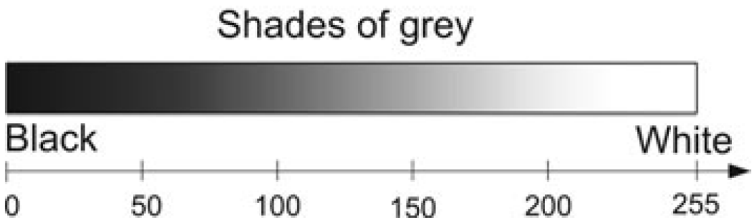
Basics of Machine Vision

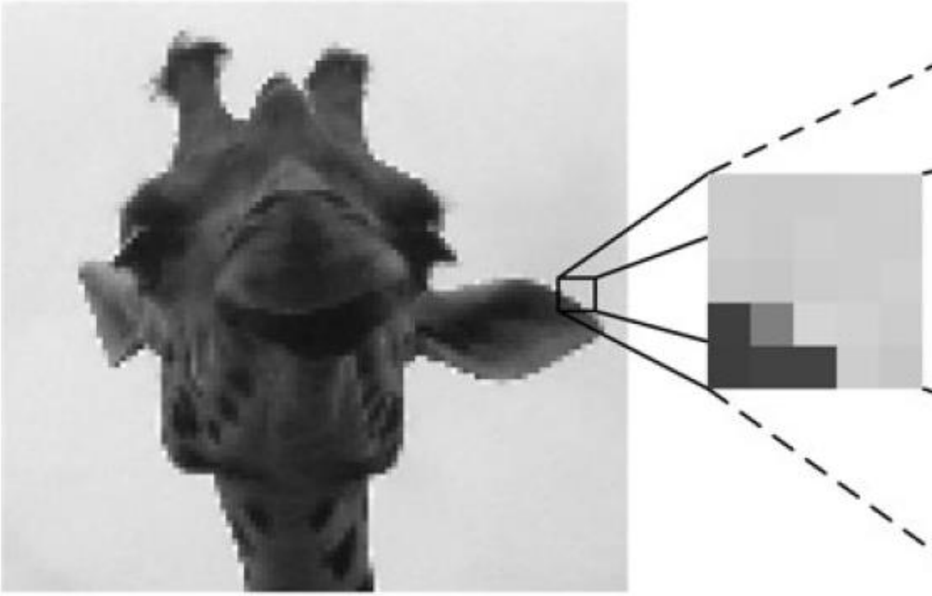
Primož Podržaj

Lecture 03

Image in “mathematical sense”



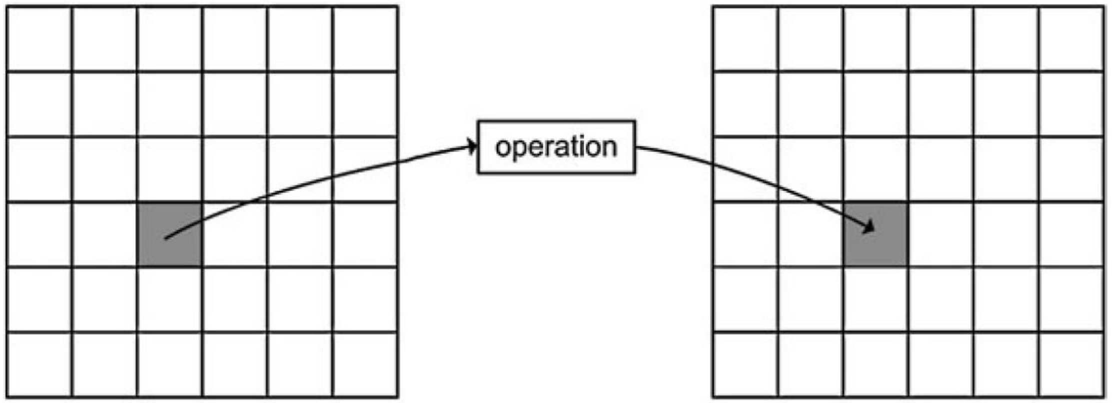
y \ x	0	1	2	3
0	10	20	12	23
1	17	100	25	95
2	9	17	8	22
3	16	89	19	92



205	204	204	206	207
206	203	208	206	206
201	199	205	206	209
61	128	213	210	205
59	65	65	206	199

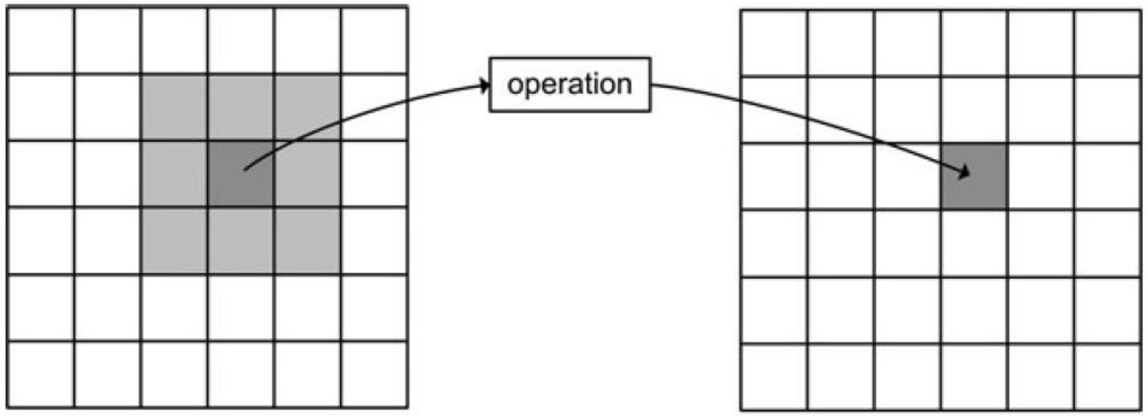
Image manipulation

- Point processing vs neighbourhood processing



Input image
f(x,y)

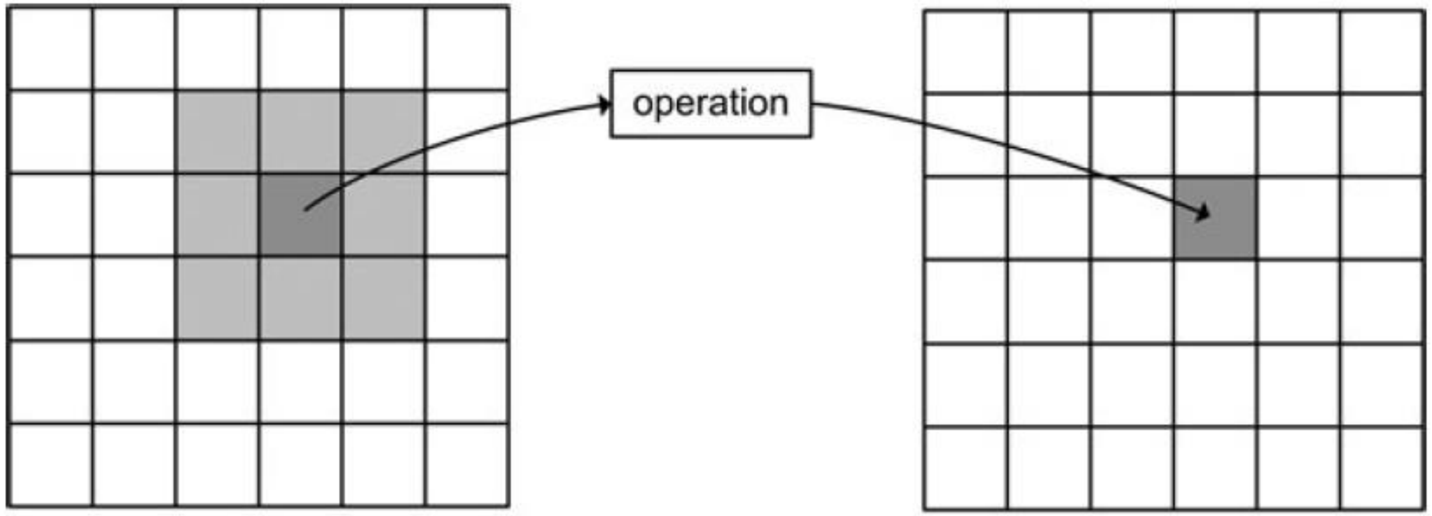
Output image
g(x,y)



Input image
f(x,y)

Output image
g(x,y)

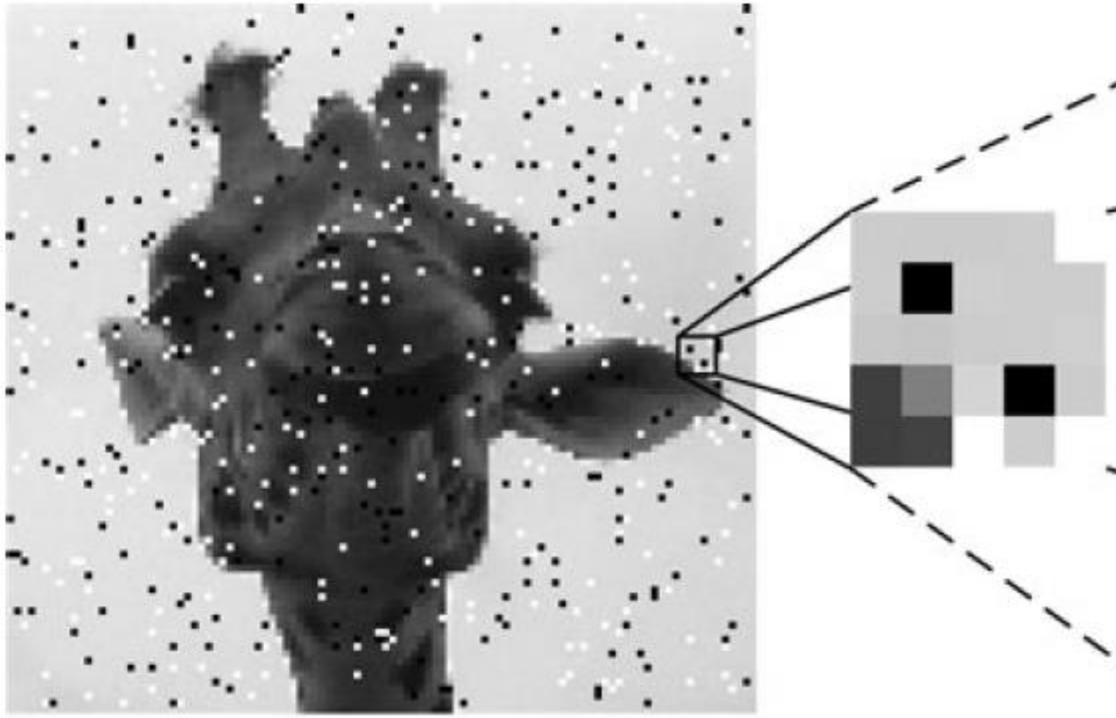
Neighborhood processing



Input image
f(x,y)

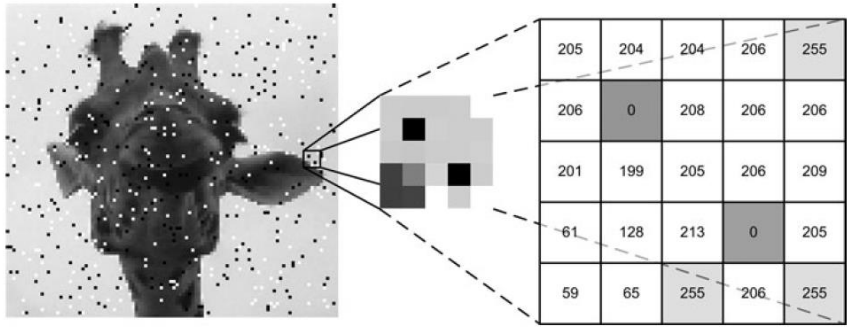
Output image
g(x,y)

Filtering



205	204	204	206	255
206	0	208	206	206
201	199	205	206	209
61	128	213	0	205
59	65	255	206	255

Mean and median filter



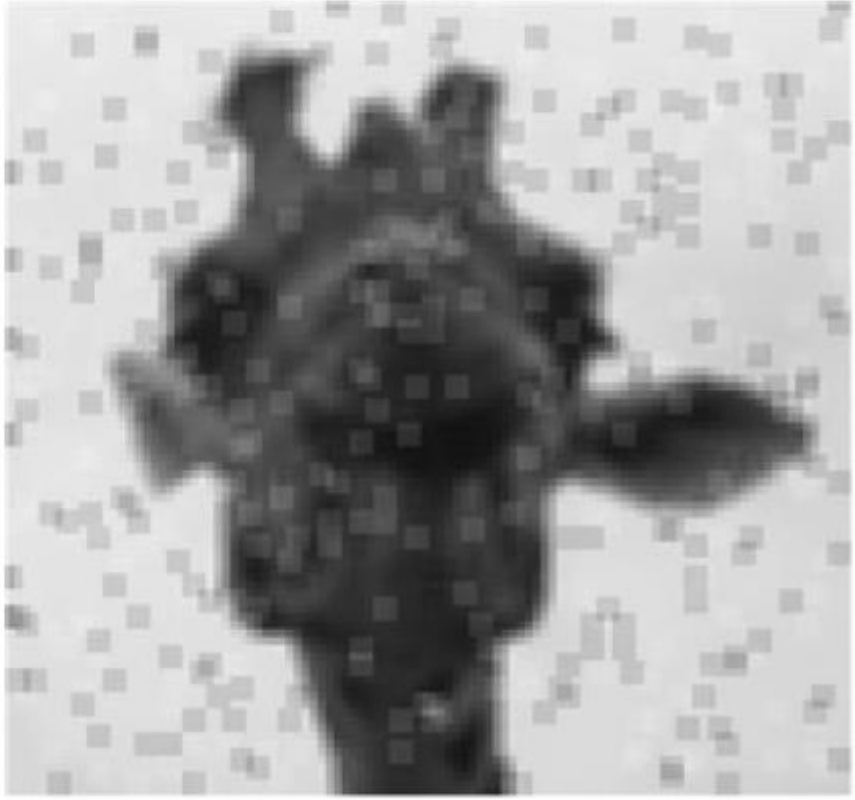
- Mean

$$\text{Mean value} = \frac{205 + 204 + 204 + 206 + 0 + 208 + 201 + 199 + 205}{9}$$
$$= 181.3 \simeq 181$$

- Median

$$\text{Ordering} : [0,\ 199,\ 201,\ 204,\ \underline{\mathit{204}},\ 205,\ 205,\ 206,\ 208]$$
$$\text{Median} = 204$$

Result



Mean filtered

Median filtered

Border problem

- **Increase the output image**

After the output image has been generated, the pixel values in the last row (if radius = 1) is duplicated and appended to the image. The same for the first row, first column and last column.

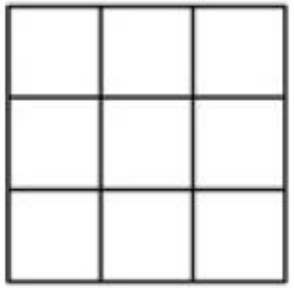
- **Increase the input image**

Before the image is filtered the pixel values in the last row (if radius = 1) of the input image is duplicated and appended to the input image. The same for the first row, first column and last column.

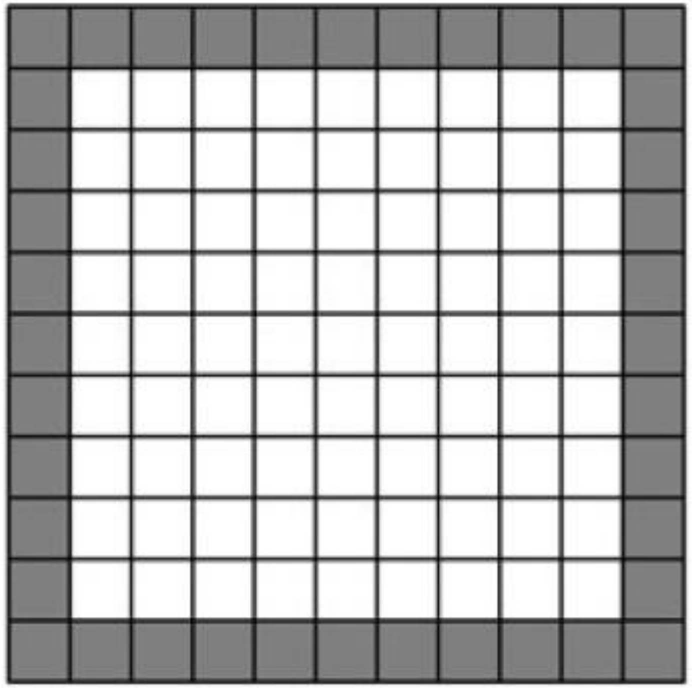
- **Apply special filters at the rim of the image**

Special filters with special sizes are defined and applied accordingly

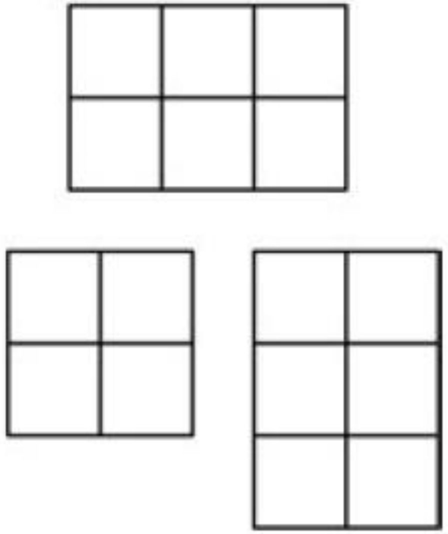
Special fileters



Kernel size 3x3



f(x,y)



Special kernel sizes

Application

Input image 11x11 kernel 29x29 kernel

Rank filters

The Median Filter belongs to a group of filters known as Rank Filters. The only difference between them is the value which is picked after the pixels have been sorted:

- **The minimum value**

This filter will make the image darker.

- **The maximum value**

This filter will make the image brighter.

- **The difference**

This filter outputs the difference between the maximum and minimum value and the result is an image where the transitions between light and dark (and opposite) are enhanced. Such a transition is often denoted an edge in an image.

Correlation

Correlation is an operation which also works by scanning through the image and applying a filter to each pixel. In correlation, however, the filter is denoted a **kernel** and plays a more active role. First of all the kernel is filled by numbers—denoted kernel coefficients.

1	1	1
1	1	1
1	1	1

3x3 Mean kernel

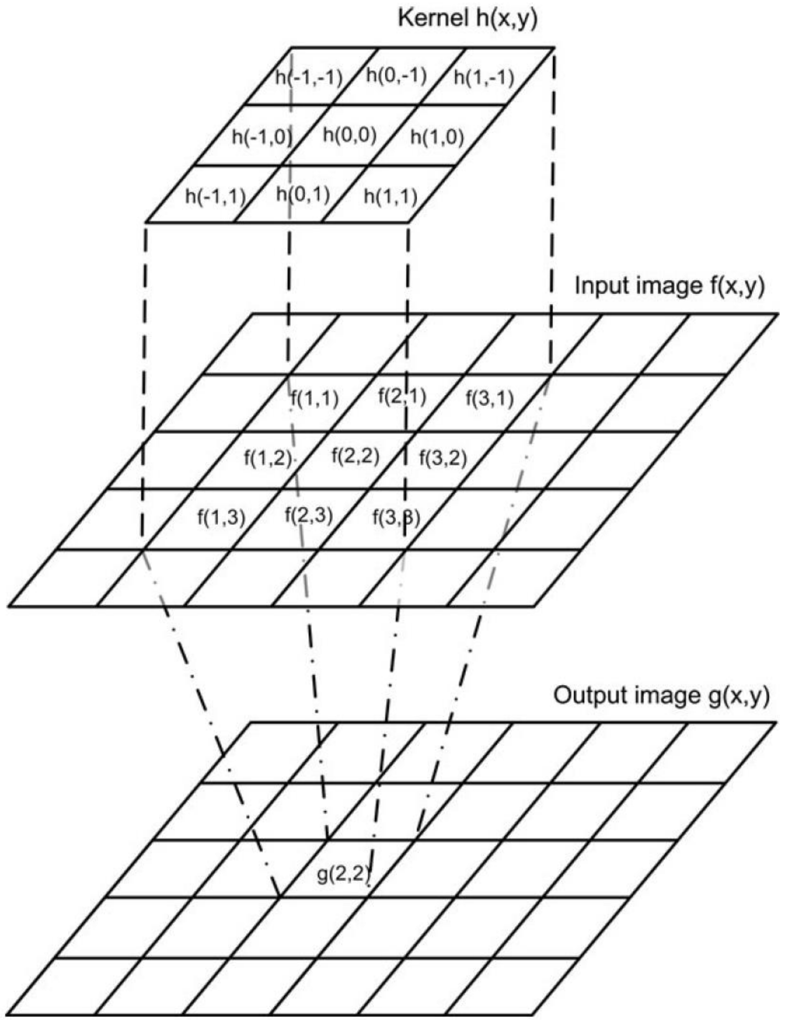
1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

5x5 Gaussian kernel

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

3x3 Sobel kernel

Graphical representation



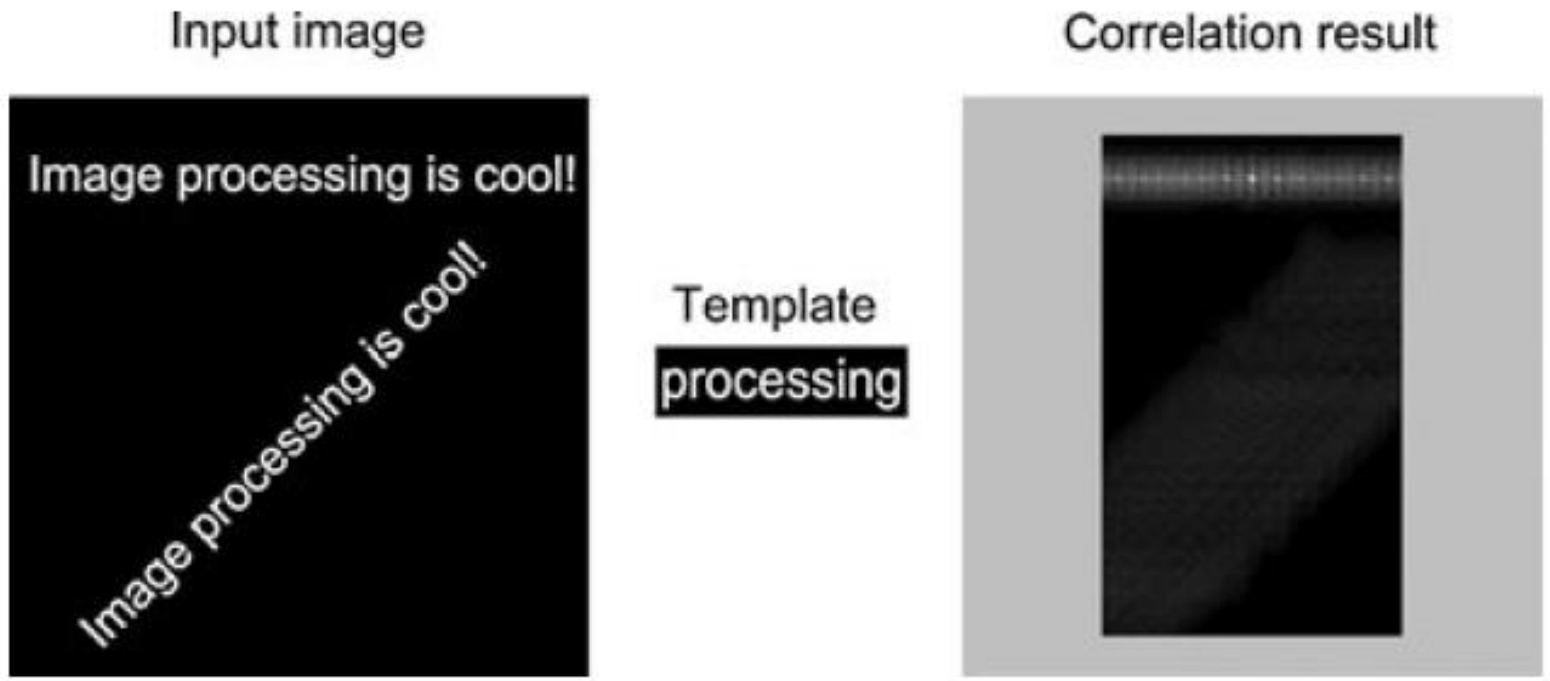
An example for one pixel and general formula

- $g(2, 2) = h(-1,-1) \cdot f(1, 1) + h(0,-1) \cdot f(2, 1) + h(1,-1) \cdot f(3, 1) + h(-1, 0) \cdot f(1, 2) + h(0, 0) \cdot f(2, 2) + h(1, 0) \cdot f(3, 2) + h(-1, 1) \cdot f(1, 3) + h(0, 1) \cdot f(2, 3) + h(1, 1) \cdot f(3, 3)$

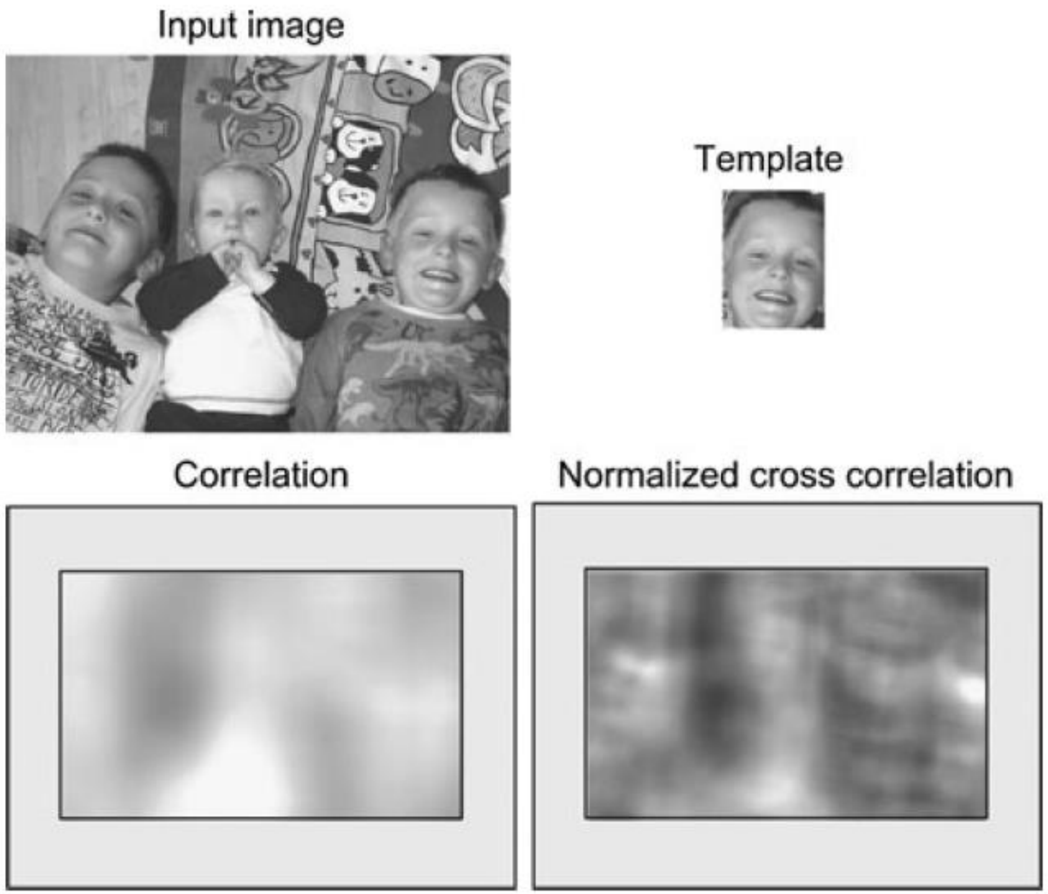
$$g(x, y) = \sum_{j=-R}^{R} \sum_{i=-R}^{R} h(i, j) \cdot f(x + i, y + j)$$

Template matching

Template matching is used to locate an object in an image. When applying template matching the kernel is denoted a **template**.



Example


Input image
Template
Correlation
Normalized cross correlation

Normalized cross-correlation (NCC)

$$\cos\theta = \frac{\vec{H} \bullet \vec{F}}{|\vec{H}| \cdot |\vec{F}|}$$

$$\text{Length of template} = \sqrt{\sum_{j=-R}^{R} \sum_{i=-R}^{R} h(i,j) \cdot h(i,j)}$$

$$\text{NCC}(x,y) = \frac{\text{Correlation}}{\text{Length of image patch} \cdot \text{Length of template}} \quad \Rightarrow$$

$$\text{NCC}(x,y) = \frac{\sum_{j=-R}^{R} \sum_{i=-R}^{R} (H \cdot F)}{\sqrt{\sum_{j=-R}^{R} \sum_{i=-R}^{R} (F \cdot F)} \cdot \sqrt{\sum_{j=-R}^{R} \sum_{i=-R}^{R} (H \cdot H)}}$$

Zero-mean normalized cross-correlation (ZMNCC)

- Here the mean values of the template and image patch are subtracted from H and F, respectively. This is known as the zero-mean normalized cross-correlation or the correlation coefficient. The output is in the interval $[-1, 1]$ where 1 indicates a maximum similarity (as for NCC) and -1 indicates a maximum negative similarity, meaning the same pattern but opposite gray-scale values: 255 instead of 0, 254 instead of 1, etc.

OpenCV possibilities

1. **method=TM_SQDIFF**

$$R(x,y)=\sum_{x',y'}(T(x',y')-I(x+x',y+y'))^2$$

2. **method=TM_SQDIFF_NORMED**

$$R(x,y)=\frac{\sum_{x',y'}(T(x',y')-I(x+x',y+y'))^2}{\sqrt{\sum_{x',y'}T(x',y')^2\cdot\sum_{x',y'}I(x+x',y+y')^2}}$$

3. **method=TM_CCORR**

$$R(x,y)=\sum_{x',y'}(T(x',y')\cdot I(x+x',y+y'))$$

4. **method=TM_CCORR_NORMED**

$$R(x,y)=\frac{\sum_{x',y'}(T(x',y')\cdot I(x+x',y+y'))}{\sqrt{\sum_{x',y'}T(x',y')^2\cdot\sum_{x',y'}I(x+x',y+y')^2}}$$

5. **method=TM_CCOEFF**

$$R(x,y)=\sum_{x',y'}(T'(x',y')\cdot I'(x+x',y+y'))$$

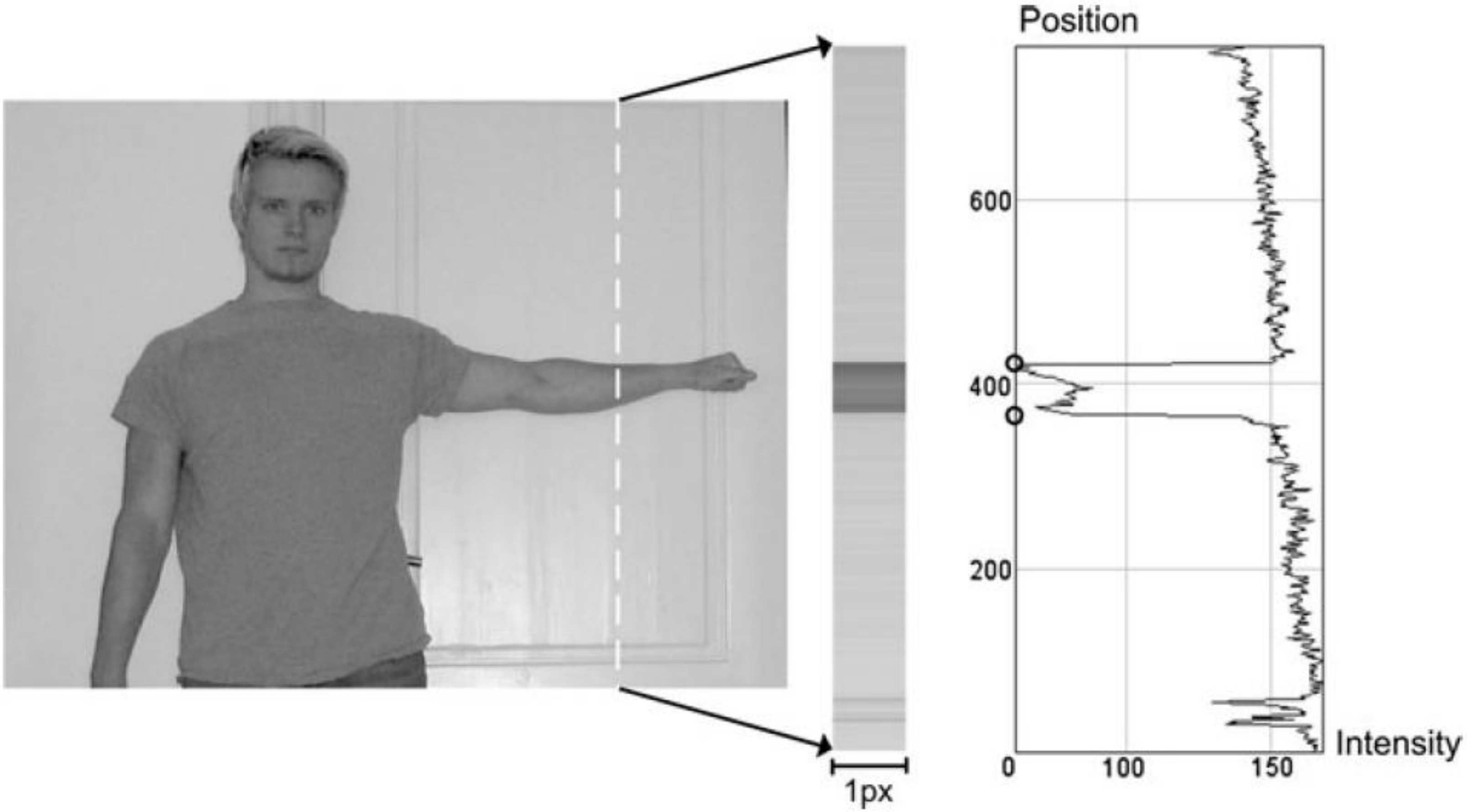
where

$$T'(x',y')=T(x',y')-1/(w\cdot h)\cdot\sum_{x'',y''}T(x'',y'')$$
$$I'(x+x',y+y')=I(x+x',y+y')-1/(w\cdot h)\cdot\sum_{x'',y''}I(x+x'',y+y'')$$

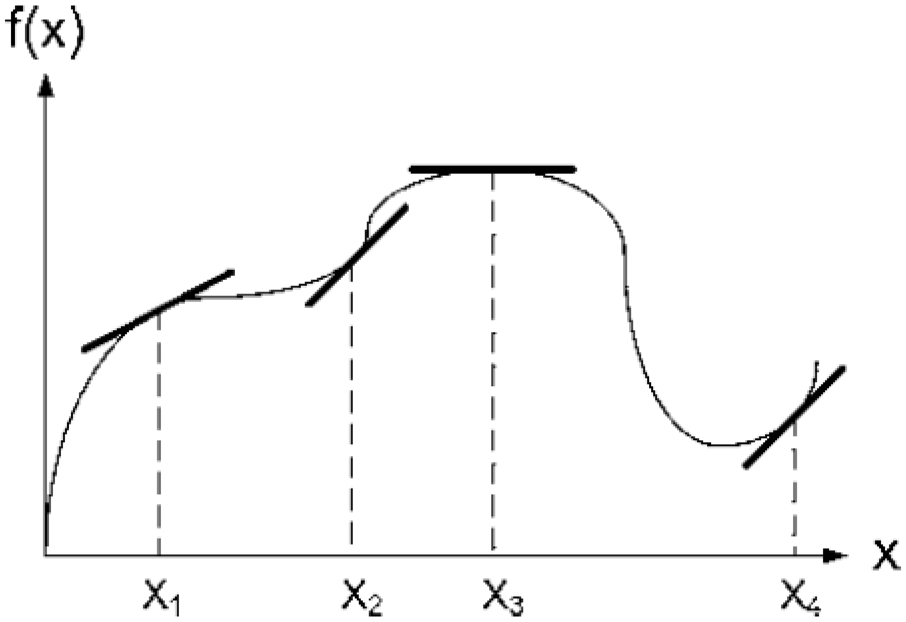
6. **method=TM_CCOEFF_NORMED**

$$R(x,y)=\frac{\sum_{x',y'}(T'(x',y')\cdot I'(x+x',y+y'))}{\sqrt{\sum_{x',y'}T'(x',y')^2\cdot\sum_{x',y'}I'(x+x',y+y')^2}}$$

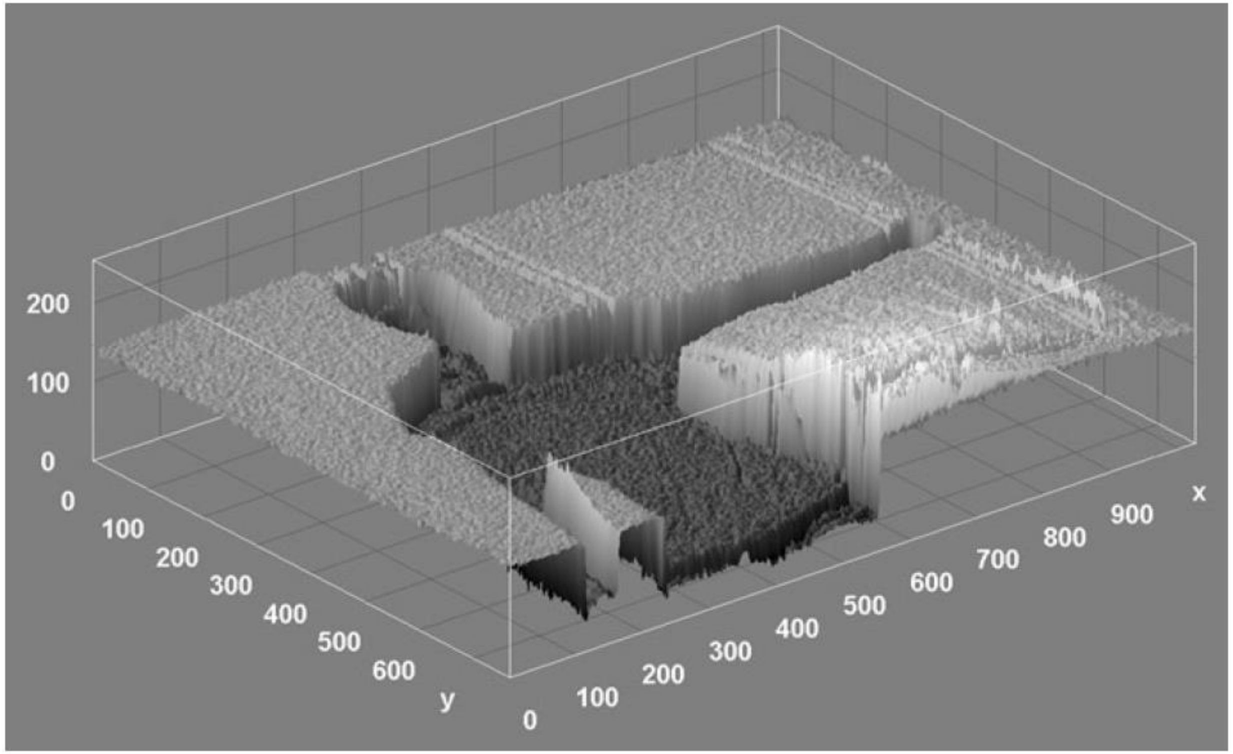
Edge detection



Gradients



f(x)
x
X1
X2
X3
X4



200
100
0
0
100
200
300
400
500
600
y
0
100
200
300
400
500
600
700
800
900
x

Image edges

- Gradient appoximations:

$g_x(x, y) \approx f(x + 1, y) - f(x - 1, y)$

$g_y(x, y) \approx f(x, y + 1) - f(x, y - 1)$

$$\text{Magnitude} = \sqrt{g_x^2 + g_y^2}$$

$$\text{Approximated magnitude} = |g_x| + |g_y|$$

Kernels

Prewitt

Vertical

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

Horizontal

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

Sobel

Vertical

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

Horizontal

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

Results

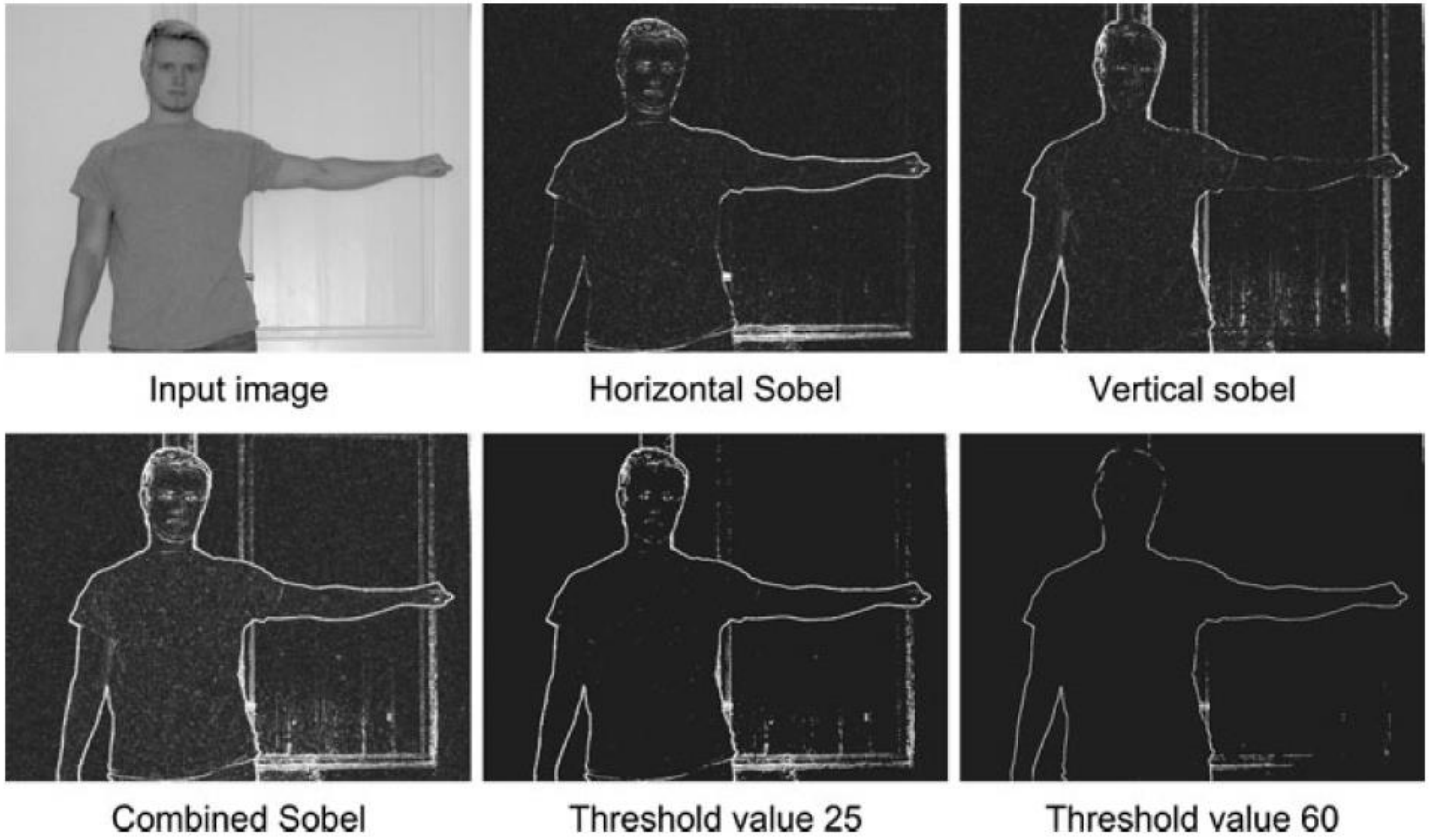
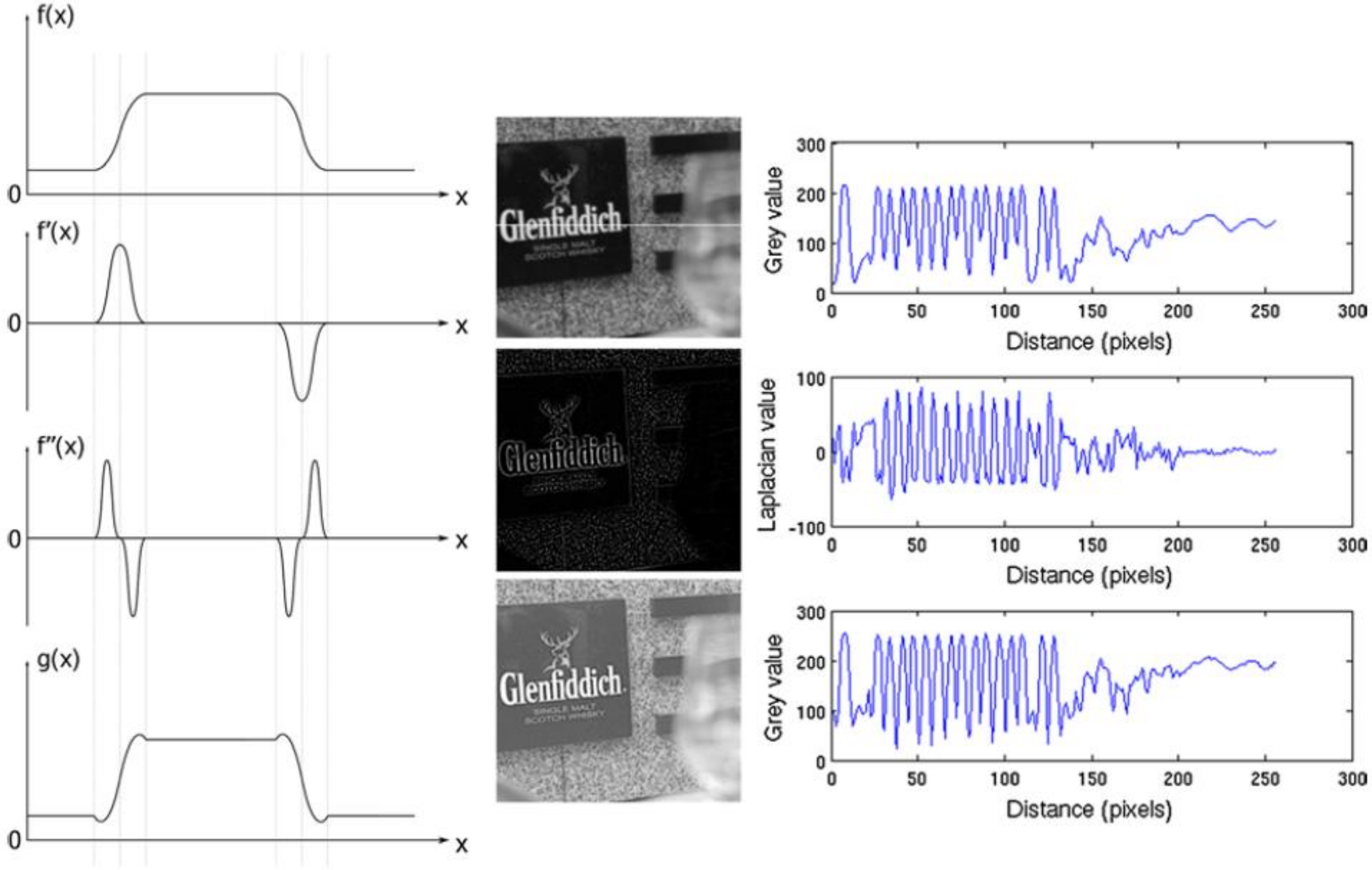

Input image
Horizontal Sobel
Vertical sobel
Combined Sobel
Threshold value 25
Threshold value 60

Image sharpening



(a)

(b)

Equations and kernels

$$g_{xx}(x, y) \approx f(x-1, y) - 2 \cdot f(x, y) + f(x+1, y)$$

$$g_{yy}(x, y) \approx f(x, y-1) - 2 \cdot f(x, y) + f(x, y+1)$$

Vertical

1
-2
1

Horizontal

1	-2	1

Laplacian

0	1	0
1	-4	1
0	1	0

$$g(x, y) = f(x, y) - c(f(x, y) \circ h(x, y))$$