Convolution

$$g(x,y) = h(x,y) * f(x,y)$$

$$f * h = \sum_{s=-a}^{a} \sum_{t=-b}^{b} h(s,t) f(x-s, y-t)$$

f = Image

h = Kernel

h

h_1	h_2	h_3
h_4	h_5	h_6
h_7	h_8	h_9

$X - flip$

h_7	h_8	h_9
h_4	h_5	h_6
h_1	h_2	h_3

$Y - flip$

f

f_1	f_2	f_3
f_4	f_5	f_6
f_7	f_8	f_9

$\otimes$

h_9	h_8	h_7
h_6	h_5	h_4
h_3	h_2	h_1

Sum of Product

$$f * h = f_1h_9 + f_2h_8 + f_3h_7 + f_4h_6 + f_5h_5 + f_6h_4 + f_7h_3 + f_8h_2 + f_9h_1$$

Filtered image generation (Mean Filter Example)

$h[.,.]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Input image f(x,y)

$g[\cdot\,,\cdot\,]$

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

Filtered output image g(x,y)

Spatial Filtering:

with coefficient mask

- Given the 3 × 3 mask with coefficients: w_1, w_2,…, w_9
- The mask cover the pixels with gray levels: z_1, z_2,…, z_9

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Sum of Product

$$z \longleftarrow z_1w_1 + z_2w_2 + z_3w_3 + \cdots + z_9w_9 = \sum_{i=1}^{9} z_i w_i$$

- z gives the output intensity value for the processed image (to be stored in a new array) at the location of z_5 in the input image

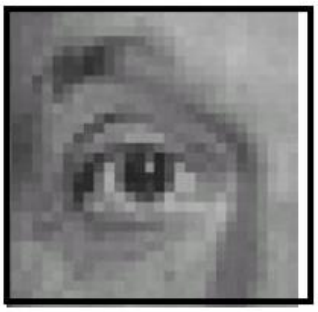
Original

0	0	0
0	1	0
0	0	0

Filtered
(no change)

Original

0	0	0
0	0	1
0	0	0



Shifted left
By 1 pixel

$$* \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \right)$$

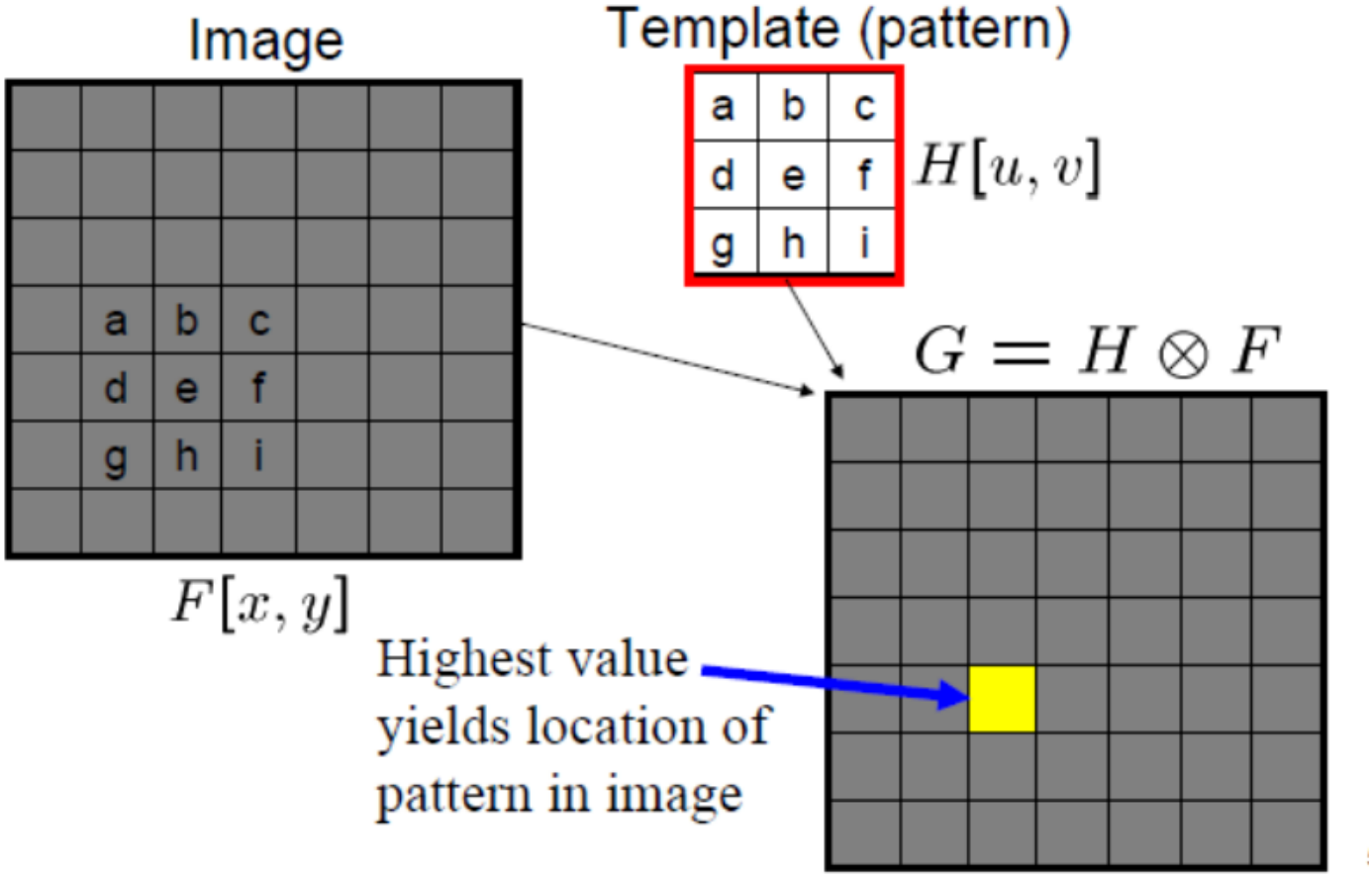
Original

Sharpening filter

-emphasize differences with local average

Cross-correlation and template matching

Cross-correlation is useful for *template matching* (locating a given pattern in an image)



Filtering an impulse

Impulse signal

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	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

$F[x, y]$

Filter Kernel

a	b	c
d	e	f
g	h	i

$H[u, v]$

$$G = H \otimes F$$

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	i	h	g	0	0
0	0	f	e	d	0	0
0	0	c	b	a	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

$G[x, y]$

Output is equal to filter kernel flipped horizontally & vertically

Flipping kernels