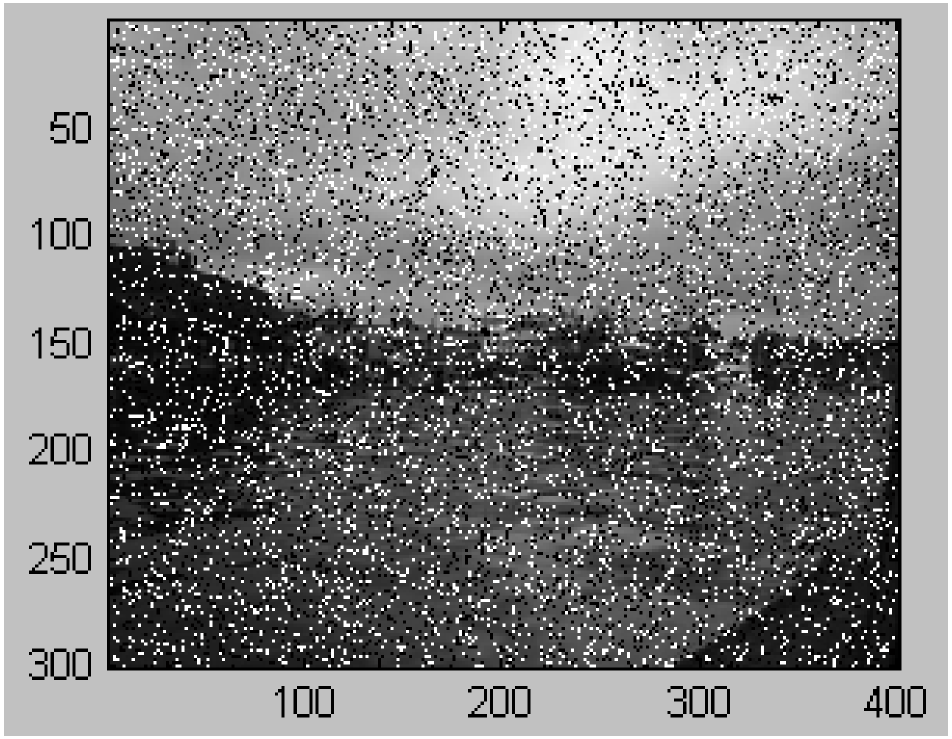
original

added noise

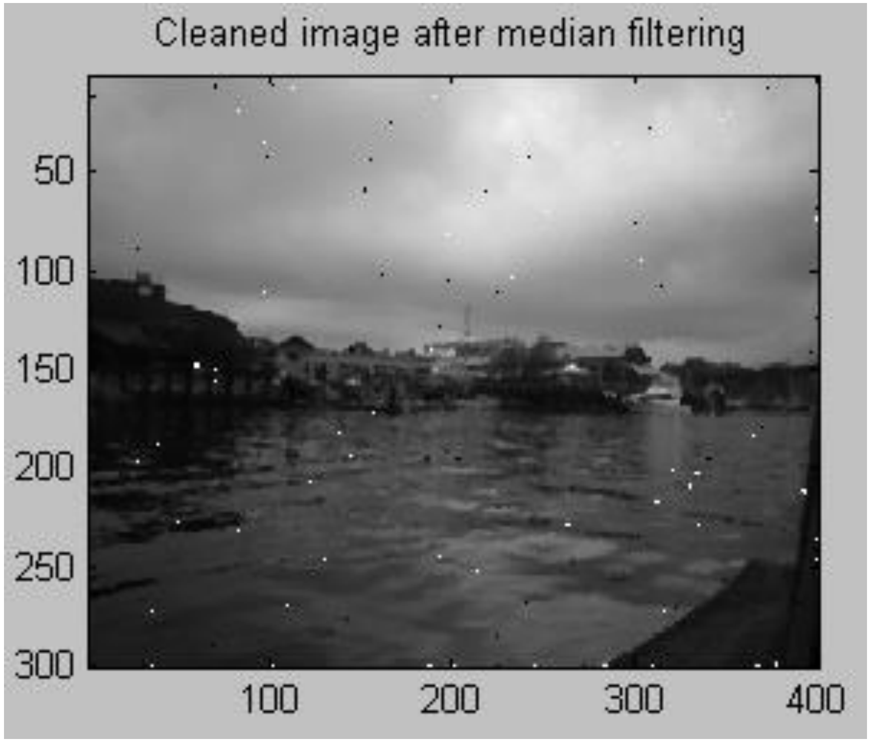
average

median

Median filter

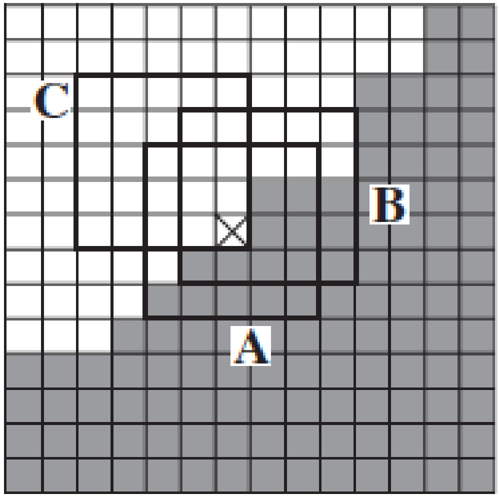


Noisy image
Can you guess what it is?



Median filter

Edge preserving filtering

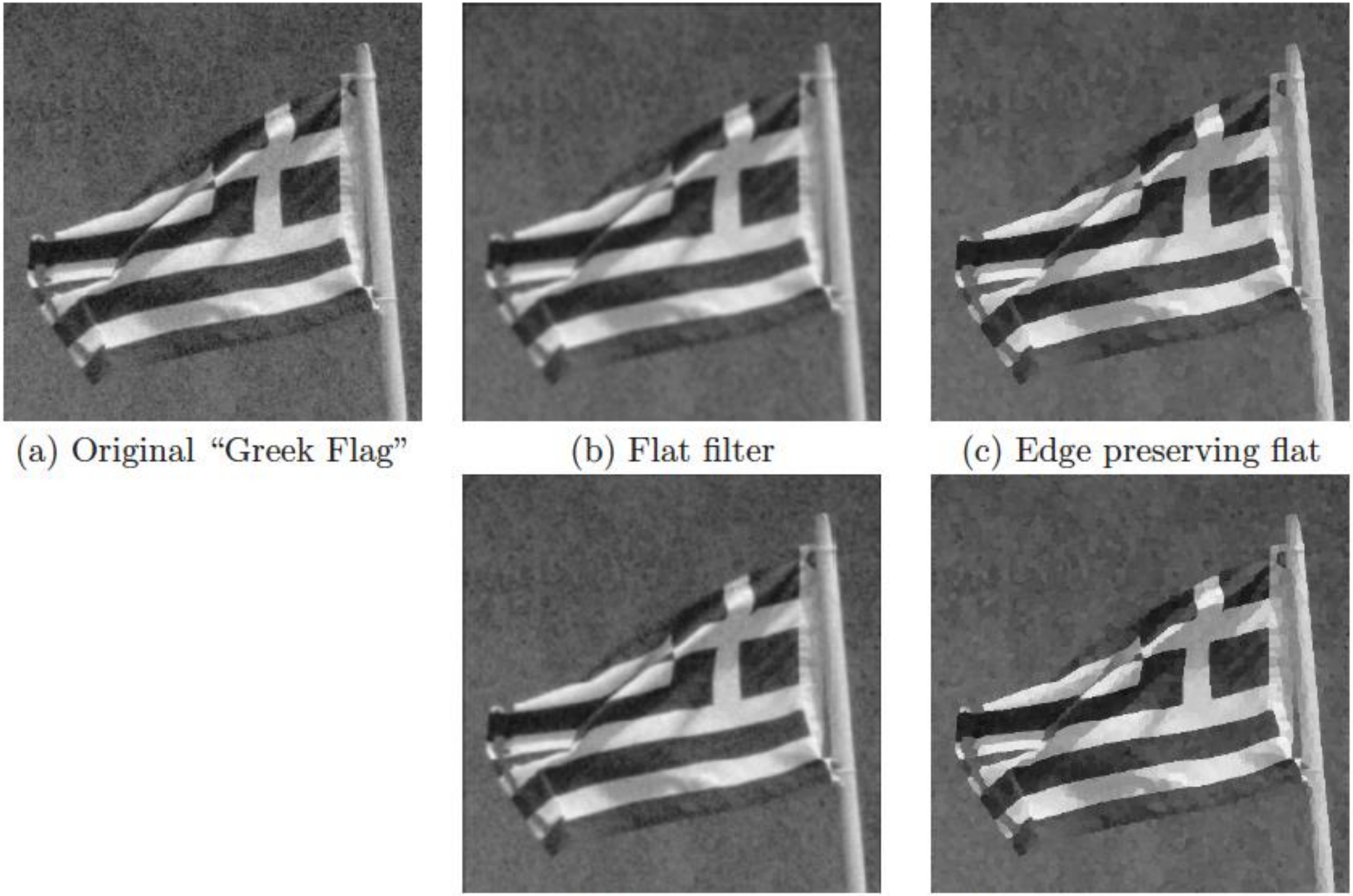


Step 1: Consider all windows of fixed size around a pixel (not necessarily centered at that pixel)

Step 2: Find a window with the least variance

Step 3: Do a linear filter in that window.

Edge preserving filtering



(a) Original "Greek Flag"

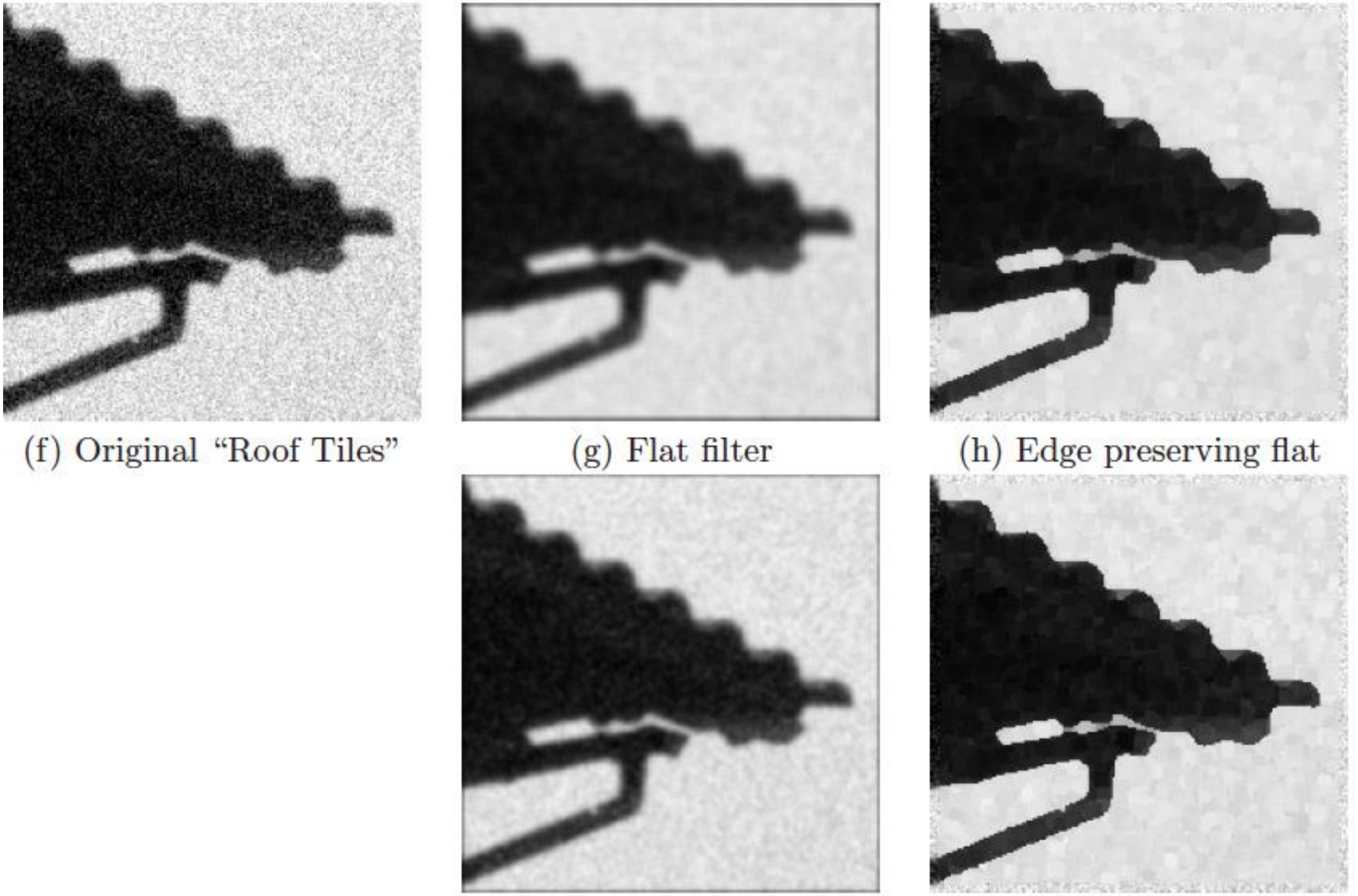
(b) Flat filter

(c) Edge preserving flat

(d) Gaussian filter

(e)Edge preserving Gaussian

Edge preserving filtering



(f) Original “Roof Tiles”

(g) Flat filter

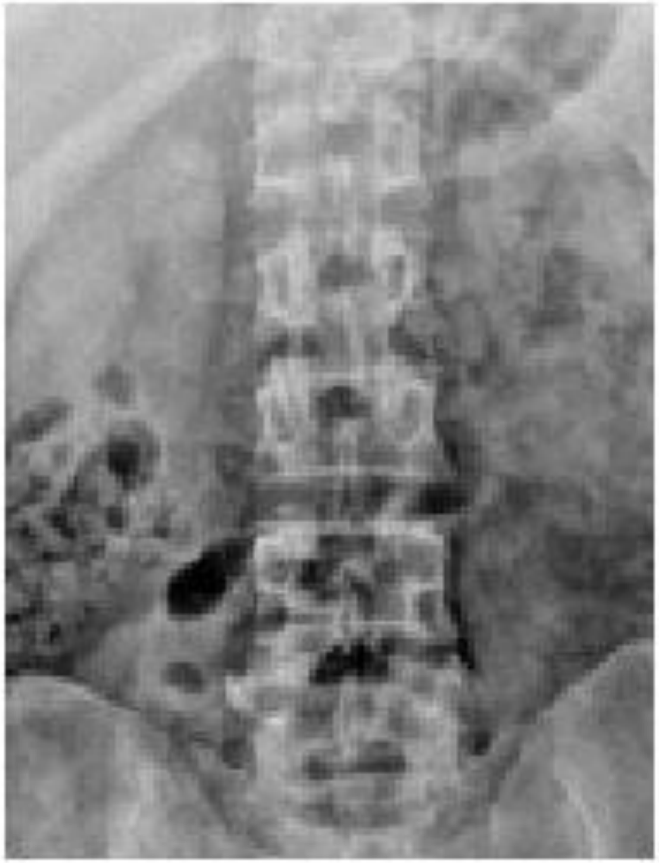
(h) Edge preserving flat

(i) Gaussian filter

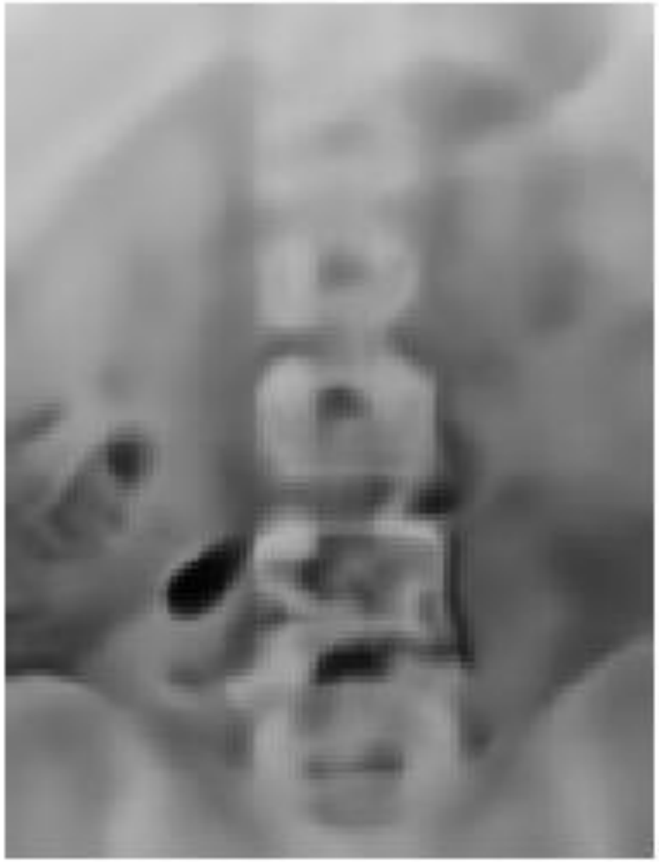
(j)Edge preserving Gaussian

Non-local mean filter

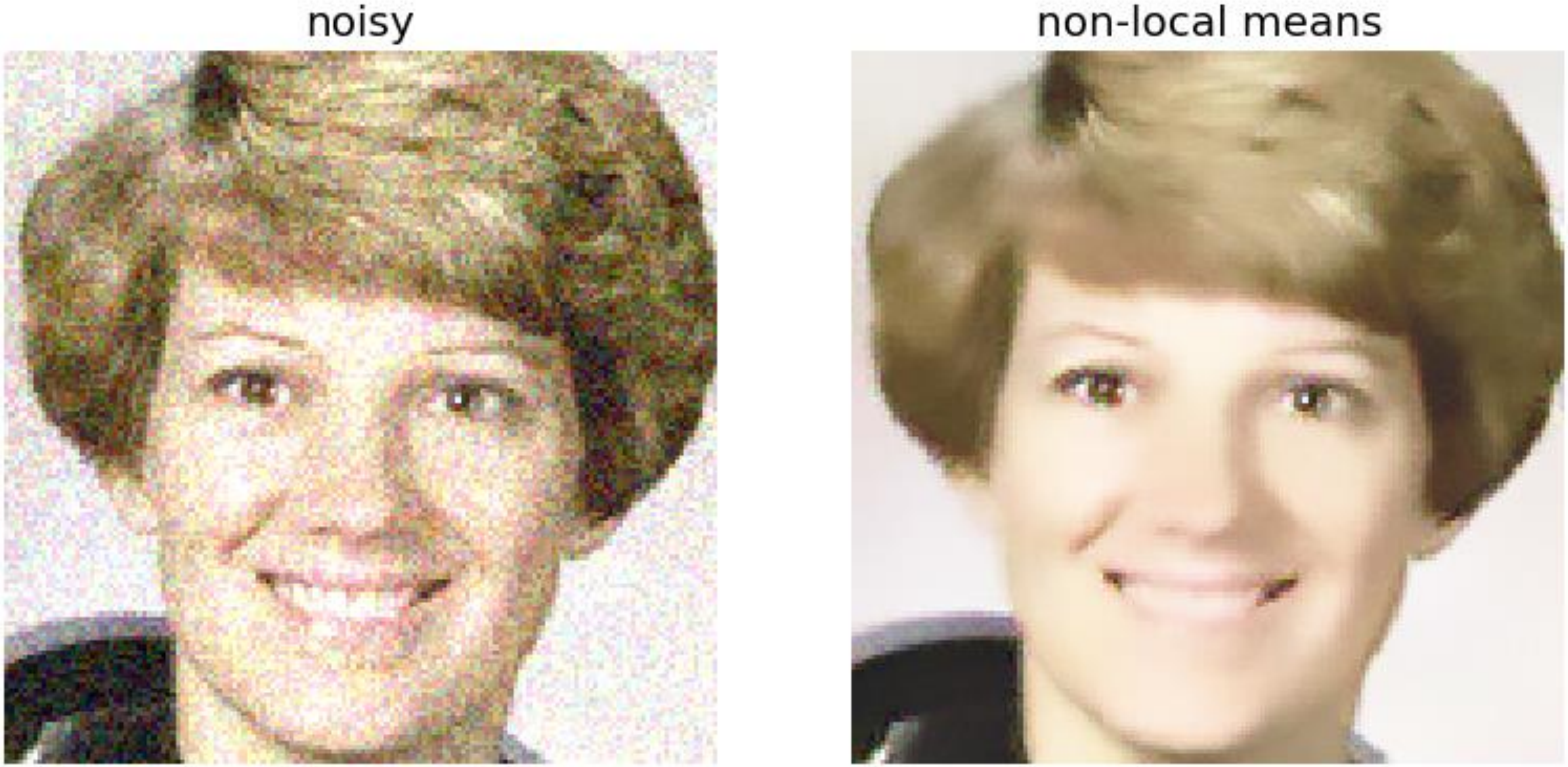
Noisy image



NL-means image



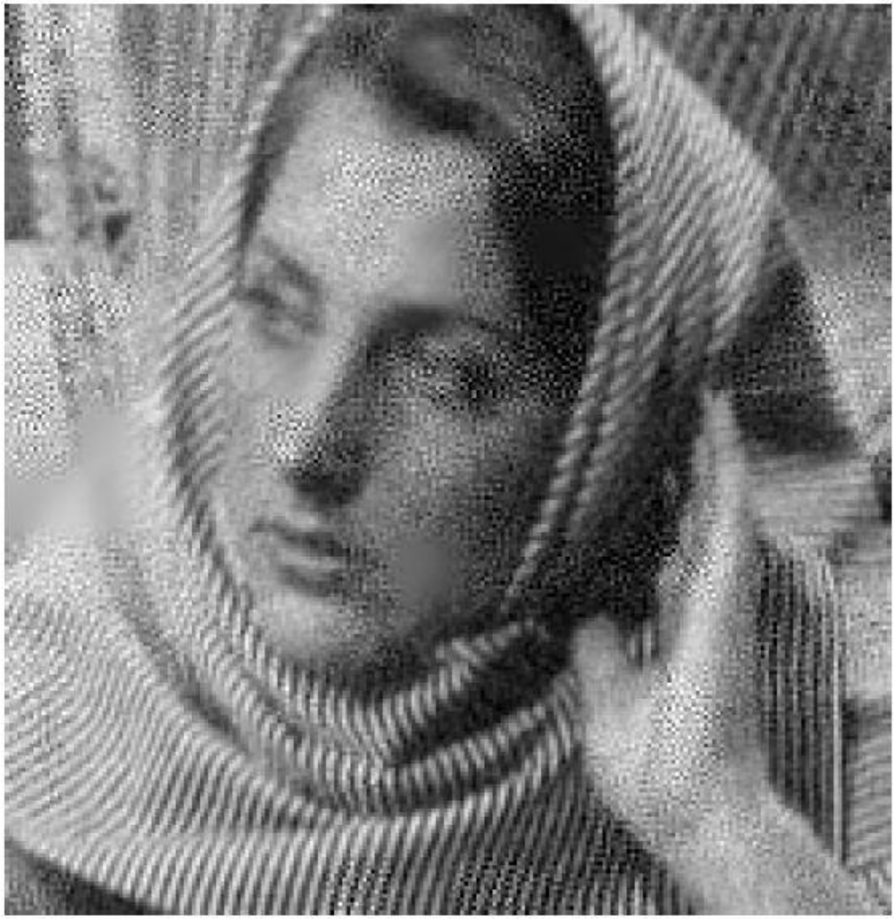
Non-local mean filter



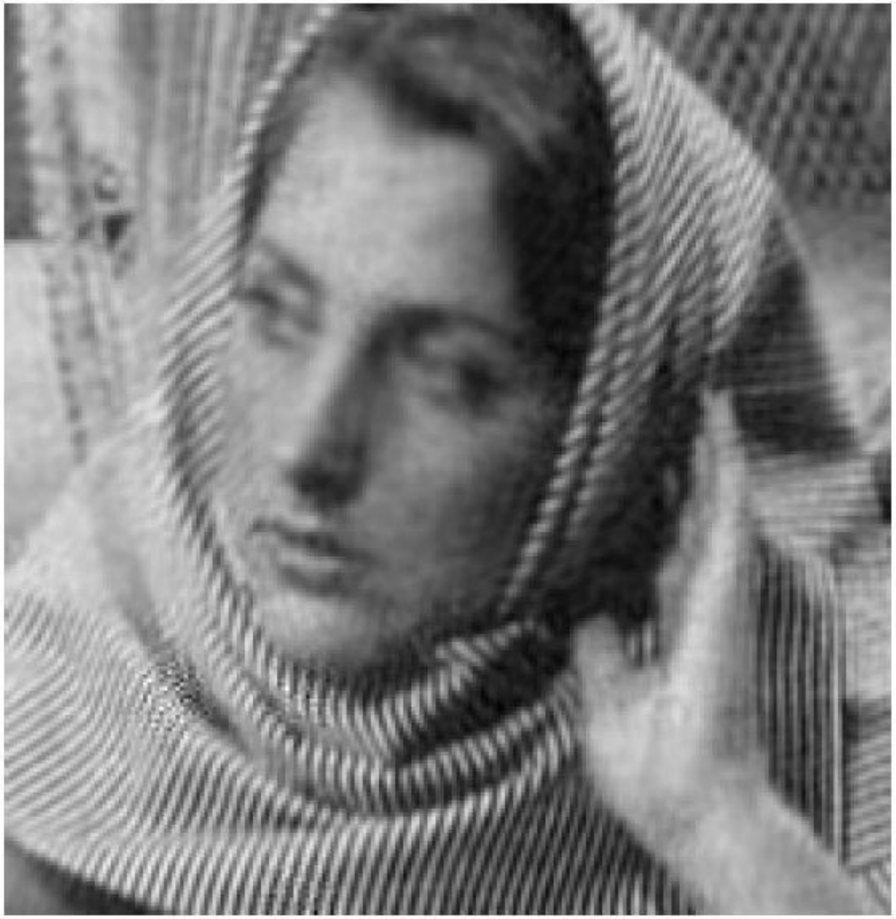
noisy

non-local means

Non-local mean filter

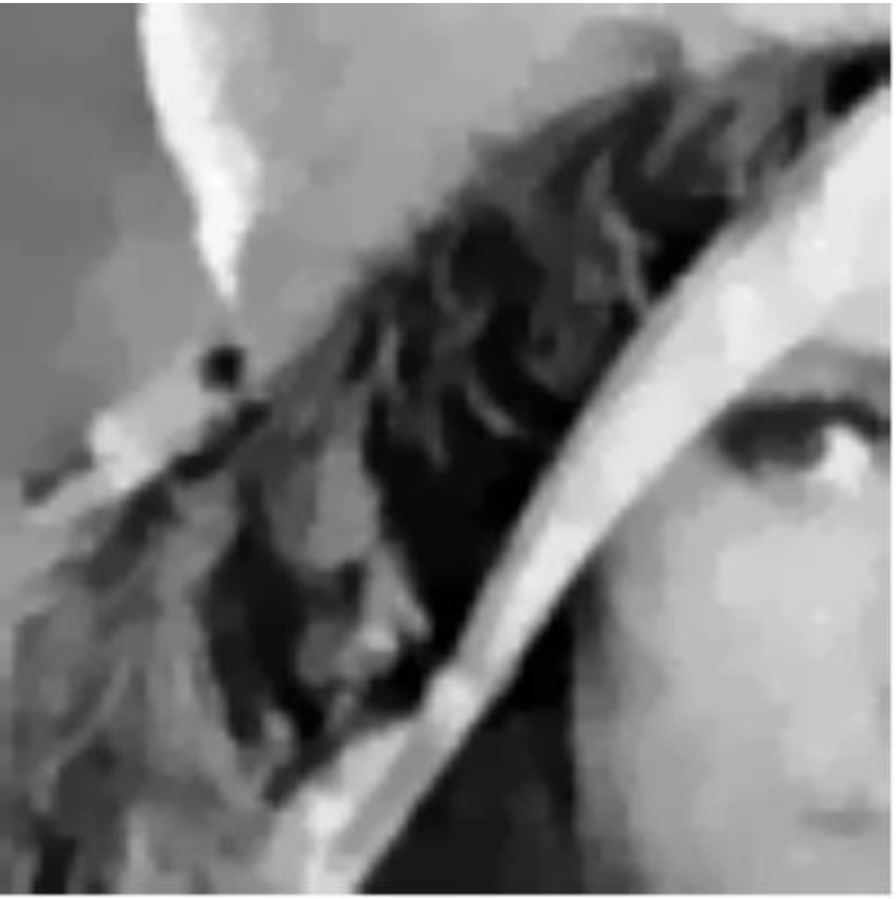


Noisy image



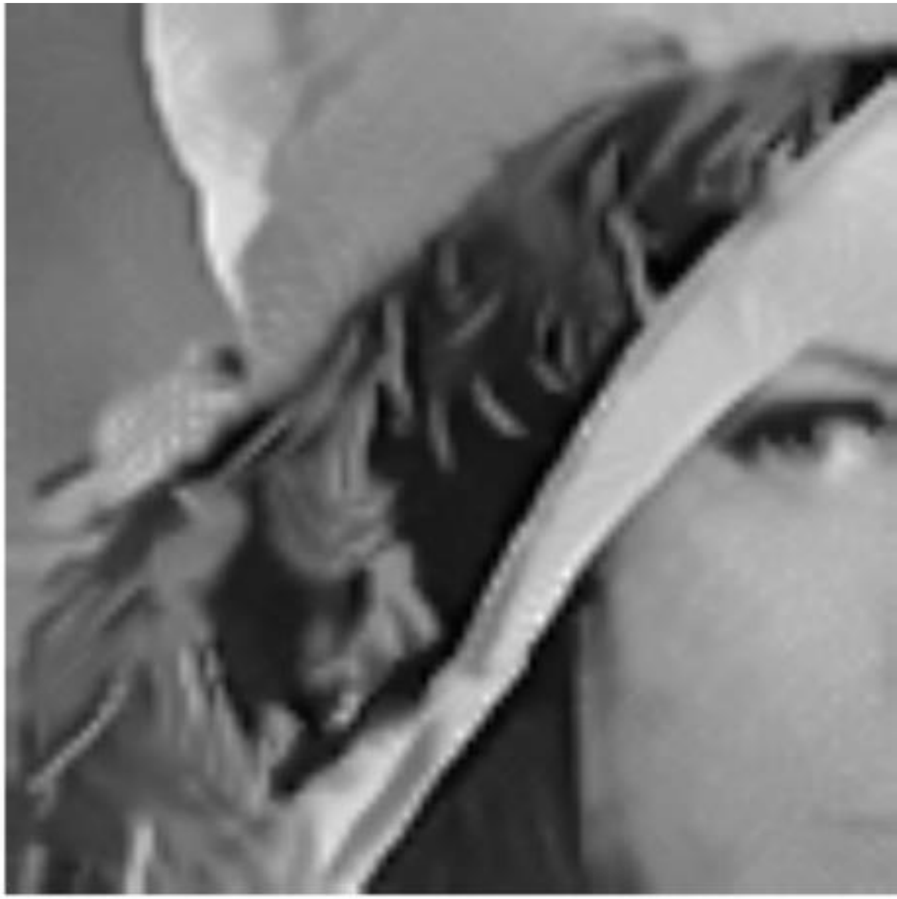
Non-local mean

Non-local mean filter



Flat filter

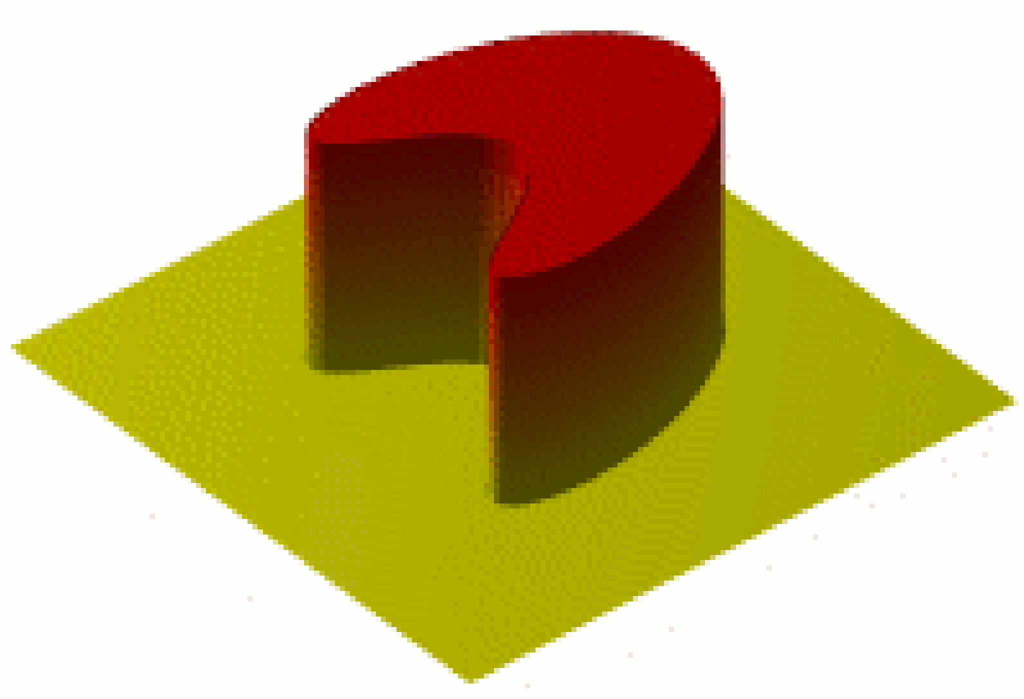
(a)



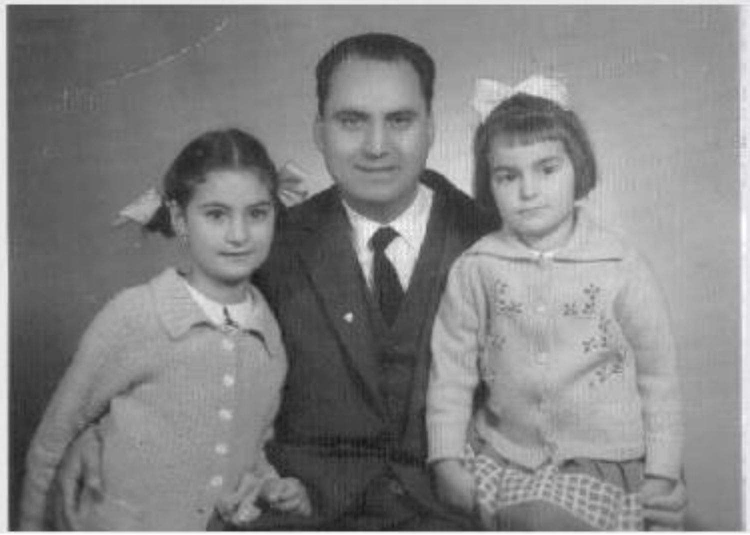
Non-local mean

(b)

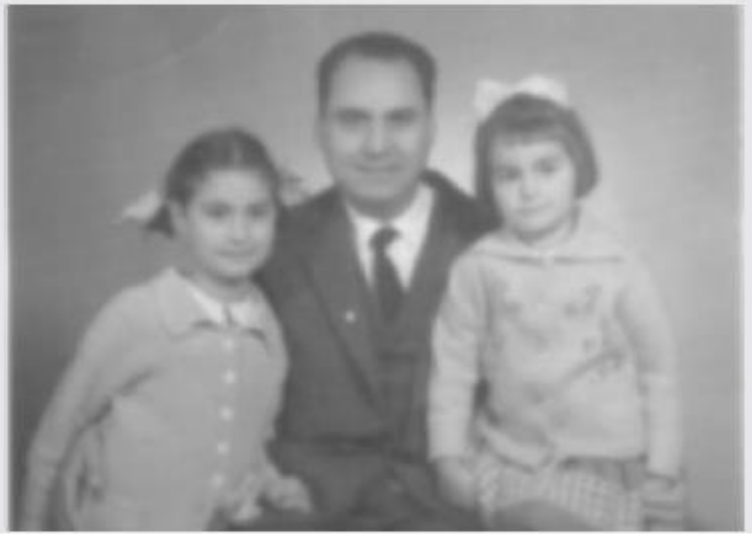
Isotropic diffusion



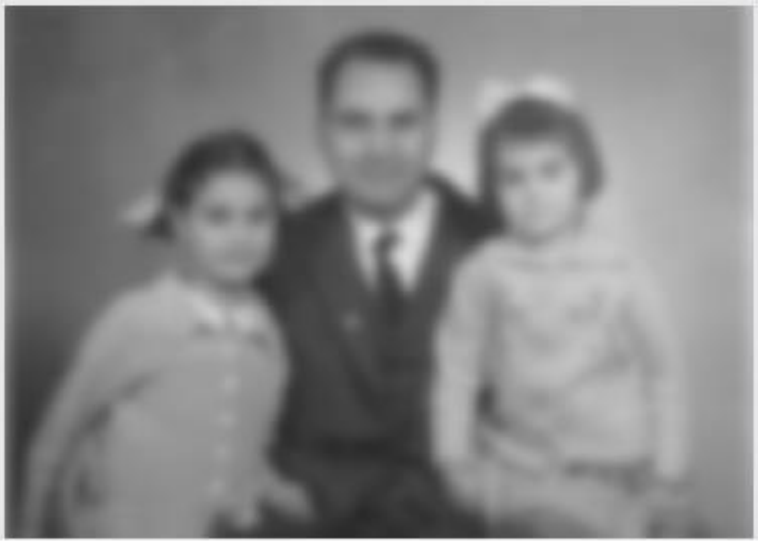
Isotropic diffusion



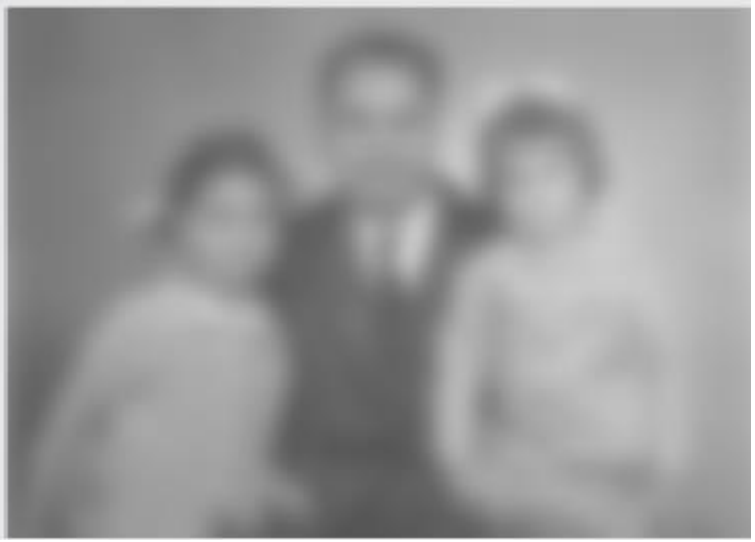
Original image



Sigma = 1.98

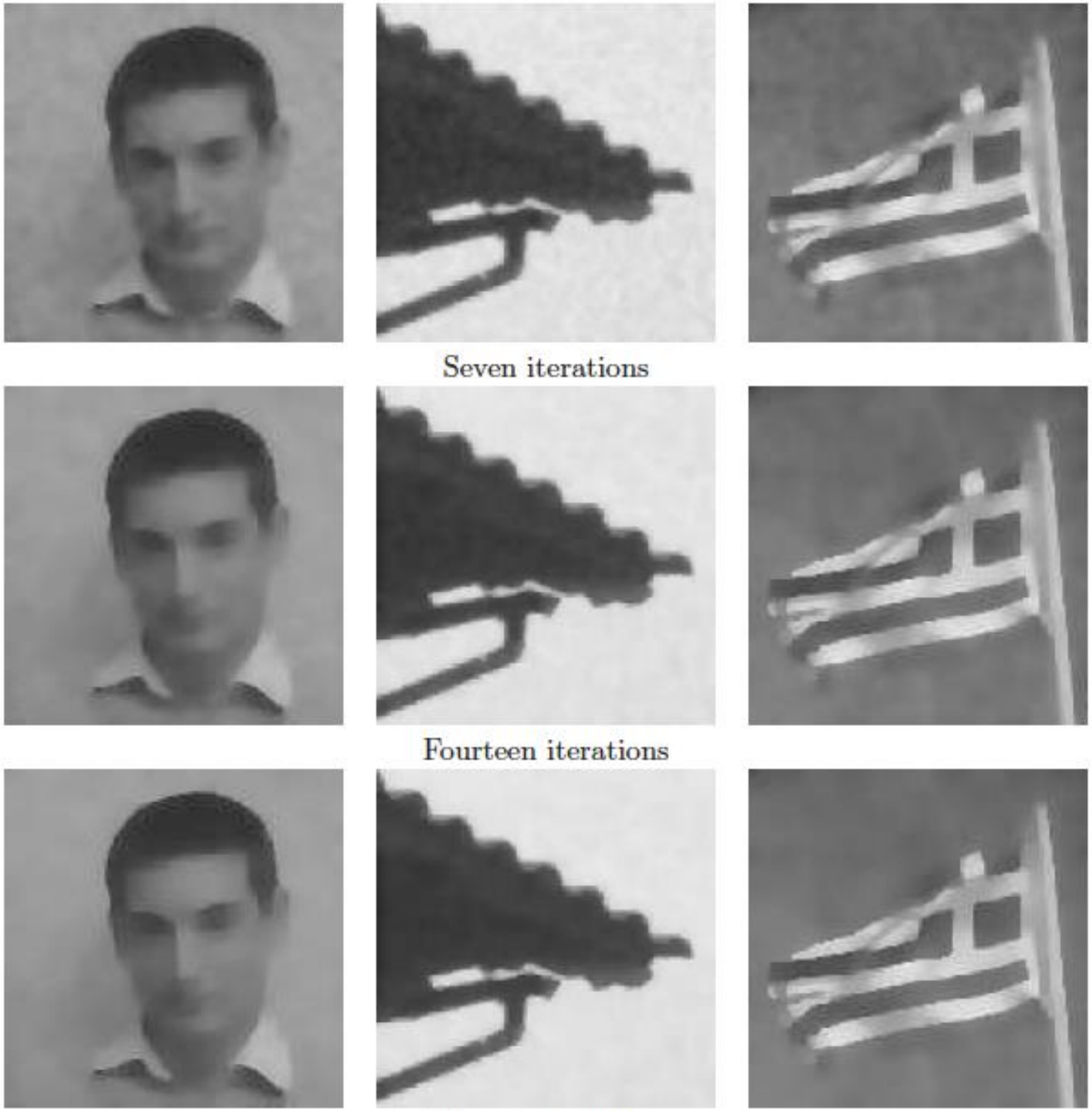


Sigma = 4.28



Sigma = 8.24

Anisotropic diffusion



Seven iterations

Fourteen iterations

Twenty iterations