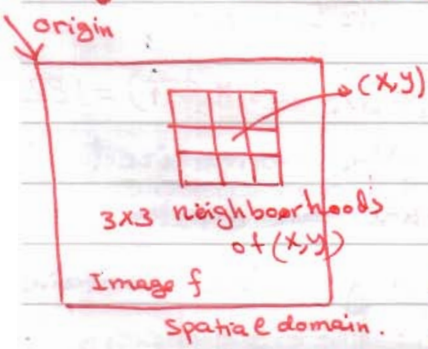
Lect "9"

"Fundamentals of Spatial Filtering"

- Filters in Frequency domain :
  - Low pass filter that passes Low frequencies used for smoothing (blurring) an image
  - High pass
- Spatial Filters using different masks (Kernels, templates or windows)
- there is a one-To-one correspondence between linear spatial filters and filters in frequency domains.
- Spatial filters can be used for linear and nonlinear filtering. (frequency domain filter just for linear filtering)
- The mechanics of spatial Filtering



spatial Filters consists of:

① neighborhood (small rectangle)

② predefined operation that is performed on the image pixel.

Filtering creates new pixel with Coordinates equal to the coordinates of the center of the neighborhood, and whose value is the result of the filtering operation

* if the operation performed on the image pixel is linear, then the filter is called a linear spatial filter, otherwise, the filter is nonlinear.

Figure 1 presents the mechanics of linear spatial filtering using a 3x3 neighborhood.

- the response (output) $g(x,y)$ of the filter at any point (x,y) in the image is the sum of products of the filter coefficients and the image pixels values :

$$g(x,y) = w(-1,-1)\,f(x-1,y-1) + w(-1,0)\,f(x-1,y) + \cdots + w(0,0)\,f(x,y) + \cdots + w(1,1)\cdot f(x+1,y+1).$$

observe that the center coefficient of the filter, $w(0,0)$, aligns with the pixel at location (x,y).

General mask of size $m \times n$:

assume that $m = 2a+1$ and $n = 2b+1$ (where a, b are positive integers). (odd filters).

In general, linear spatial filtering of an image of size $M \times N$ with a filter of size $m \times n$ is given by the expression:

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) \cdot f(x+s, y+t).$$

where x and y are varied so that each pixel in w visits every pixel in f.

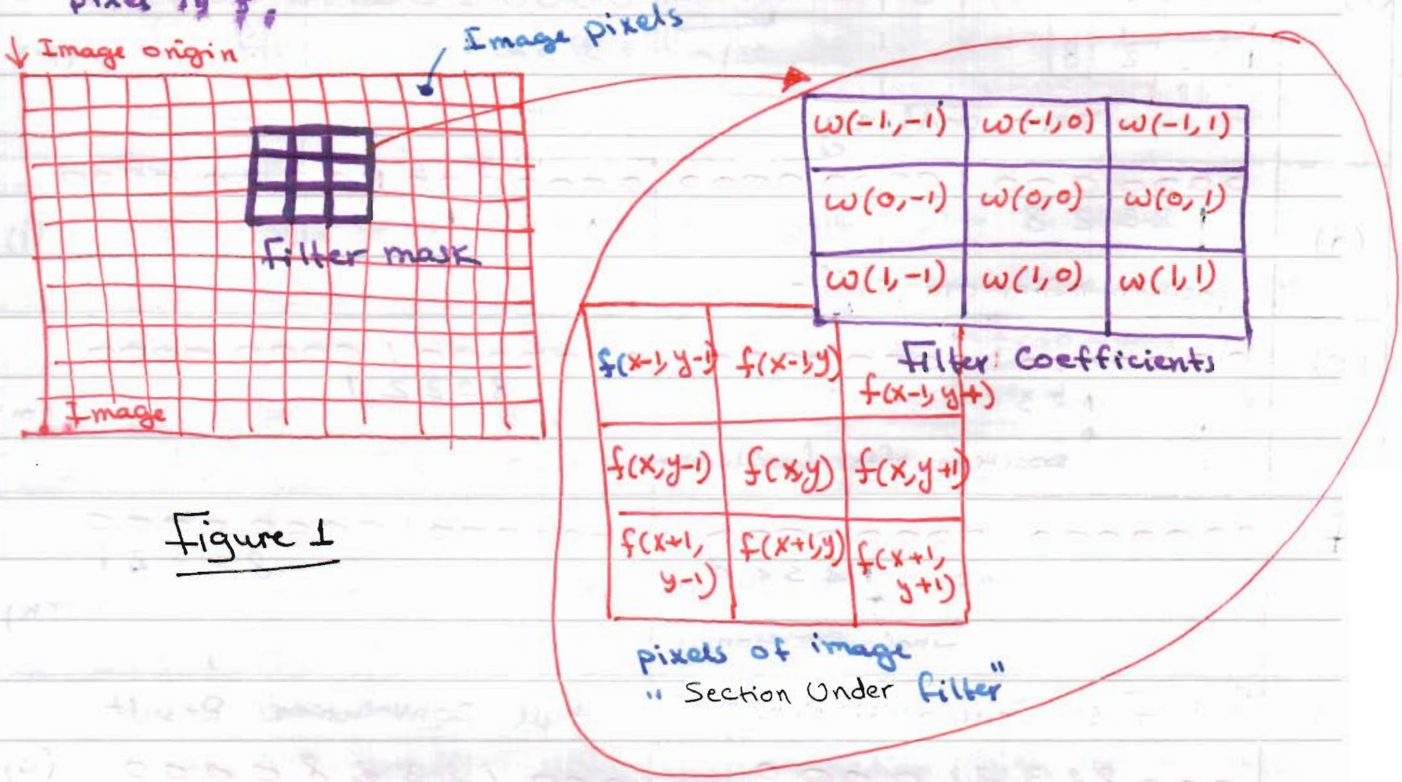


Figure 1

Spatial Correlation and Convolution

Correlation: the process of moving a filter mask over the image and computing the sum of products at each location.

Convolution: the same process as correlation, except that the filter is first rotated by 180°.

example : 1-D illustration : (Figure 2)

assume that f is a 1-D function, and w is a filter

	Correlation		Convolution	
(a)	origin f 0 0 0 1 0 0 0 0	w 1 2 3 2 8	0 0 0 1 0 0 0 0	w rotated 180° 8 2 3 2 1 (i)
(b)	0 0 0 1 0 0 0 0 1 2 3 2 8 Starting point alignment		0 0 0 1 0 0 0 0 8 2 3 2 1	(j)
(c)	0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 2 3 2 $\boxed{8}$ zero padding		0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 8 2 3 2 1	(k)
(d)	0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 2 3 2 8 position after one shift		0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 8 2 3 2 1	(l)
(e)	0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 2 3 2 8 position after fourth shift		0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 8 2 3 2 1	(m)
f	0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 2 3 2 8 Final position		0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 8 2 3 2 1	(n)
(g)	'full' Full Correlation Result 0 0 0 8 2 3 2 1 0 0 0 0		Full Convolution Result 'full' 0 0 0 1 2 3 2 8 0 0 0 0	(o)
(h)	'same' Cropped Correlation Result 0 8 2 3 2 1 0 0		Cropped convolution Result 'same' 0 1 2 3 2 8 0 0	(p)
	Correlation		Convolution	

1-D Correlation and convolution of a filter with discrete unit impulse.

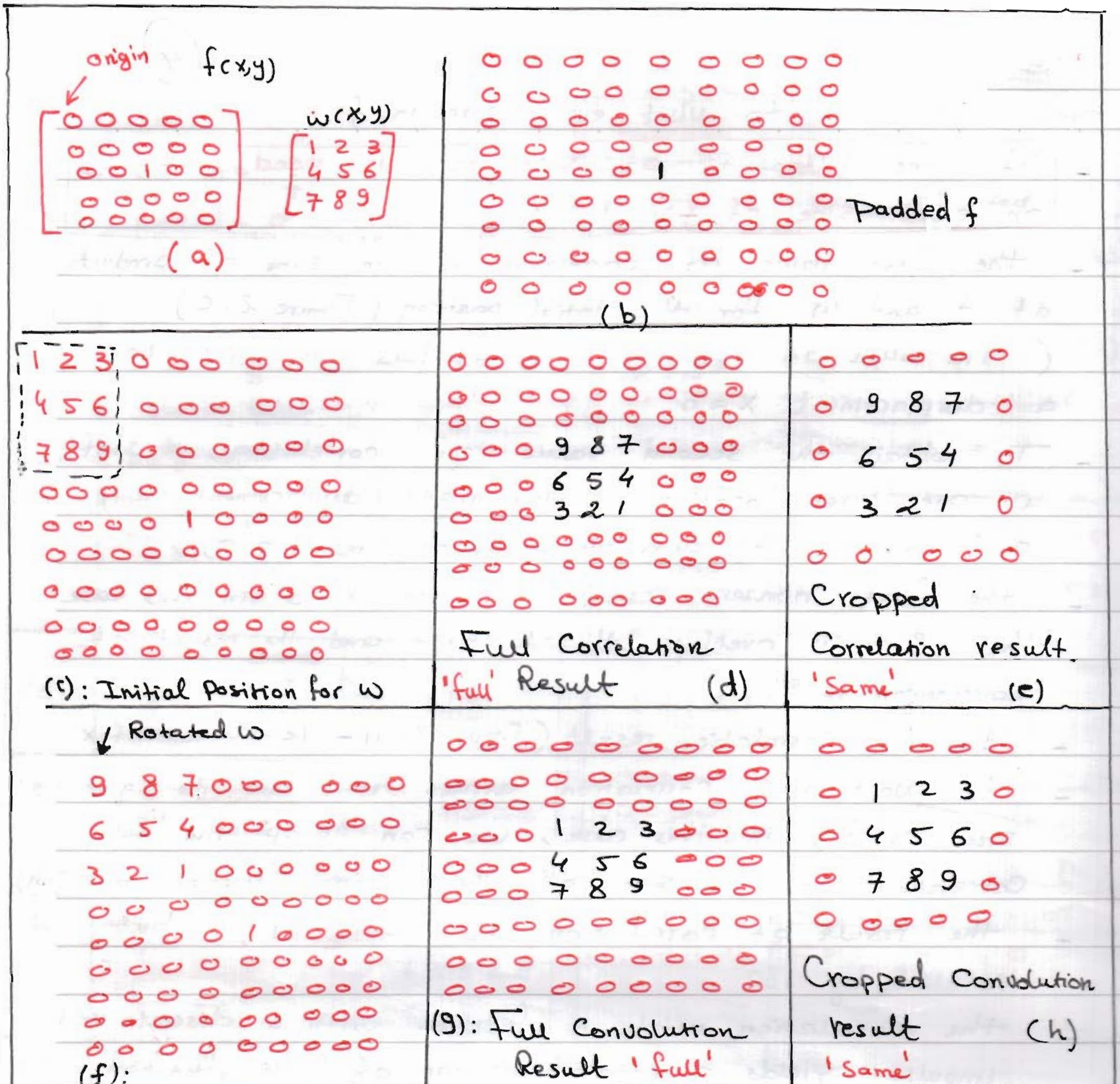
Notes :

- there are parts of the functions (images) that do not overlap (the solution of this problem is pad f with enough 0s on either side to allow each pixel in w to visit every pixel in f.

- if the filter is of size m, we need $(m-1)$ 0s on either side of f.
- the first value of correlation is the sum of products of f and w for the intial position (Figure 2.c) (the sum of product = 0) → this corresponds to a displacement $x = 0$.
- To obtain the second value of correlation, we shift w one pixel location to the right (displacement $x = 1$) and compute the sum of products (result = 0).
- the first nonzero result is when $x = 3$, in this case the 8 in w overlaps the 1 in f and the result of correlation is 8.
- the full correlation result (Figure 2.g) - 12 values of x
- to work with correlation arrays that are the same size as f, in this case, we can crop the full correlation to the size of the original function. (Figure 2. h).
- the result of correlation is a copy of w, but rotated by 180°
- the correlation with a function with a discrete unit impulse yields a rotated version of the function at the location of the impulse.
- the convolution with a function with a discrete unit impulse yields a copy of that function at the location of the impulse.

- Correlation & Convolution with images:

* For a filter of size $m \times n$, we pad the image with a minimum of $m-1$ rows of 0s at the top and the bottom, and $n-1$ columns of 0s on the left and right.



Correlation (middle Row) and Convolution (last Row) of a 2D filter with a 2-D discrete unit impulse

- if the filter mask is symmetric, Correlation and Convolution yield the same result.

Summary:

Correlation of a filter $w(x,y)$ of size $m \times n$ with an image $f(x,y)$ denoted as

$$w(x,y) \star f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

In Similar manner, the Convolution of $w(x,y)$ and $f(x,y)$ denoted by $w(x,y) * f(x,y)$ is given by:

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

where the minus sign on the right flip (rotate by 180°)
(we can flip & shif either f or w)

* Vector Representation of Linear Filtering:

- Correlation:

$$R = w_1 \cdot z_1 + w_2 \cdot z_2 + \cdots + w_{mn} \cdot z_{mn}$$
$$= \sum_{k=1}^{mn} w_k \cdot z_k = w^T \cdot z.$$

R - the response of a mask.
w_k - the coefficients of an $m \times n$ filter
z_k - the corresponding image intensities encompassed by the filter

- Convolution:

we simply rotate the mask by 180°

Example:

the general 3x3 mask equation:

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9$$
$$= \sum_{k=1}^{9} w_k \cdot z_k = w^T \cdot z$$

where:
w and z are 9-dimensional vectors (mask & image)

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Another representation of 3x3 filter mask