Lecture 4

Systems and Convolutions

Administrative

A0 is out.

- It is upgraded
- Meant to help you with python and numpy basics
- Learn how to do homeworks and submit them on gradescope.
- It is due today.

A1 is out

- It is graded
- Due **Tue, Apr 16**

Administrative

Recitation sections on fridays

- (optional)
- Fridays 12:30pm-1:20pm
- JHU 102
- It will be recorded

This week:
We will go over Python & Numpy basics

So far: 2D discrete system (filters)

S is the **system operator**, defined as a mapping or assignment of possible inputs f[n,m] to some possible outputs g[n,m].

$$f[n,m] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow g[n,m]$$

So far: Moving Average

$$h[\cdot,\cdot] = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$g[n,m] = \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1} f[n-k, m-l]$$

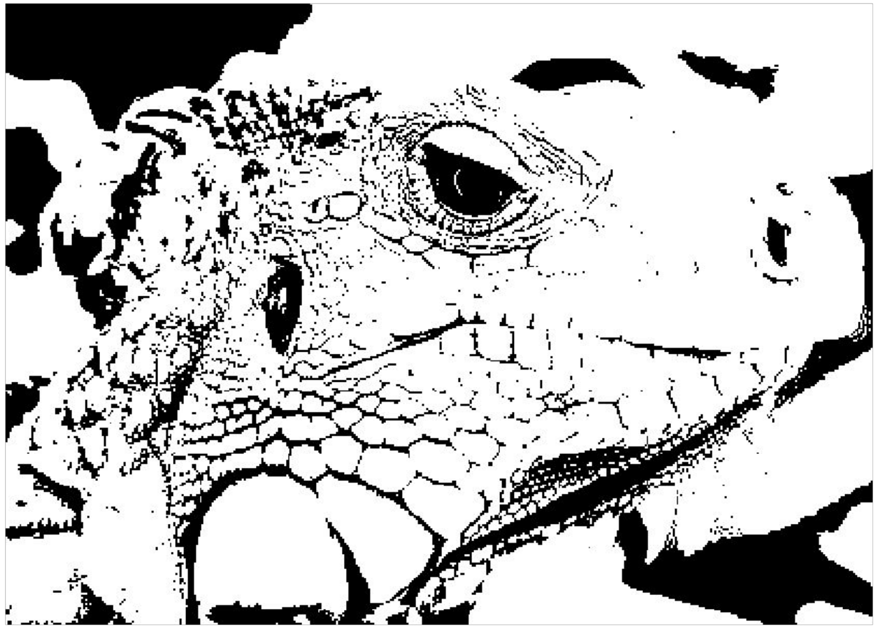
Original image

Smoothed image

So far: Image Segmentation

- Use a simple pixel threshold:

$$g[n,m] = \begin{cases} 255, & f[n,m] > 100 \\ 0, & \text{otherwise.} \end{cases}$$



So far: Properties of systems

- **Amplitude properties:**
  - Additivity

$$\mathcal{S}[f_i[n,m] + f_j[n,m]] = \mathcal{S}[f_i[n,m]] + \mathcal{S}[f_j[n,m]]$$

  - Homogeneity

$$\mathcal{S}[\alpha f[n,m]] = \alpha \mathcal{S}[f[n,m]]$$

  - Superposition

$$\mathcal{S}[\alpha f_i[n,m] + \beta f_j[n,m]] = \alpha \mathcal{S}[f_i[n,m]] + \beta \mathcal{S}[f_j[n,m]]$$

  - Stability

$$\text{If } \forall n,m,\ |f[n,m]| \leq k \implies |\mathcal{S}[f[n,m]]| \leq ck \text{ for some constant c and k}$$

  - Invertibility

$$\mathcal{S}^{-1}\mathcal{S}[f[n,m]] = f[n,m]$$

So far: Properties of systems

- **Spatial properties**
  - Causality

$$\text{for } n < n_0, m < m_0, \text{ if } f[n,m] = 0 \implies g[n,m] = 0$$

  - Shift invariance:

$$f[n - n_0, m - m_0] \xrightarrow{\mathcal{S}} g[n - n_0, m - m_0]$$

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

Linear Systems (filters)

$$f[n,m] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow g[n,m]$$

- Linear filtering:
  - Form a new image whose pixels are a weighted sum of original pixel values
  - Use the same set of weights at each point
- **S** is a linear system (function) iff it *S satisfies*

$$S[\alpha f_i[n,m] + \beta f_j[k,l]] = \alpha S[f_i[n,m]] + \beta S[f_j[k,l]]$$

superposition property

Linear Systems (filters)

$$f[n,m] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow g[n,m]$$

- Q. Is the moving average a linear system?

$$g[n,m] = \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} f[n-k, m-l]$$

Linear Systems (filters)

$$f[n,m] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow g[n,m]$$

- Q. Is the moving average a linear system?

$$g[n,m] = \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} f[n-k, m-l]$$

- Q. Is thresholding a linear system?

$$g[n,m] = \begin{cases} 1, & f[n,m] > 100 \\ 0, & \text{otherwise.} \end{cases}$$

Linear Systems (filters)

$$f[n,m] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow g[n,m]$$

- Q. Is the moving average a linear system?

$$g[n,m] = \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} f[n-k, m-l]$$

- Q. Is thresholding a linear system?
  - Let f1[0,0] = f2[n,m] = 0.4
  - Let T = 0.5
  - So, S[f1[0,0]] = S[f2[0,0]] = 0
  - But S[f1[0,0] + f2[0,0]] = 1

$$g[n,m] = \begin{cases} 1, & f[n,m] > 100 \\ 0, & \text{otherwise.} \end{cases}$$

Linear shift invariant (LSI) systems

- Satisfies two properties:

- Superposition property

$$S[\alpha f_i[n,m] + \beta f_j[k,l]] = \alpha S[f_i[n,m]] + \beta S[f_j[k,l]]$$

- Shift invariance:

$$f[n-n_0, m-m_0] \xrightarrow{\mathcal{S}} g[n-n_0, m-m_0]$$

Moving average system is linear shift invariant (LSI)

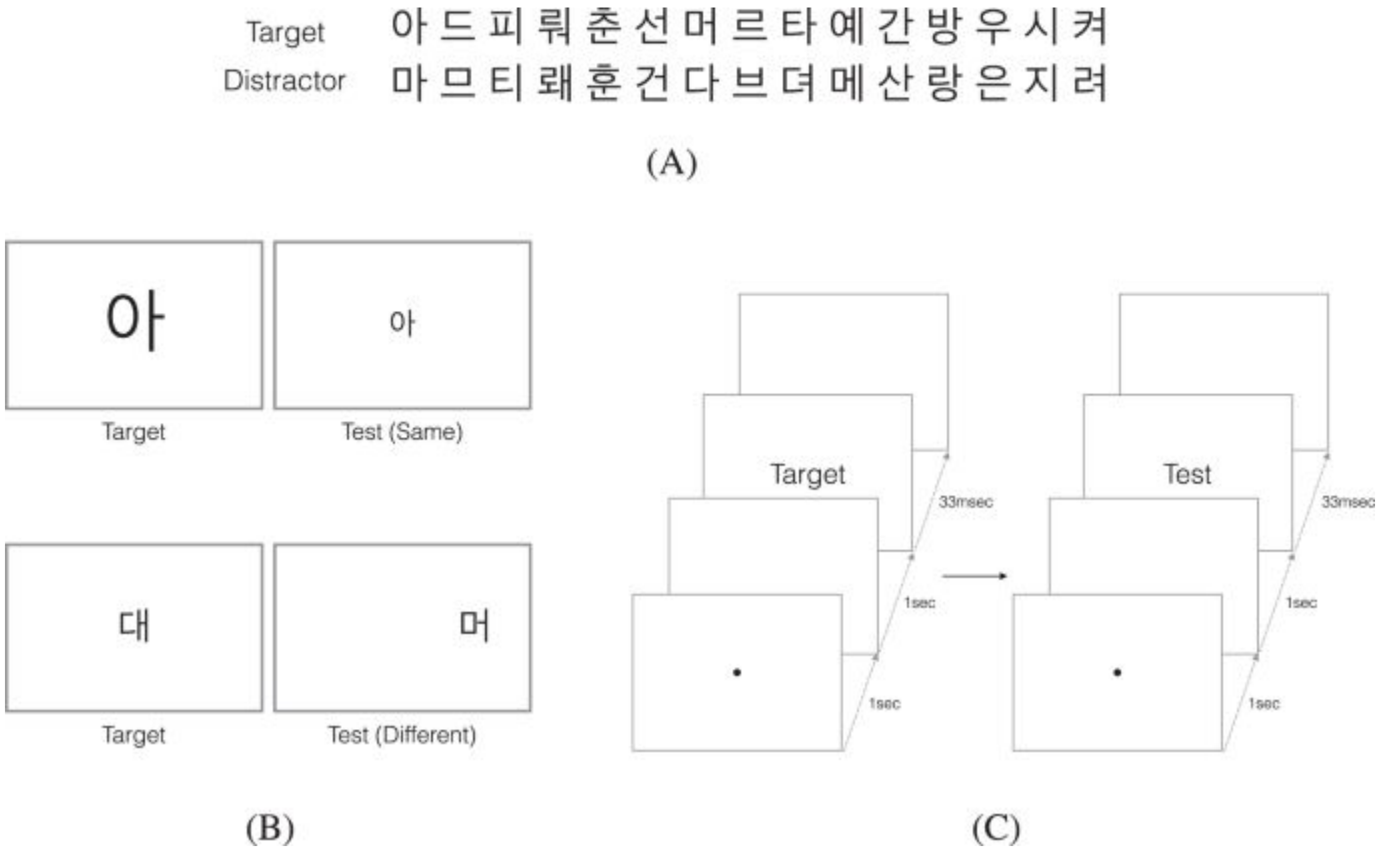
- We are going to use this as an example to dive into interesting properties about linear shift-invariant systems.

- Why are linear shift invariant systems important?

Our visual system is a

linear shift invariant system

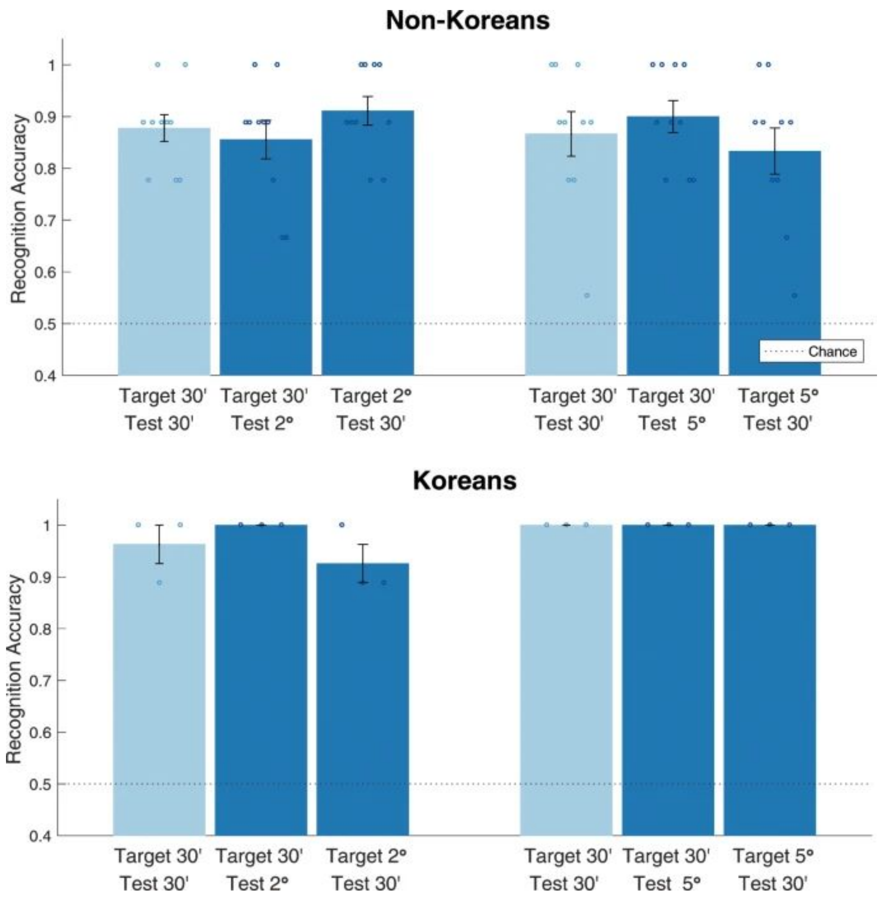
Human vision are scale and translation invariant



Participants were shown some target Korean character once and were tested on whether they can identify the targets from other distractors

Han et al. Scale and translation-invariance for novel objects in human vision. Nature 2020 [link]

Human vision are scale and translation invariant



Very high recognition accuracies

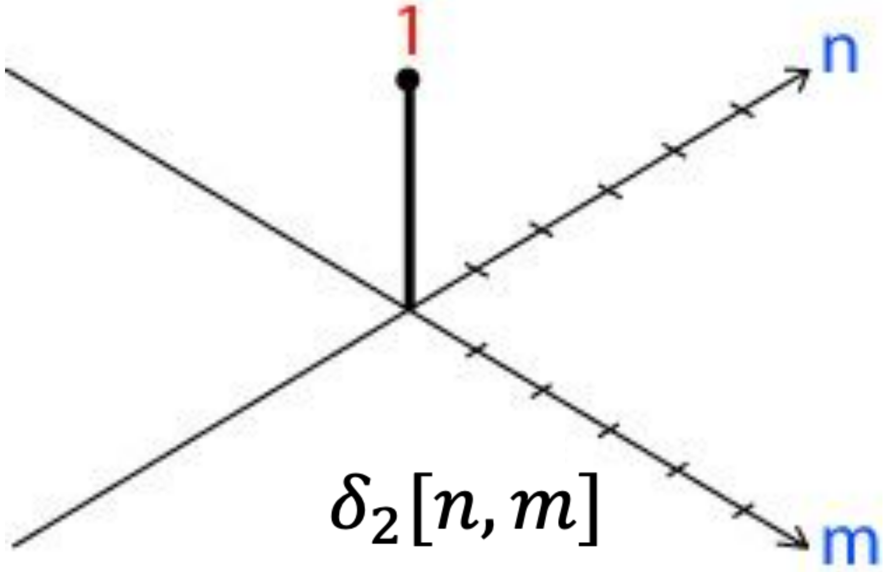
Han et al. Scale and translation-invariance for novel objects in human vision. Nature 2020 [link]

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

2D impulse function

- Let's look at a special function
- 1 at the origin [0,0].
- 0 everywhere else



2D impulse function as an image

- Let's look at a special function
- 1 at the origin [0,0].
- 0 everywhere else

	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	

What happens when we pass an impulse function through a LSI systems

- The moving average filter equation again: $g[n,m] = \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1} f[n-k, m-l]$

$$\delta_2[n,m] \xrightarrow{S} h[n,m]$$

Pass in an impulse function **Record its response**

- By passing an impulse function into an LSI system, we get it's impulse response.
    - We will use h[n, m] to refer to the impulse response

What happens when we pass an impulse function through a LSI systems

Before we do this, let's remember how we used the moving average filter last lecture

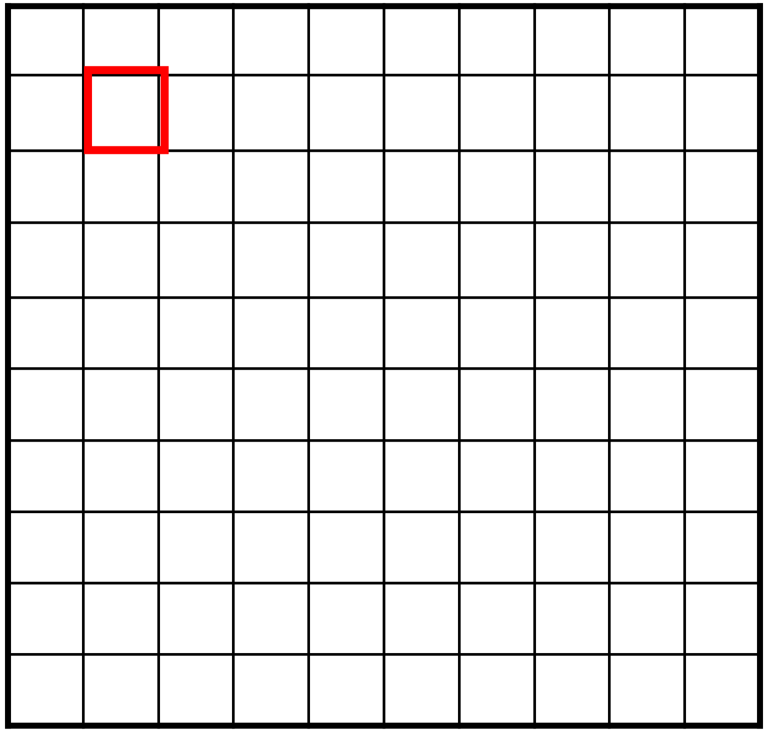
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	1	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	

Remember the Moving Average filter from last lecture

$f[n, m]$

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

$g[n, m]$



Courtesy of S. Seitz

Remember the Moving Average filter from last lecture

$f[n,m]$

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

$g[n,m]$

	0	10							

Remember the Moving Average filter from last lecture

$f[n,m]$

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

$g[n,m]$

	0	10	20						

Remember the Moving Average filter from last lecture

$f[n, m]$

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

$g[n, m]$

	0	10	20	30					

Remember the Moving Average filter from last lecture

$f[n,m]$

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

$g[n,m]$

	0	10	20	30	30				

Remember the Moving Average filter from last lecture

$f[n, m]$

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

$g[n, m]$

	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	

Now let's do the same thing with an impulse function

$\mathrm{f}[n,m]$

	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	

$h[n,m]$

Now let’s do the same thing with an impulse function

$\mathrm{f}[n, m]$

	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	

$h[n, m]$

		?						

Now let's do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

		0						

Now let's do the same thing with an impulse function

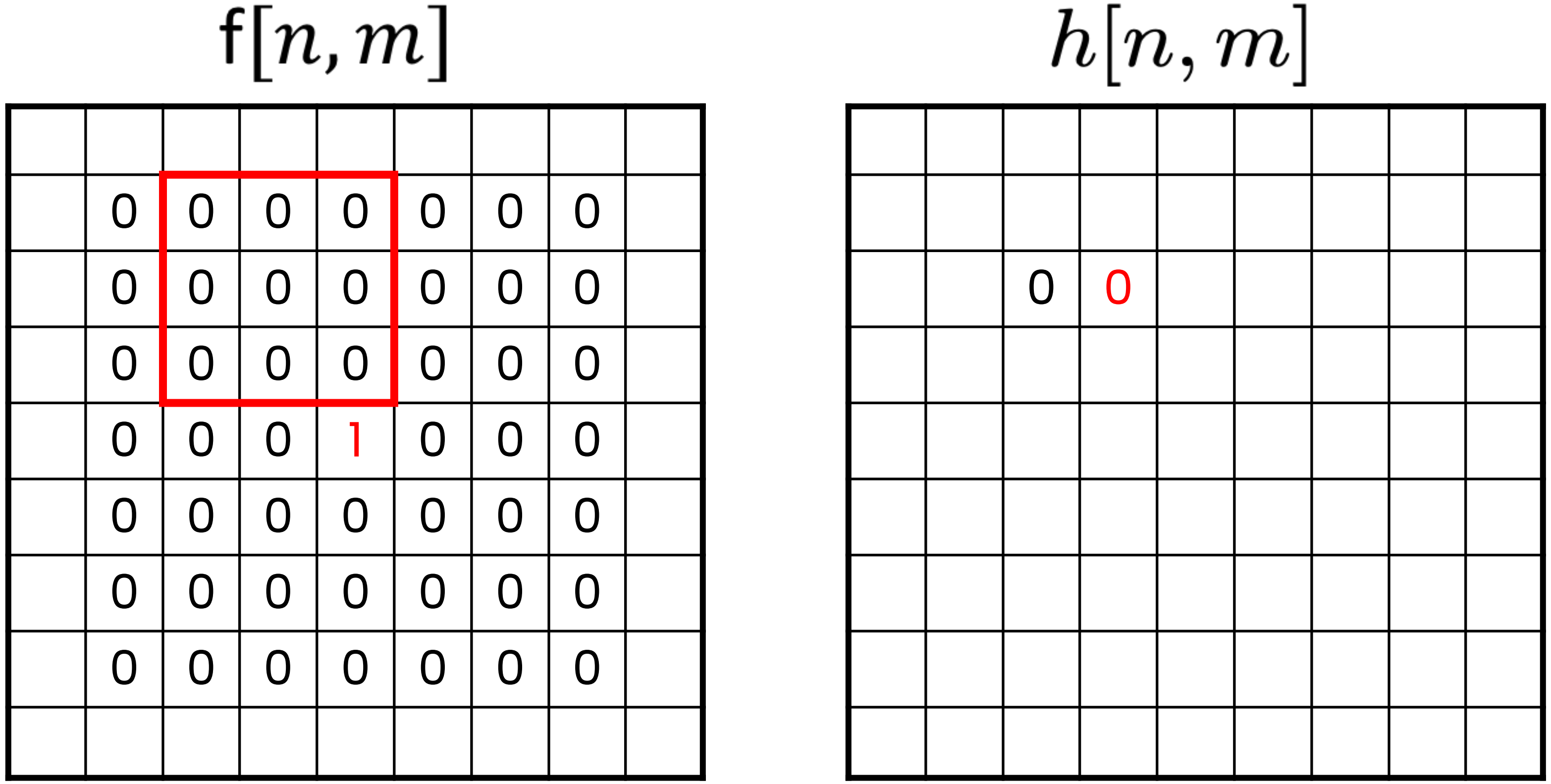
$f[n, m]$

	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	

$h[n, m]$

		0	?					

Now let’s do the same thing with an impulse function



Now let's do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	?						

Now let's do the same thing with an impulse function

$\mathsf{f}[n, m]$

	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	

$h[n, m]$

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

Now let's do the same thing with an impulse function

$f[n, m]$

	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	

$h[n, m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	?					

Now let's do the same thing with an impulse function

$\mathrm{f}[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9					

Now let's do the same thing with an impulse function

$\mathsf{f}[n, m]$

	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	

$h[n, m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	?				

Now let's do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	1/9				

Now let’s do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	?					

Now let's do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	1/9					

Now let's do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	1/9	?				

Now let's do the same thing with an impulse function

$f[n,m]$

	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	

$h[n,m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	1/9	1/9				

Now let's do the same thing with an impulse function

$\mathrm{f}[n, m]$

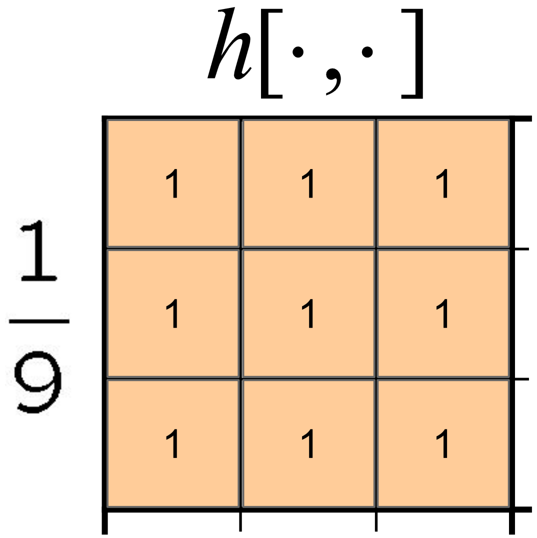
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	1	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	

$h[n, m]$

	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	1/9	1/9	1/9	0	0	
	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	

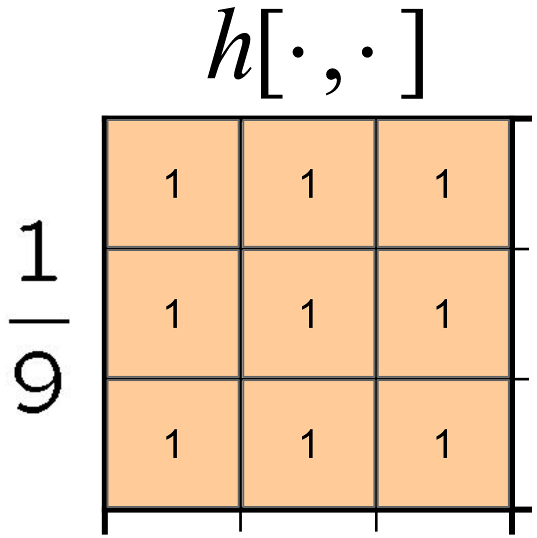
Impulse response of the 3 by 3 moving average filter

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & \underline{1/9} & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$



Notice that any filter can be written as a summation of shifted delta functions

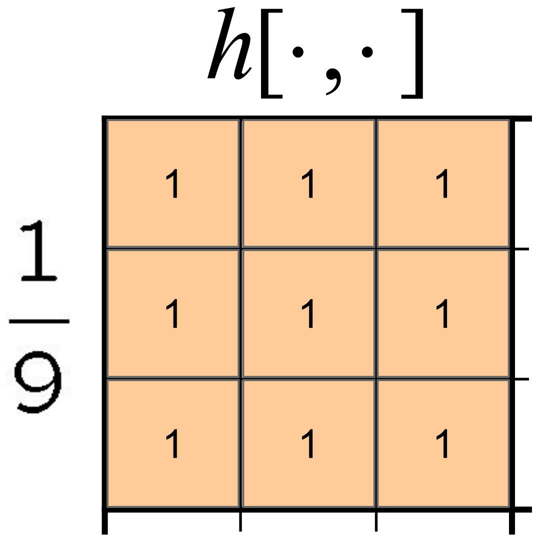
$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & \boxed{\underline{1/9}} & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$



$$h[0,0] = \frac{1}{9}\delta_2[0,0]$$

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

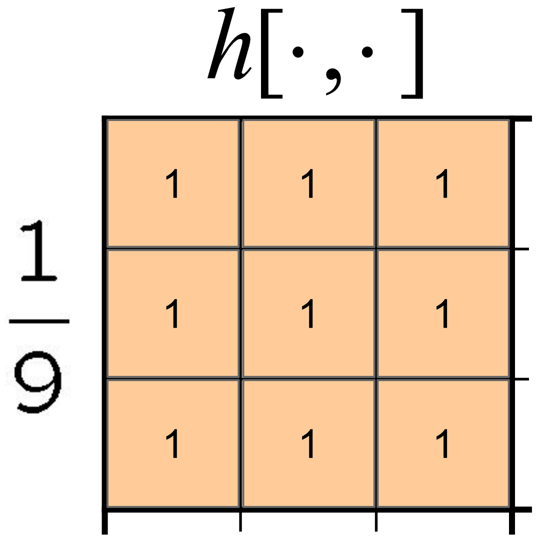


$$h[0,0] = \frac{1}{9}\delta_2[0,0]$$

$$h[0,1] = \frac{1}{9}\delta_2[0,0]$$

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & \boxed{1/9} \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$



$$h[0,0] = \frac{1}{9}\delta_2[0,0]$$

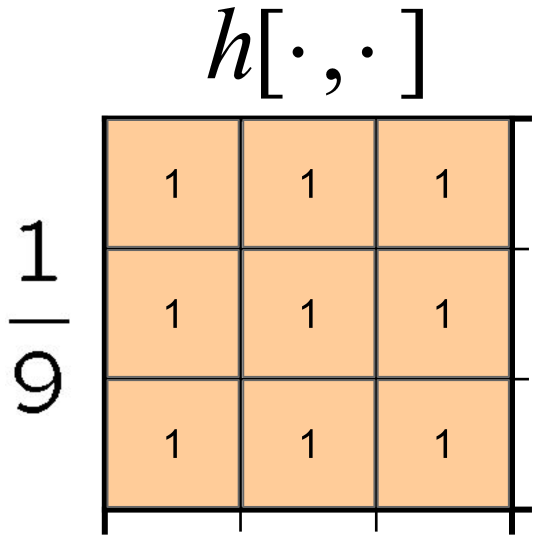
$$h[0,1] = \frac{1}{9}\delta_2[0,0]$$

Q. For what values of **n** and **m** is h[,] **not** zero?

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[n-k, m-l]$$

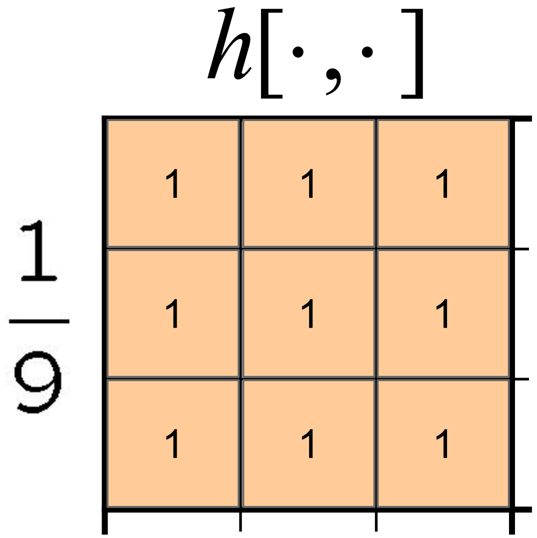


The general form for a moving average h[n,m]

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[n-k, m-l]$$

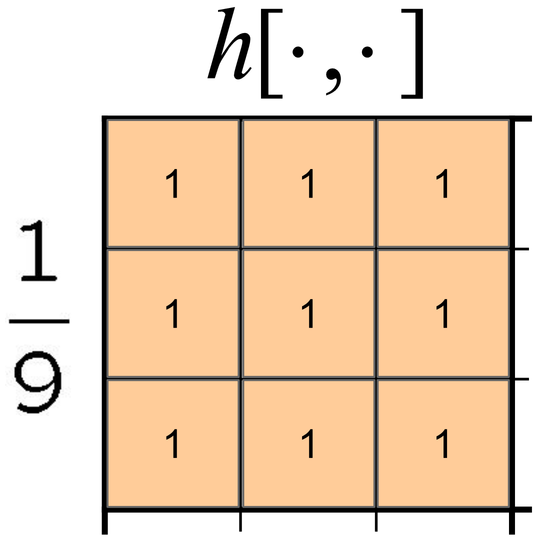


Q. Why is this the general form?

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[n-k, m-l]$$



Q. Why is this the general form?

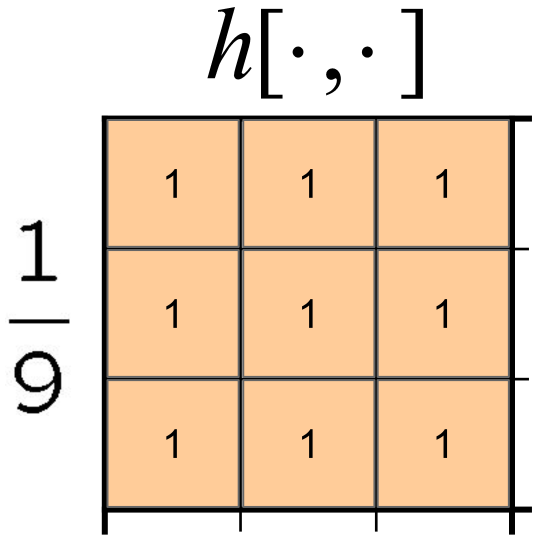
As long as n-1, n, or n+1 is 0, the value is 1/9

Same for m

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[n-k, m-l]$$



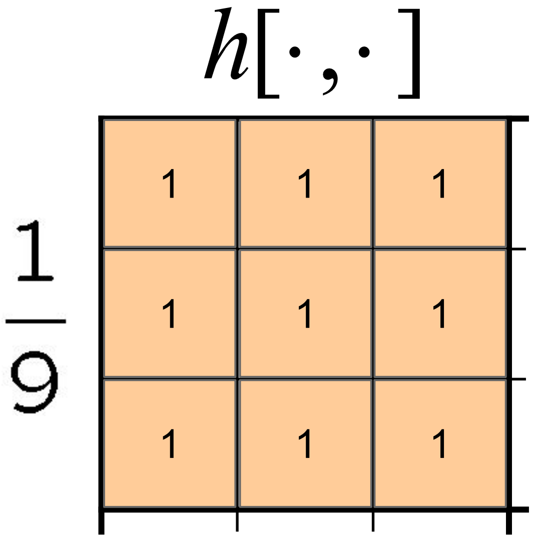
Q. What if we swap n-k for k-n. Does that also work?

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[k-n, l-m]$$

Notice that any filter can be written as a summation of shifted delta functions

$$h[n,m] = \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & \underline{1/9} & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k, m-l]$$



Q. What if we swap n-k for k-n. Does that also work?

$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[k-n, l-m]$$

Yes because h is symmetric across the origin

Q. What if h was the filter on the right:

h[:, -1] = 0

$h[n, m]$

(A) $= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[n-k, m-l]$

(B) $= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} \delta_2[k-n, l-m]$

$h[\cdot, \cdot]$

$$\frac{1}{9} \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Is A correct?
Is B correct?
Are both correct?
Are both wrong?

Q. What if h was the filter on the right:

h[:, -1] = 0

$$h[n,m] = \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=0}^{1} \delta_2[n-k, m-l]$$

$$h[\cdot,\cdot] = \frac{1}{9} \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Q. What if h was the filter on the right:

h[:, -1] = 0

$$h[n,m] = \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=0}^{1} \delta_2[n-k, m-l]$$

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{0} \delta_2[k-n, l-m]$$

Because h is not symmetric, we need to invert the range if we invert m-l to l-m

$h[\cdot,\cdot]$

$\frac{1}{9}$

0	1	1
0	1	1
0	1	1

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

Property of (LSI) systems

- An LSI system is completely specified by its impulse response.
  - For any input f, we can compute g using only the impulse response h.

$$f[n,m] \xrightarrow{S} g[n,m]$$

Property of (LSI) systems

- An LSI system is completely specified by its impulse response.
  - For any input f, we can compute g using only the impulse response h.

$$f[n, m] \xrightarrow{S} g[n, m]$$

  - **Let's derive an expression for g in terms of h.**

Recall the 3 properties about LSI systems:

1. We know what happens when we send a delta function through an LSI system:

$$\delta_2[n,m] \longrightarrow \boxed{\text{System } \mathcal{S}} \longrightarrow h[n,m]$$

Recall the 3 properties about LSI systems:

1. We know what happens when we send a delta function through an LSI system:

$$\delta_2[n,m] \longrightarrow \boxed{\text{System } \mathcal{S}} \longrightarrow h[n,m]$$

2. We also know that LSI systems shift the output if the input is shifted:

$$\delta_2[n-k,m-l] \longrightarrow \boxed{\text{System } \mathcal{S}} \longrightarrow h[n-k,m-l]$$

Recall the 3 properties about LSI systems:

1. We know what happens when we send a delta function through an LSI system:

$$\delta_2[n,m] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow h[n,m]$$

2. We also know that LSI systems shift the output if the input is shifted:

$$\delta_2[n-k,m-l] \rightarrow \boxed{\text{System } \mathcal{S}} \rightarrow h[n-k,m-l]$$

3. Finally, the superposition principle:

$$S\{\alpha f_1[n,m] + \beta f_2[n,m]\} = \alpha S\{f_1[n,m]\} + \beta S\{f_2[n,m]\}$$

Key idea: write down f as a sum of impulses

Let's say our input f is a 3x3 image:

f[0,0]	f[0,1]	f[1,1]
f[1,0]	f[1,1]	f[1,2]
f[2,0]	f[2,1]	f[2,2]

=

f[0,0]	0	0
0	0	0
0	0	0

+

0	f[0,1]	0
0	0	0
0	0	0

+ . . . +

0	0	0
0	0	0
0	0	f[2,2]

Key idea: write down f as a sum of impulses

Let’s say our input f is a 3x3 image:

$$\begin{bmatrix} f[0,0] & f[0,1] & f[1,1] \\ f[1,0] & f[1,1] & f[1,2] \\ f[2,0] & f[2,1] & f[2,2] \end{bmatrix} = \begin{bmatrix} f[0,0] & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & f[0,1] & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \ldots + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & f[2,2] \end{bmatrix}$$

$$= f[0,0] \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + f[0,1] \times \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \ldots + f[2,2] \times \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Key idea: write down f as a sum of impulses

Let's say our input f is a 3x3 image:

f[0,0]	f[0,1]	f[1,1]
f[1,0]	f[1,1]	f[1,2]
f[2,0]	f[2,1]	f[2,2]

=

f[0,0]	0	0
0	0	0
0	0	0

+

0	f[0,1]	0
0	0	0
0	0	0

+ … +

0	0	0
0	0	0
0	0	f[2,2]

= f[0,0] ×

1	0	0
0	0	0
0	0	0

\+ f[0,1] ×

0	1	0
0	0	0
0	0	0

\+ … + f[2,2] ×

0	0	0
0	0	0
0	0	1

$$= f[0,0] \cdot \delta_2[n,m] + f[0,1] \cdot \delta_2[n,m-1] + \ldots + f[2,2] \cdot \delta_2[n-2,m-2]$$

Key idea: write down f as a sum of impulses

- More generally:

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

Key idea: write down f as a sum of impulses

- More generally:

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

Key idea: write down f as a sum of impulses

- More generally:

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \boxed{f[k,l]} \cdot \delta_2[n-k, m-l]$$

For given k, l,
this is a constant

Key idea: write down f as a sum of impulses

- More generally:

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \boxed{f[k,l]} \cdot \boxed{\delta_2[n-k, m-l]}$$

For given k, l,
this is a constant

This is a function
of n, m

Key idea: write down f as a sum of impulses

- Superposition

$$S\{\alpha f_1[n,m] + \beta f_2[n,m]\} = \alpha S\{f_1[n,m]\} + \beta S\{f_2[n,m]\}$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

For given k, l, this is a constant

This is a function of n, m

Key idea: write down f as a sum of impulses

- Superposition

$$S\{\alpha f_1[n,m] + \beta f_2[n,m]\} = \alpha S\{f_1[n,m]\} + \beta S\{f_2[n,m]\}$$

$$\mathcal{S}[\sum_i \alpha_i f_i[n,m]] = \sum_i \alpha_i \mathcal{S}[f_i[n,m]]$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

For given k, l, this is a constant

This is a function of n, m

Key idea: write down f as a sum of impulses

- Superposition:

$$S\{\alpha f_1[n,m] + \beta f_2[n,m]\} = \alpha S\{f_1[n,m]\} + \beta S\{f_2[n,m]\}$$

$$\mathcal{S}[\sum_i \alpha_i f_i[n,m]] = \sum_i \alpha_i \mathcal{S}[f_i[n,m]]$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

$$\xrightarrow{S} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot S\{\delta_2[n-k, m-l]\}$$

Key idea: write down f as a sum of impulses

- Superposition:

$$S\{\alpha f_1[n,m] + \beta f_2[n,m]\} = \alpha S\{f_1[n,m]\} + \beta S\{f_2[n,m]\}$$

$$\mathcal{S}[\sum_i \alpha_i f_i[n,m]] = \sum_i \alpha_i \mathcal{S}[f_i[n,m]]$$

- We can now use superposition to see what the output g is:

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

$$\xrightarrow{S} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \boxed{S\{\delta_2[n-k, m-l]\}}$$

Key idea: write down f as a sum of impulses

- From previous slide:

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

$$\xrightarrow{S} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot S\{\delta_2[n-k, m-l]\}$$

- Using shift invariance, we get a shifted impulse response:

$$S\{\delta_2[n-k, m-l]\} = h[n-k, m-l]$$

We can write g as a function of h

- We have:

$$f[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot \delta_2[n-k, m-l]$$

$$\xrightarrow{S} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot S\{\delta_2[n-k, m-l]\}$$

- Which means:

$$f[n,m] \xrightarrow{S} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$

Linear Shift Invariant (LSI) systems

- An LSI system is completely specified by its impulse response.
  - For any input f, we can compute the output g in terms of the impulse response h.

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$f[n,m] \xrightarrow{S} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$

Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$

Linear Shift Invariant (LSI) systems

- An LSI system is completely specified by its impulse response.

$$f[n,m] \xrightarrow{S} g[n,m]$$

$$g[n,m] = f[n,m] * h[n,m]$$

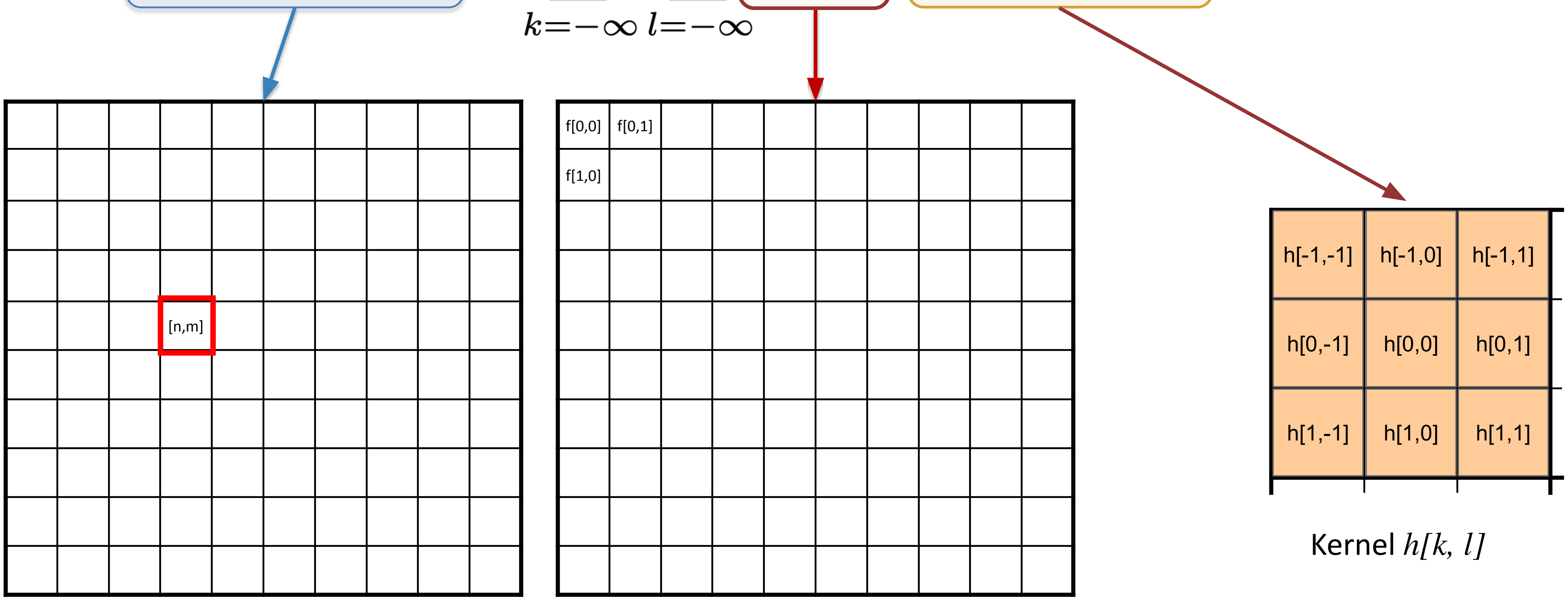
$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

2D Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$



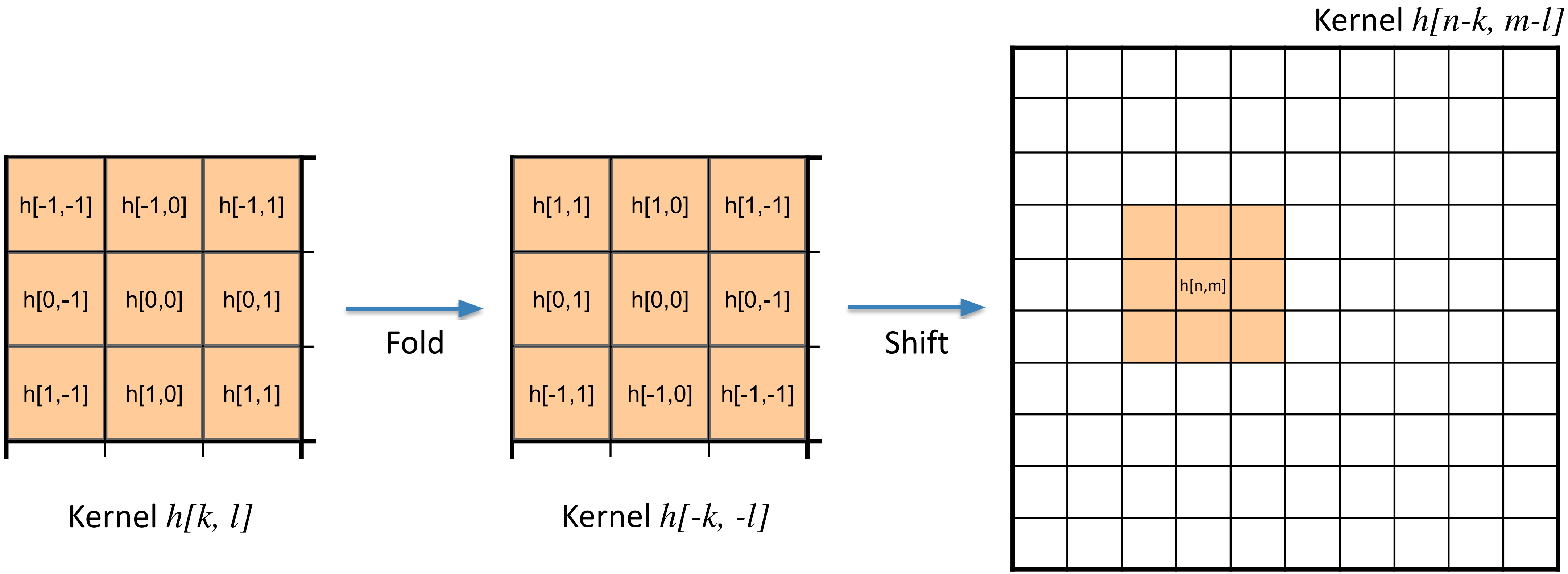
Kernel $h[k, l]$

Output $f * h$

Image $f[k, l]$

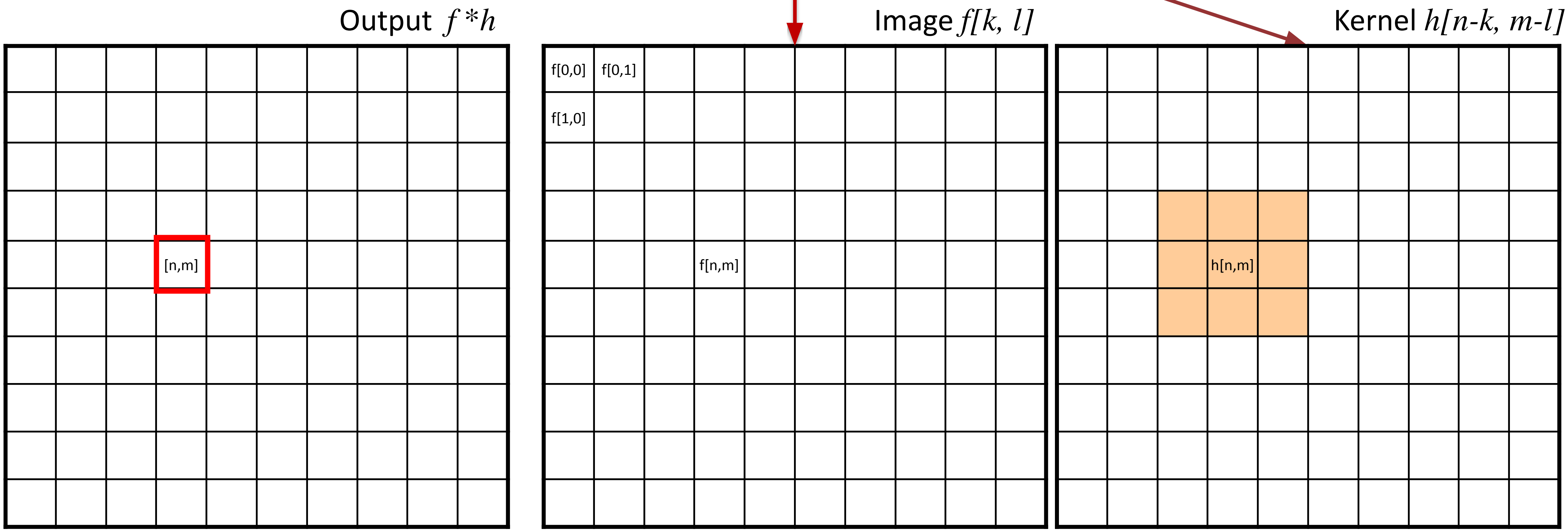
2D Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$



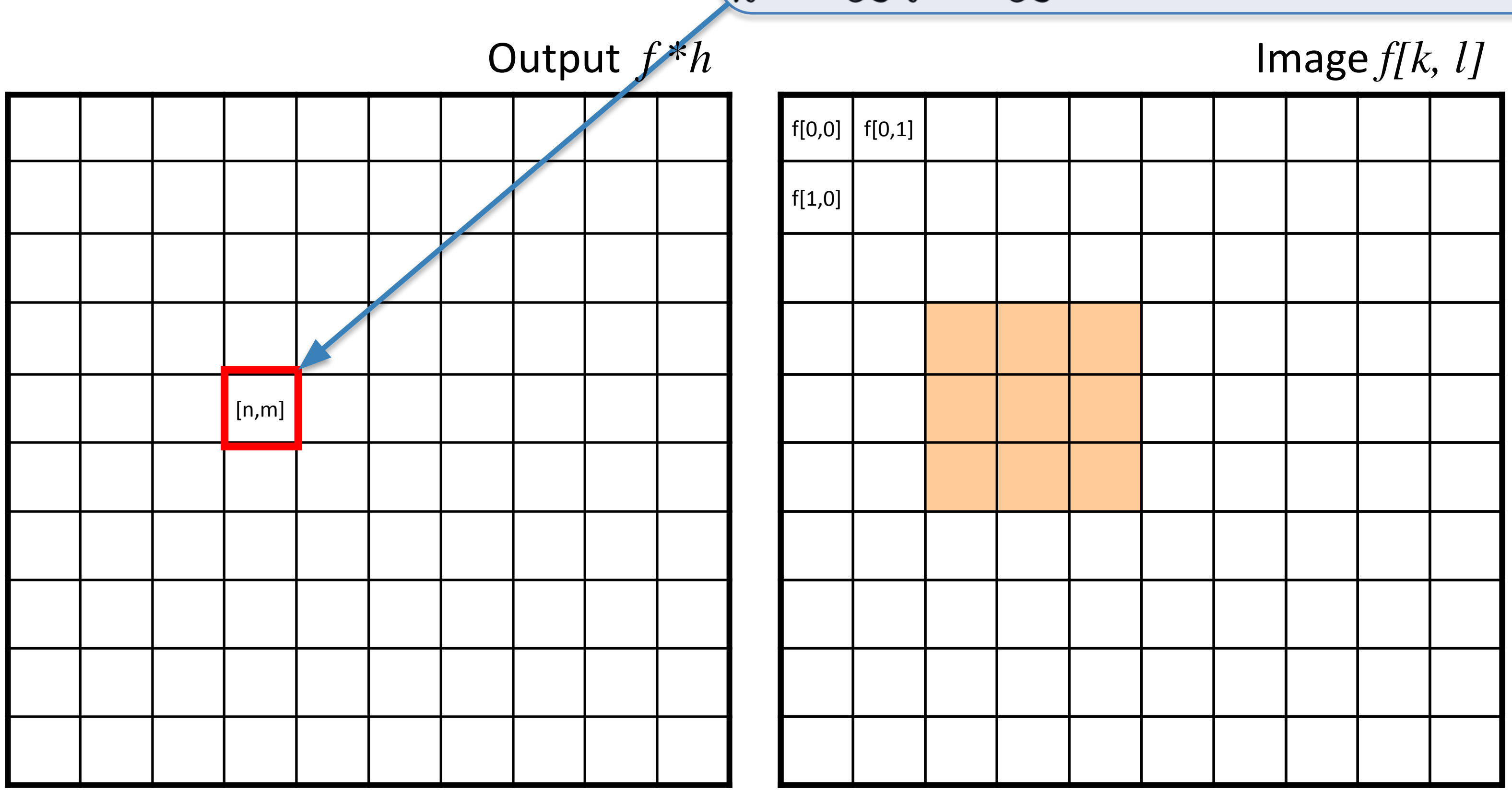
2D Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$



2D Discrete Convolution

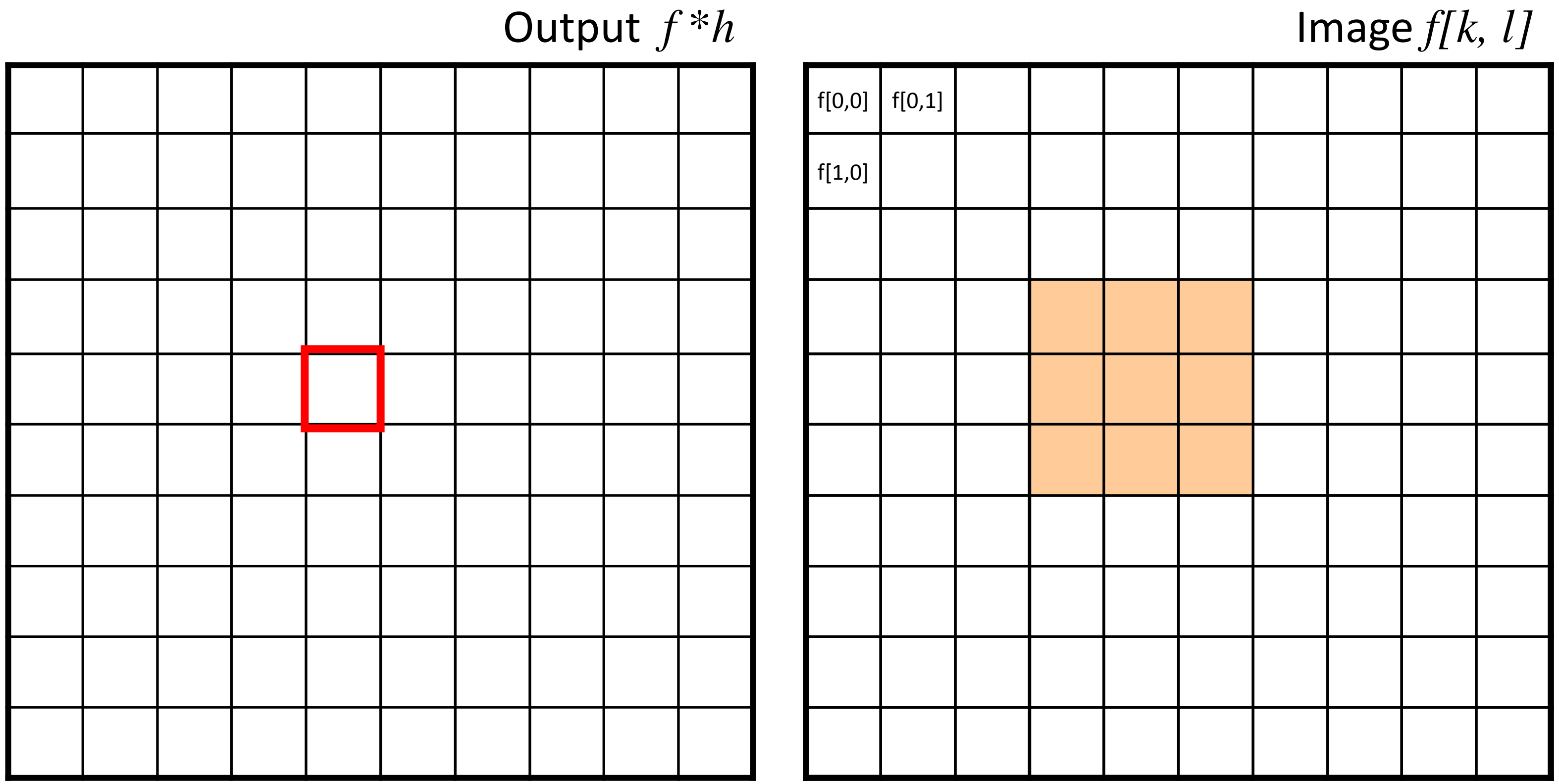
$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$



Element-wise multiplication
Image $f[k, l]$ • Kernel $h[n-k, m-l]$

2D Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$



Element-wise multiplication
Image $f[k, l]$ • Kernel $h[n\text{-}k, m\text{-}l]$

2D Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$

Output $f * h$

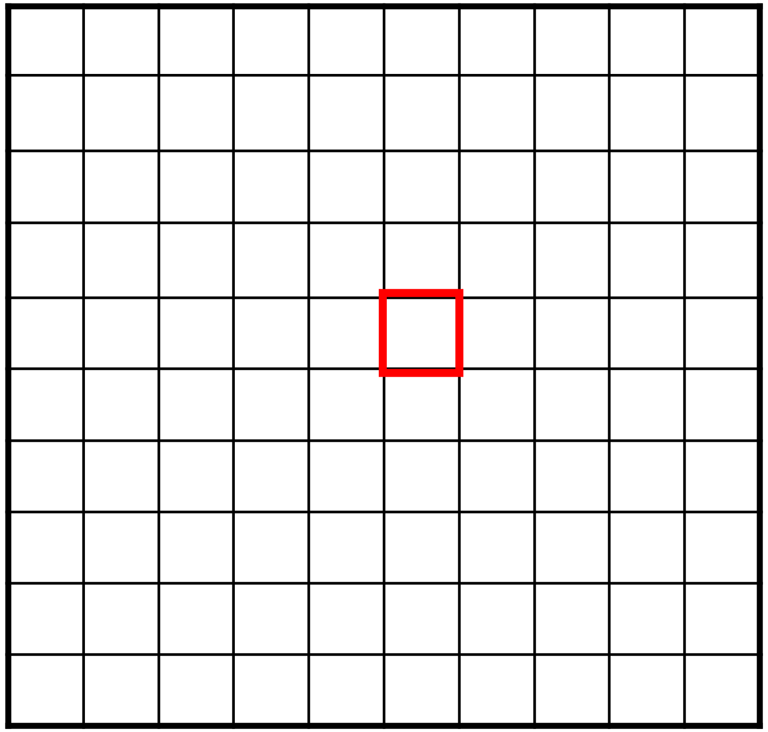
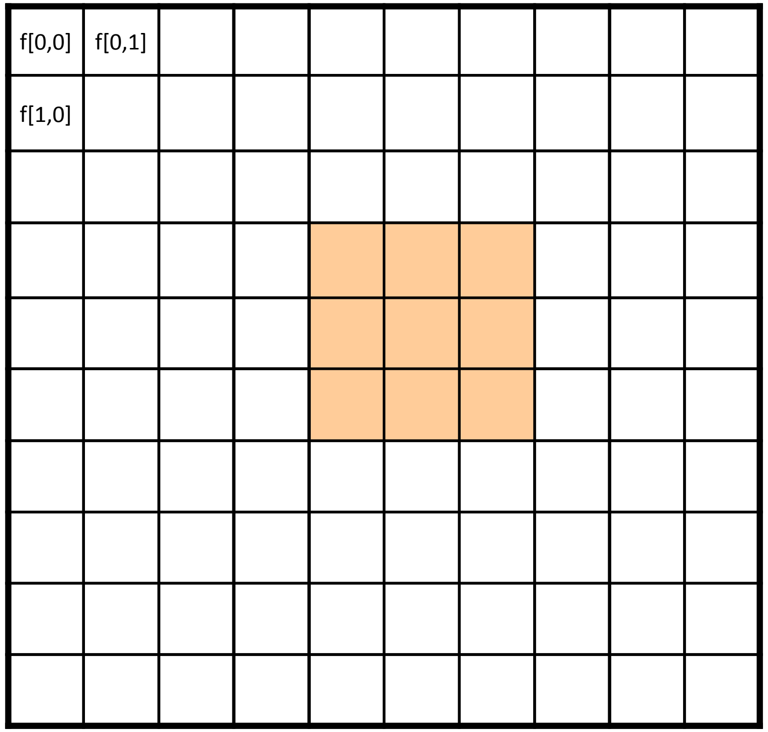


Image $f[k, l]$

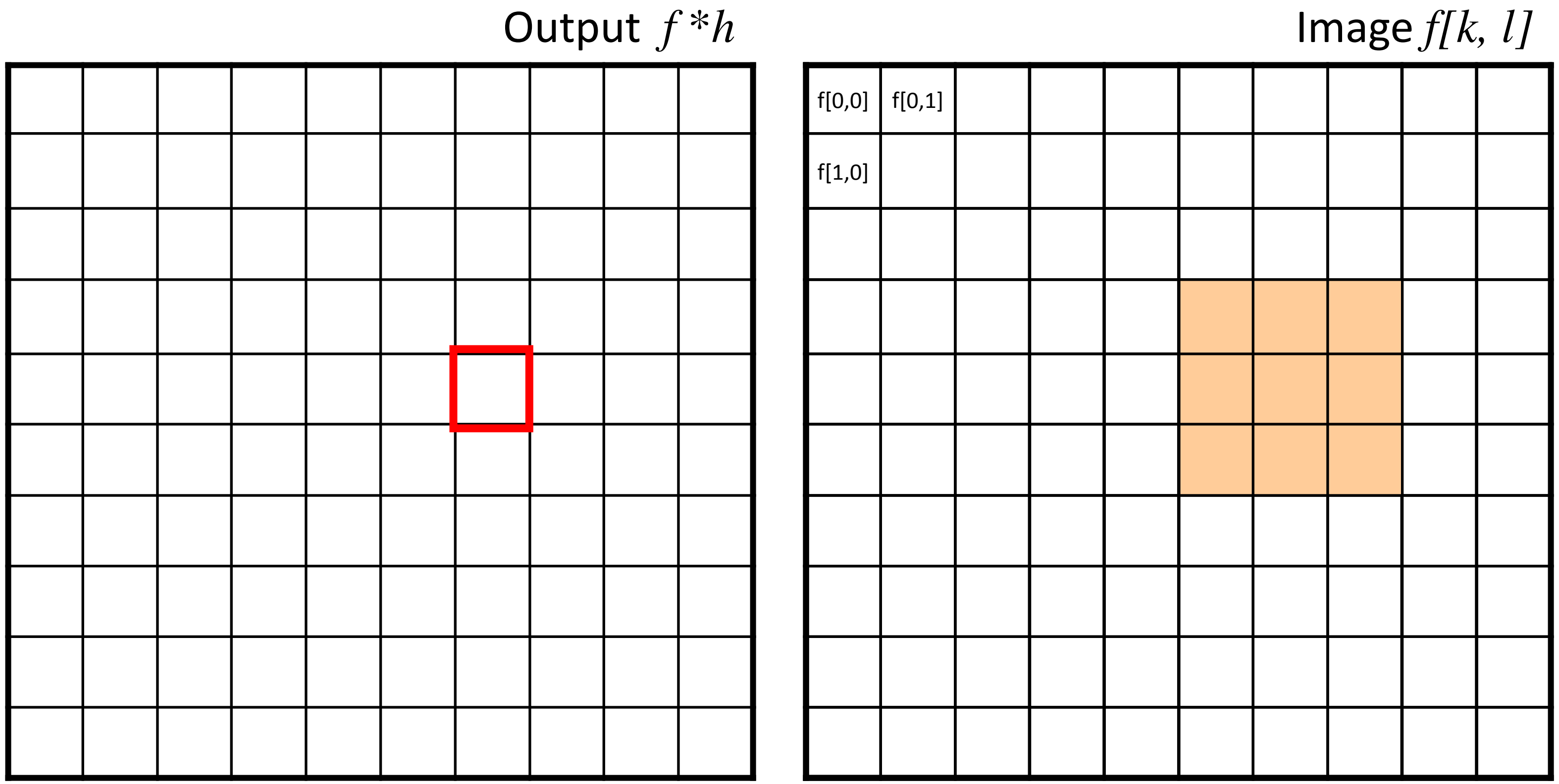


Element-wise multiplication

Image $f[k, l]$ • Kernel $h[n\text{-}k, m\text{-}l]$

2D Discrete Convolution

$$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$



Element-wise multiplication
Image $f[k, l]$ • Kernel $h[n\text{-}k, m\text{-}l]$

2D Discrete Convolution

- $$f[n,m] * h[n,m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k,l] \cdot h[n-k, m-l]$$

Algorithm:

- Fold $h[k,l]$ about origin to form $h[-k,-l]$
- Shift the folded results by n, m to form $h[n-k, m-l]$
- Multiply $h[n-k, m-l]$ by $f[k, l]$
- Sum over all k, l, store result in output position $[n, m]$
- Repeat for every n, m

2D convolution example

1	2	3
4	5	6
7	8	9

Input

n \ m	-1	0	1
-1	-1	-2	-1
0	0	0	0
1	1	2	1

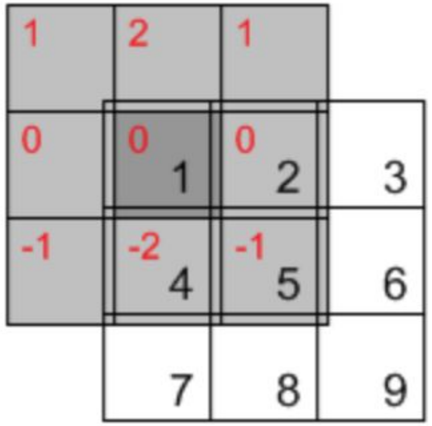
Kernel

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

2D convolution example



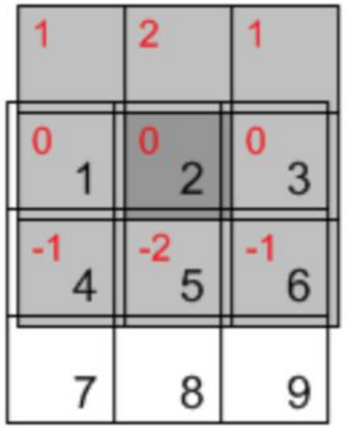
$$= x[-1,-1]\cdot h[1,1] + x[0,-1]\cdot h[0,1] + x[1,-1]\cdot h[-1,1]$$
$$+ x[-1,0]\cdot h[1,0] + x[0,0]\cdot h[0,0] + x[1,0]\cdot h[-1,0]$$
$$+ x[-1,1]\cdot h[1,-1] + x[0,1]\cdot h[0,-1] + x[1,1]\cdot h[-1,-1]$$
$$= 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 0\cdot 0 + 1\cdot 0 + 2\cdot 0 + 0\cdot(-1) + 4\cdot(-2) + 5\cdot(-1) = -13$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

2D convolution example



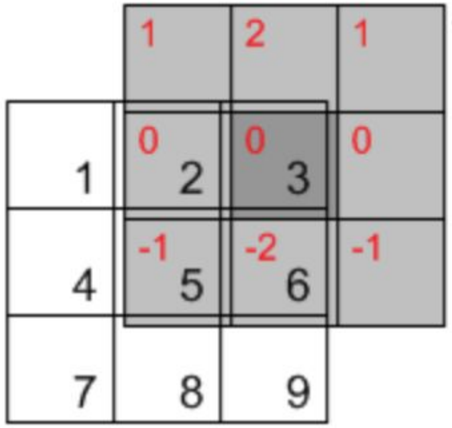
$$
\begin{aligned}
&= x[0,-1]\cdot h[1,1] + x[1,-1]\cdot h[0,1] + x[2,-1]\cdot h[-1,1] \\
&\quad + x[0,0]\cdot h[1,0] + x[1,0]\cdot h[0,0] + x[2,0]\cdot h[-1,0] \\
&\quad + x[0,1]\cdot h[1,-1] + x[1,1]\cdot h[0,-1] + x[2,1]\cdot h[-1,-1] \\
&= 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 1\cdot 0 + 2\cdot 0 + 3\cdot 0 + 4\cdot(-1) + 5\cdot(-2) + 6\cdot(-1) = -20
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

2D convolution example



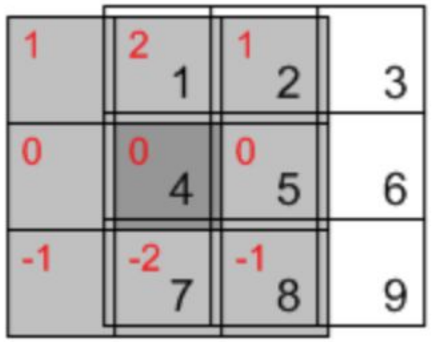
$$
\begin{aligned}
&= x[1,-1]\cdot h[1,1] + x[2,-1]\cdot h[0,1] + x[3,-1]\cdot h[-1,1] \\
&\quad + x[1,0]\cdot h[1,0] + x[2,0]\cdot h[0,0] + x[3,0]\cdot h[-1,0] \\
&\quad + x[1,1]\cdot h[1,-1] + x[2,1]\cdot h[0,-1] + x[3,1]\cdot h[-1,-1] \\
&= 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 2\cdot 0 + 3\cdot 0 + 0\cdot 0 + 5\cdot(-1) + 6\cdot(-2) + 0\cdot(-1) = -17
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

2D convolution example



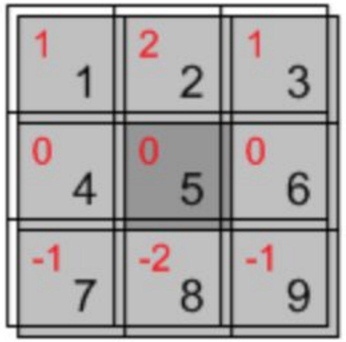
$$
\begin{aligned}
&= x[-1,0]\cdot h[1,1] + x[0,0]\cdot h[0,1] + x[1,0]\cdot h[-1,1] \\
&\quad + x[-1,1]\cdot h[1,0] + x[0,1]\cdot h[0,0] + x[1,1]\cdot h[-1,0] \\
&\quad + x[-1,2]\cdot h[1,-1] + x[0,2]\cdot h[0,-1] + x[1,2]\cdot h[-1,-1] \\
&= 0\cdot 1 + 1\cdot 2 + 2\cdot 1 + 0\cdot 0 + 4\cdot 0 + 5\cdot 0 + 0\cdot(-1) + 7\cdot(-2) + 8\cdot(-1) = -18
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

2D convolution example



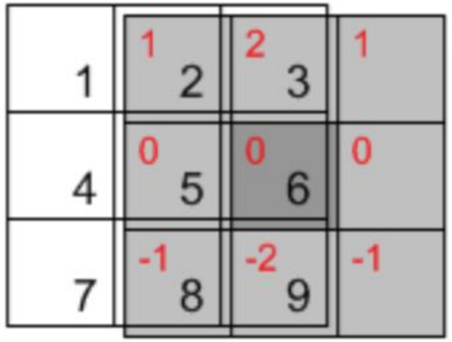
$$
\begin{aligned}
&= x[0,0]\cdot h[1,1] + x[1,0]\cdot h[0,1] + x[2,0]\cdot h[-1,1] \\
&\quad + x[0,1]\cdot h[1,0] + x[1,1]\cdot h[0,0] + x[2,1]\cdot h[-1,0] \\
&\quad + x[0,2]\cdot h[1,-1] + x[1,2]\cdot h[0,-1] + x[2,2]\cdot h[-1,-1] \\
&= 1\cdot 1 + 2\cdot 2 + 3\cdot 1 + 4\cdot 0 + 5\cdot 0 + 6\cdot 0 + 7\cdot(-1) + 8\cdot(-2) + 9\cdot(-1) = -24
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

2D convolution example



$$
\begin{aligned}
&= x[1,0]\cdot h[1,1] + x[2,0]\cdot h[0,1] + x[3,0]\cdot h[-1,1] \\
&\quad + x[1,1]\cdot h[1,0] + x[2,1]\cdot h[0,0] + x[3,1]\cdot h[-1,0] \\
&\quad + x[1,2]\cdot h[1,-1] + x[2,2]\cdot h[0,-1] + x[3,2]\cdot h[-1,-1] \\
&= 2\cdot 1 + 3\cdot 2 + 0\cdot 1 + 5\cdot 0 + 6\cdot 0 + 0\cdot 0 + 8\cdot(-1) + 9\cdot(-2) + 0\cdot(-1) = -18
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Slide credit: Song Ho Ahn

Practice with convolution

Original

$*$

0	0	0
0	1	0
0	0	0

= ?

Practice with convolution

Original

*

0	0	0
0	1	0
0	0	0

=

Filtered
(no change)

Practice with convolution

Original

$$*$$

0	0	0
0	0	1
0	0	0

= ?

Practice with convolution

Original

*

0	0	0
0	0	1
0	0	0

=

Shifted right
By 1 pixel

Practice with convolution

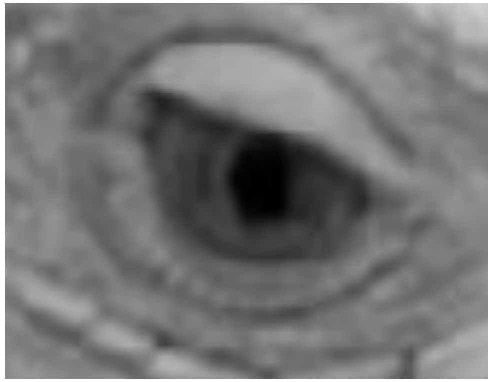
Original

$$* \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \ ?$$

Practice with convolution

Original

$$* \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} =$$



Blurry output

What happens if a system contains multiple filters?

Original

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = ?$$

(Note that filter sums to 1)

What happens if a system contains multiple filters?

Original

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

What does blurring take away?

–

=

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

What does blurring take away?

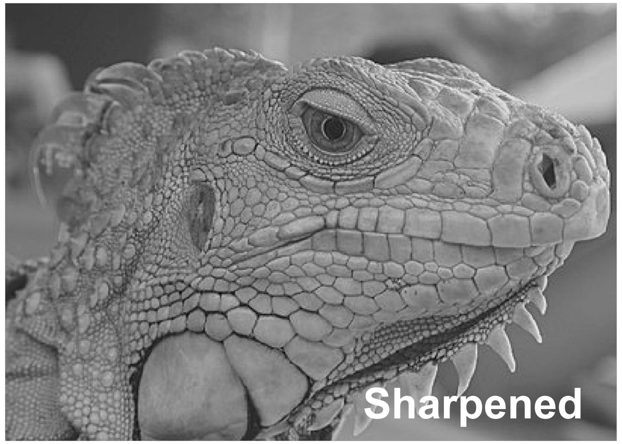
–

=

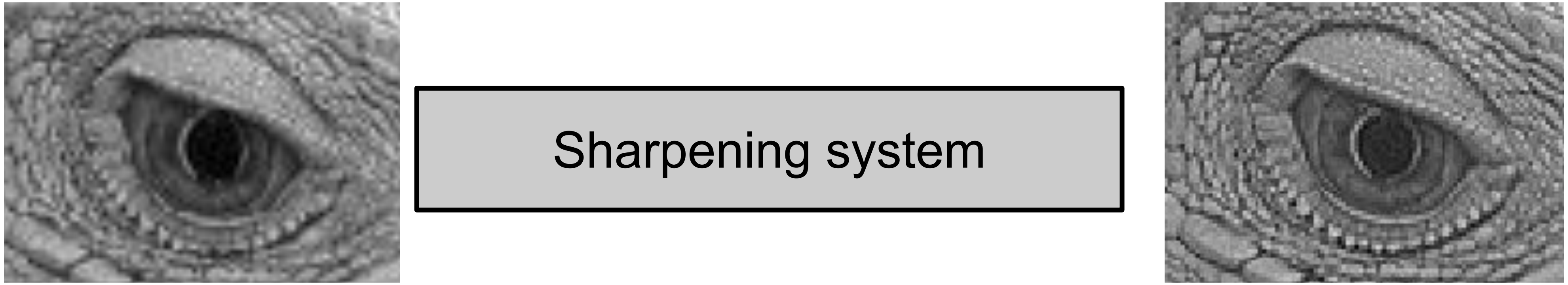
Let’s add it back to get a sharpening system:

+

=



Convolution in 2D – Sharpening filter

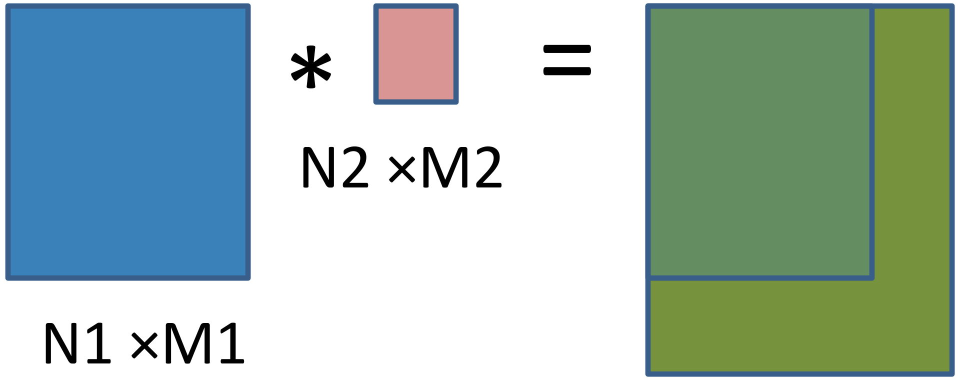


Original

Sharpening system: Accentuates differences with local average

Implementation detail: Image support and edge effect

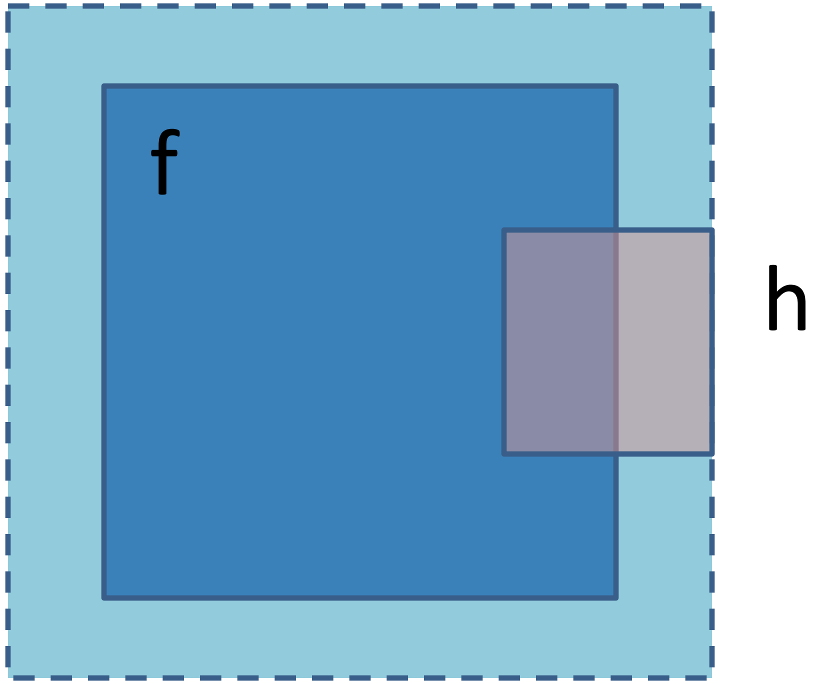
- A computer will only convolve **finite support signals.**
  - That is: images that are zero for n,m outside some rectangular region
- numpy's convolution performs 2D convolution of finite-support signals.



$(N1 + N2 - 1) \times (M1 + M2 - 1)$

Image support and edge effect

- A computer will only convolve **finite support signals.**
- What happens at the edge?



- zero "padding"
- edge value replication
- mirror extension
- more (beyond the scope of this class)

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

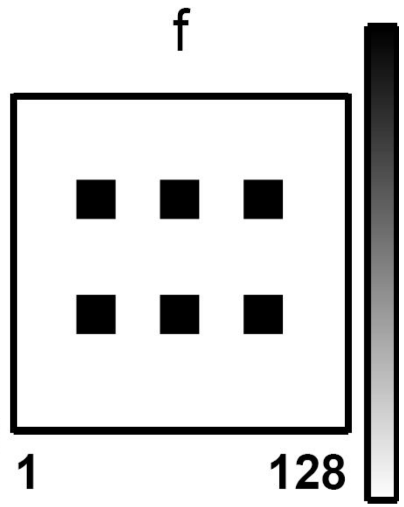
(Cross) correlation – symbol: $**$

Cross correlation of two 2D signals f[n,m] and h[n,m]

$$f[n,m] ** h[n,m] = \sum_{k} \sum_{l} f[k,l] h[n+k, m+l]$$

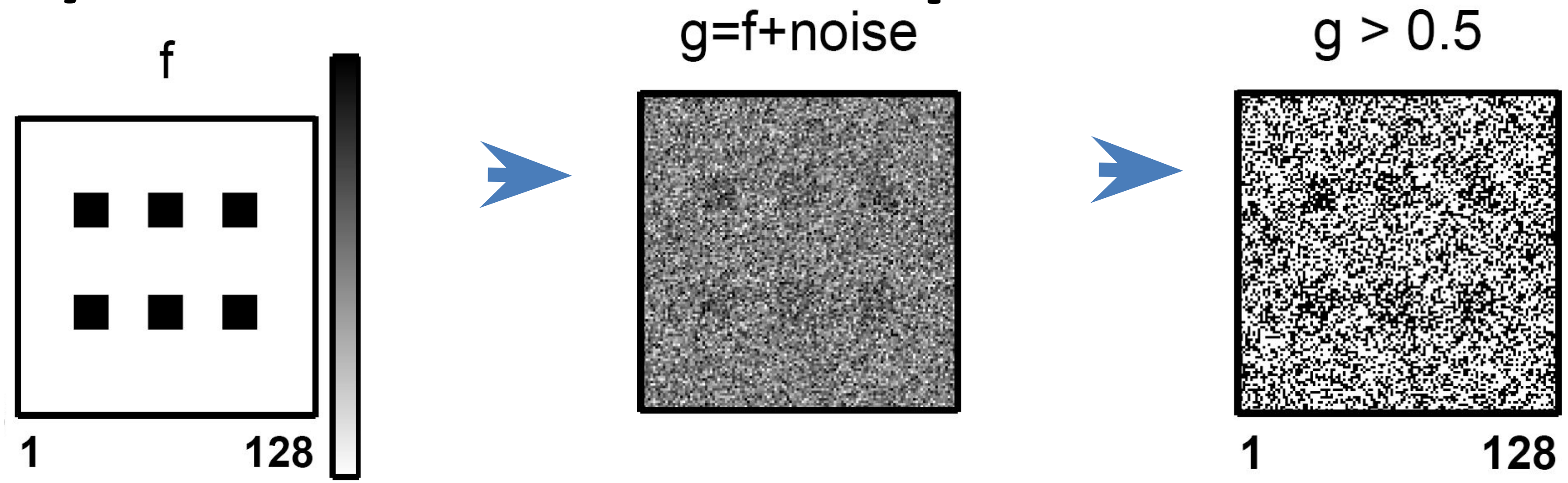
- Equivalent to a convolution without the flip
- Use it to measure 'similarity' between f and h.

(Cross) correlation – example



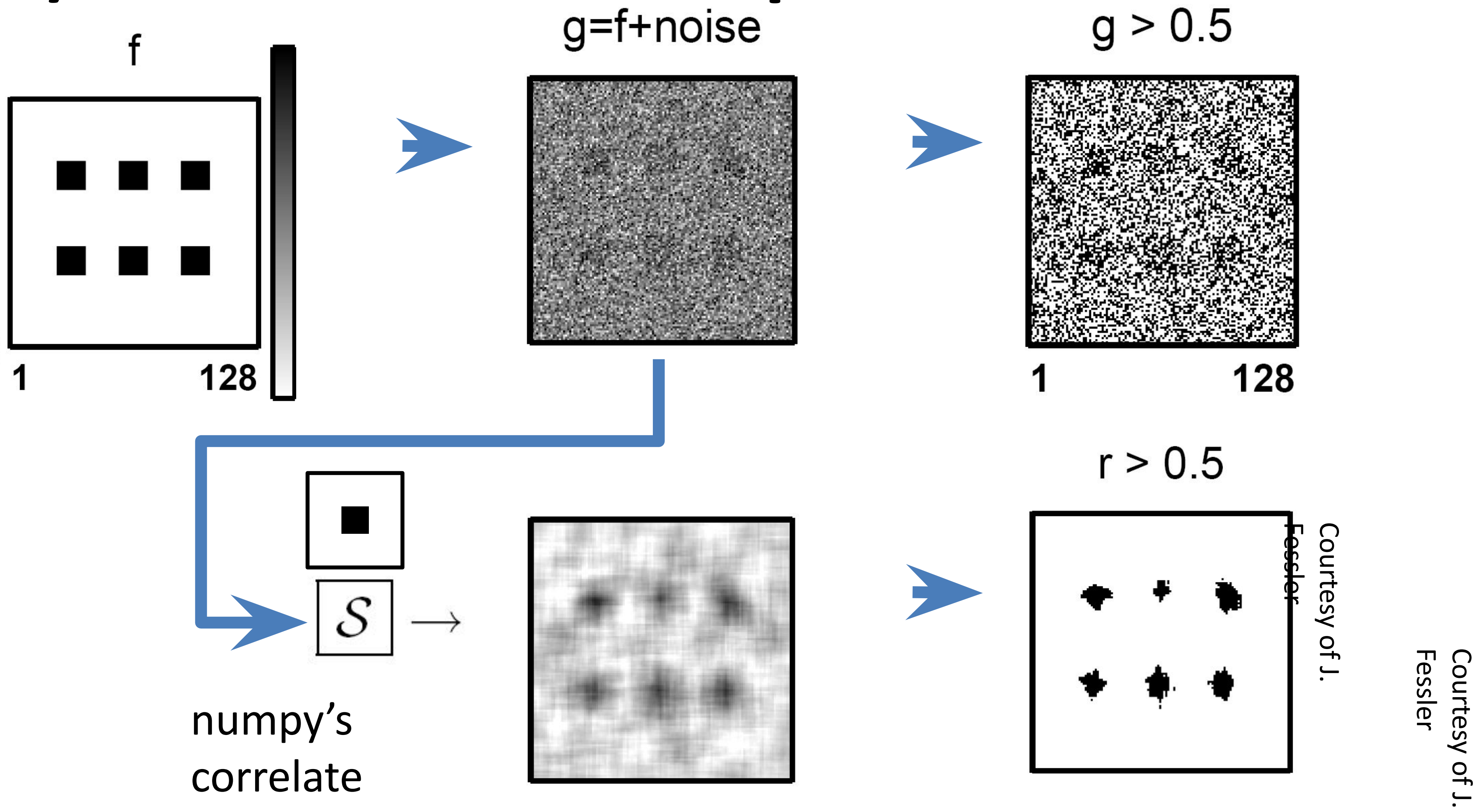
Courtesy of J. Fessler

(Cross) correlation – example

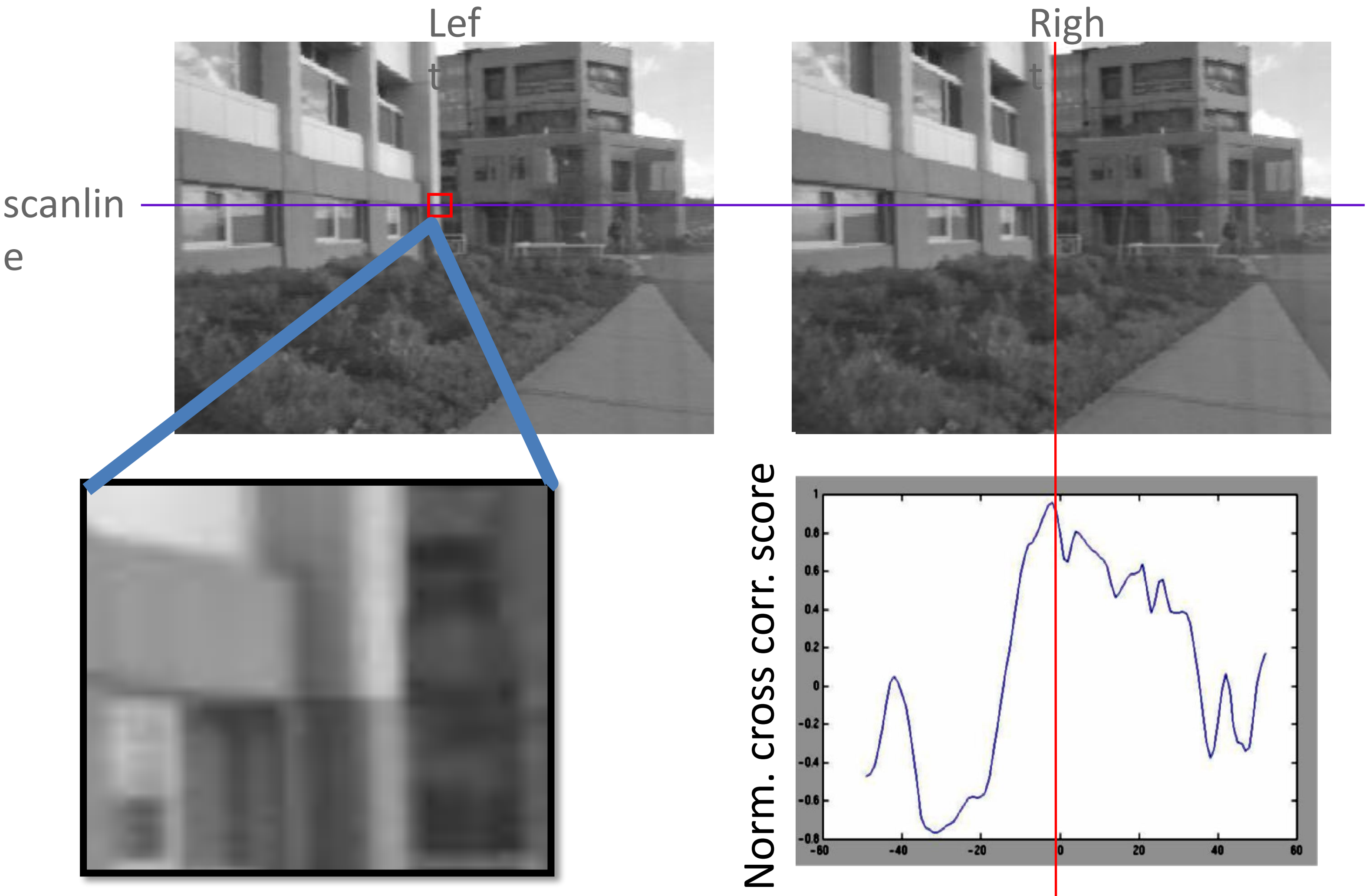


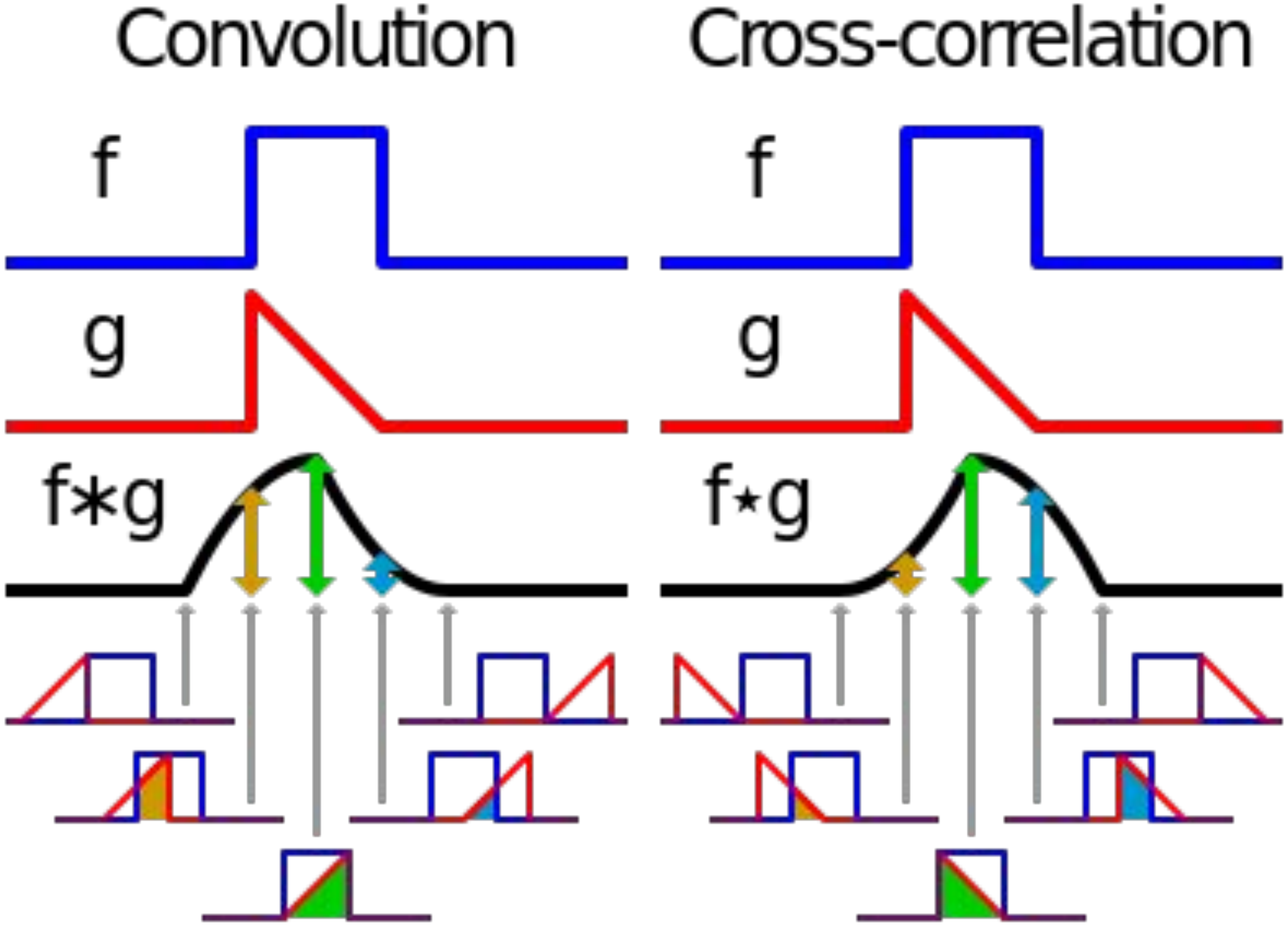
Courtesy of J. Fessler

(Cross) correlation – example



(Cross) correlation – example




Convolution
Cross-correlation
f
g
f∗g
f
g
f⋆g

Cross Correlation Application: Vision system for TV remote control

- uses template matching

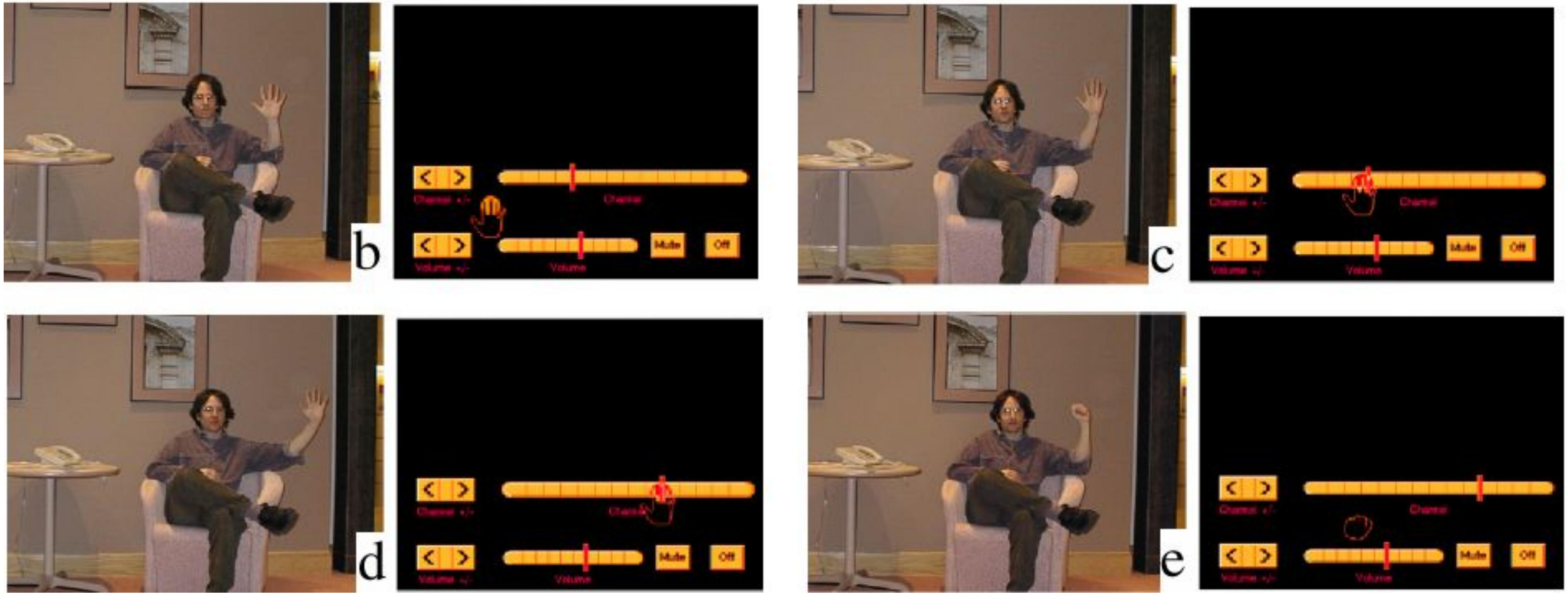


Figure from "Computer Vision for Interactive Computer Graphics," W.Freeman et al, IEEE Computer Graphics and Applications, 1998 copyright 1998, IEEE

Properties of cross correlation

- Associative property:

$$(f ** h_1) ** h_2 = f ** (h_1 ** h_2)$$

- Distributive property:

$$f ** (h_1 + h_2) = (f ** h_1) + (f ** h_2)$$

The order doesn't matter! $h_1 ** h_2 = h_2 ** h_1$

Convolution vs. (Cross) Correlation

- When is correlation equivalent to convolution?
- In other words, Q. when is f**g = f*g?

Convolution vs. (Cross) Correlation

- A **<u>convolution</u>** is an integral that expresses the amount of overlap of one function as it is shifted over another function.
  - convolution is a **filtering** operation

- **<u>Correlation</u>** compares the ***similarity*** *of* ***two*** *sets of* ***data.*** Correlation computes a measure of similarity of two input signals as they are shifted by one another. The correlation result reaches a maximum at the time when the two signals match best .
  - correlation is a measure of relatedness of two signals

What we will learn today?

- Linear shift invariant systems
- Impulse functions
- LSI + impulse response
- Convolutions and Cross-Correlation

Next time:

Edges and lines