

2-D Finite Impulse Response (FIR) Filters

- Difference equation

$$y(m, n) = \sum_{k=-N}^N \sum_{l=-N}^N h(k, l) x(m - k, n - l)$$

- For $N = 2$ - ○ input points; × output point

$$\begin{array}{ccccc} \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \times & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \end{array}$$

- Number of multiplies per output point

$$\text{Multiplies} = (2N + 1)^2$$

- Transfer function

$$H(z_1, z_2) = \sum_{k=-N}^N \sum_{l=-N}^N h(k, l) z_1^{-k} z_2^{-l}$$

$$H(e^{j\mu}, e^{j\nu}) = \sum_{k=-N}^N \sum_{l=-N}^N h(k, l) e^{-j(k\mu + l\nu)}$$

Spatial FIR Smoothing Filtering

- Filter point spread function (PSF) or impulse response:
The box, $\boxed{X}$, indicates the center element of the filter.

$$\begin{array}{ccc} 1 & 2 & 1 \\ 2 & \boxed{4} & 2 \cdot \frac{1}{16} \\ 1 & 2 & 1 \end{array}$$

- Apply filter using free boundary condition: Assume that pixels outside the image are 0.

0 0 0 0 0 0 0		0 0 0 0 0 0 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 3 4 4 3
0 0 0 16 16 16 16	$\Rightarrow$	0 0 3 9 12 12 9
0 0 0 16 16 16 16		0 0 4 12 16 16 12
0 0 0 16 16 16 16		0 0 4 12 16 16 12
0 0 0 16 16 16 16		0 0 3 9 12 12 9
$\underbrace{\hspace{10em}}$ Input Image		$\underbrace{\hspace{10em}}$ Output Image

PSF for FIR Smoothing Filter

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & \boxed{4} & 2 \\ 1 & 2 & 1 \end{bmatrix} \cdot \frac{1}{16}$$

Spatial FIR Horizontal Derivative Filtering

- Filter point spread function (PSF) or impulse response:
The box, $\boxed{X}$, indicates the center element of the filter.

$$\begin{array}{ccc} 2 & 0 & -2 \\ 4 & \boxed{0} & -4 \\ 2 & 0 & -2 \end{array} \cdot \frac{1}{16}$$

- Apply filter using free boundary condition: Assume that pixels outside the image are 0.

$$\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 16 & 16 & 16 & 16 & \Rightarrow 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 16 & 16 & 16 & 16 & 0 & 0 & 8 & 8 & 0 & 0 & -8 \\ 0 & 0 & 0 & 16 & 16 & 16 & 16 & 0 & 0 & 8 & 8 & 0 & 0 & -8 \\ 0 & 0 & 0 & 16 & 16 & 16 & 16 & 0 & 0 & 6 & 6 & 0 & 0 & -6 \end{array}$$

$\underbrace{\hspace{15em}}$ Input Image
 $\underbrace{\hspace{15em}}$ Output Image

PSF of FIR Horizontal Derivative Filter

$$\begin{bmatrix} 2 & 0 & -2 \\ 4 & \boxed{0} & -4 \\ 2 & 0 & -2 \end{bmatrix} \cdot \frac{1}{16}$$

Spatial FIR Vertical Derivative Filtering

- Filter point spread function (PSF) or impulse response:
The box, $\boxed{X}$, indicates the center element of the filter.

$$\begin{array}{ccc} 2 & 4 & 2 \\ 0 & \boxed{0} & 0 \\ -2 & -4 & -2 \end{array} \cdot \frac{1}{16}$$

- Apply filter using free boundary condition: Assume that pixels outside the image are 0.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	2	6	8	8	6
0	0	0	16	16	16	16	$\Rightarrow$	0	0	2	6	8	8	6
0	0	0	16	16	16	16		0	0	0	0	0	0	0
0	0	0	16	16	16	16		0	0	0	0	0	0	0
0	0	0	16	16	16	16		0	0	-2	-6	-8	-8	-6
Input Image								Output Image						

PSF of FIR Vertical Derivative Filter

$$\begin{array}{ccc} 2 & 4 & 2 \\ 0 & \boxed{0} & 0 \\ -2 & -4 & -2 \end{array} \cdot \frac{1}{16}$$

Example 1: 2-D FIR Filter

- Consider the impulse response $h(m, n) = h_1(m)h_1(n)$ where

$$4h_1(n) = (\dots, 0, 1, 2, 1, 0, \dots)$$

$$h_1(n) = (\delta(n+1) + 2\delta(n) + \delta(n-1))/4$$

Then $h(m, n)$ is a separable function with

$$16h(m, n) = \begin{array}{cccccc} & \vdots & \vdots & \vdots & \vdots & \vdots \\ & 0 & 0 & 0 & 0 & 0 \\ \dots & 0 & 1 & 2 & 1 & 0 & \dots \\ \dots & 0 & 2 & 4 & 2 & 0 & \dots \\ \dots & 0 & 1 & 2 & 1 & 0 & \dots \\ & 0 & 0 & 0 & 0 & 0 \\ & \vdots & \vdots & \vdots & \vdots & \vdots \end{array}$$

- The DTFT of $h_1(n)$ is

$$\begin{aligned} H_1(e^{j\omega}) &= \frac{1}{4} (e^{j\omega} + 2 + e^{-j\omega}) \\ &= \frac{1}{2} (1 + \cos(\omega)) \end{aligned}$$

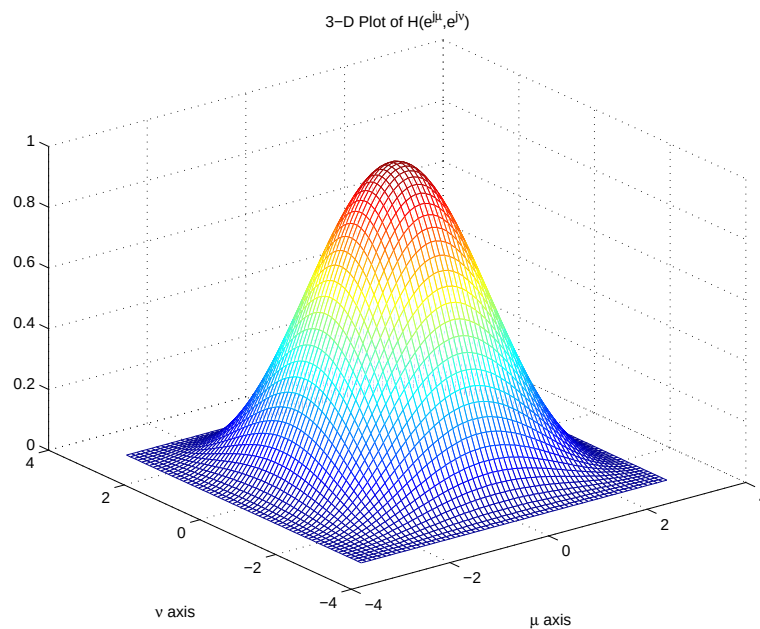
- The DSFT of $h(m, n)$ is

$$\begin{aligned} H(e^{j\mu}, e^{j\nu}) &= H_1(e^{j\mu})H_1(e^{j\nu}) \\ &= \frac{1}{4} (1 + \cos(\mu)) (1 + \cos(\nu)) \end{aligned}$$

Example 1: Frequency Response of 2-D FIR Filter

- Plot of frequency response

$$H(e^{j\mu}, e^{j\nu}) = \frac{1}{4} (1 + \cos(\mu)) (1 + \cos(\nu))$$



- This is a low pass filter with $H(e^{j0}, e^{j0}) = 1$

Example 2: 2-D FIR Filter

- Consider the impulse response $h(m, n) = h_1(m)h_1(n)$ where

$$4h_1(n) = (\dots, 0, 1, -2, 1, 0, \dots)$$

$$h_1(n) = (\delta(n+1) - 2\delta(n) + \delta(n-1))/4$$

Then $h(m, n)$ is a separable function with

$$16h(m, n) = \begin{array}{cccccc} & \vdots & \vdots & \vdots & \vdots & \vdots \\ & 0 & 0 & 0 & 0 & 0 \\ \dots & 0 & 1 & -2 & 1 & 0 & \dots \\ \dots & 0 & -2 & 4 & -2 & 0 & \dots \\ \dots & 0 & 1 & -2 & 1 & 0 & \dots \\ & 0 & 0 & 0 & 0 & 0 \\ & \vdots & \vdots & \vdots & \vdots & \vdots \end{array}$$

- The DTFT of $h_1(n)$ is

$$H_1(e^{j\omega}) = \frac{1}{4}(e^{j\omega} - 2 + e^{-j\omega})$$

$$= -\frac{1}{2}(1 - \cos(\omega))$$

- The DSFT of $h(m, n)$ is

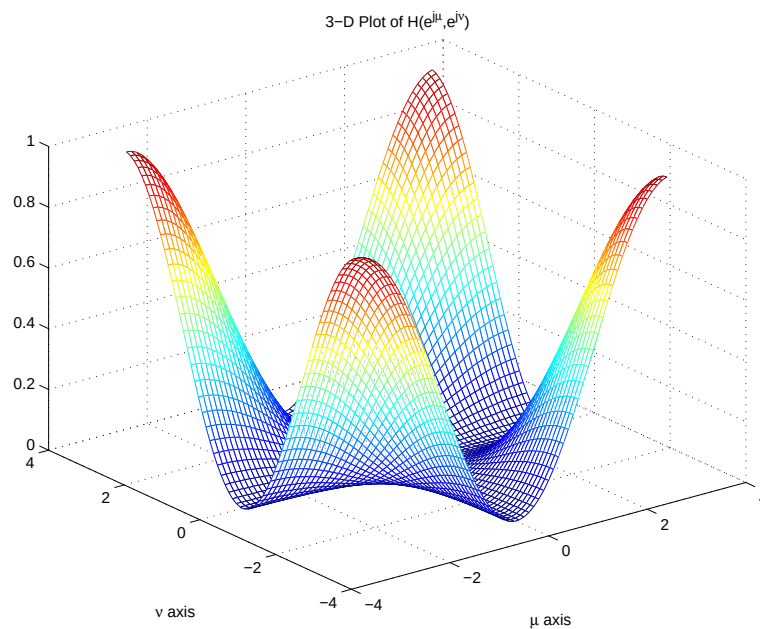
$$H(e^{j\mu}, e^{j\nu}) = H_1(e^{j\mu})H_1(e^{j\nu})$$

$$= \frac{1}{4}(1 - \cos(\mu))(1 - \cos(\nu))$$

Example 2: Frequency Response of 2-D FIR Filter

- Plot of frequency response

$$H(e^{j\mu}, e^{j\nu}) = \frac{1}{4} (1 - \cos(\mu)) (1 - \cos(\nu))$$



- This is a **high pass** filter with $H(e^{j0}, e^{j0}) = 0$

Ordering of Points in a Plane

- Recursive filter implementations require the ordering of points in the plane.
- Let $s = (s_1, s_2) \in \mathbb{Z}^2$ and $r = (r_1, r_2) \in \mathbb{Z}^2$.
- Quarter plane - then $s < r$ means:

$$(s_2 < r_2) \text{ and } (s_1 < r_1) \text{ and } s \neq r$$

○ ○ ○
○ ○ ○
○ ○ ×

- Symmetric half plane - then $s < r$ means:

$$(s_2 < r_2)$$

○ ○ ○ ○ ○
○ ○ ○ ○ ○
×

- Nonsymmetric half plane - then $s < r$ means:

$$(s_2 < r_2) \text{ or } ((s_2 = r_2) \text{ and } (s_1 < r_1))$$

○ ○ ○ ○ ○
○ ○ ○ ○ ○
○ ○ ×

2-D Infinite Impulse Response (IIR) Filters

- Difference equation

$$\begin{aligned}
 y(m, n) = & \sum_{k=-N}^N \sum_{l=-N}^N b(k, l) x(m - k, n - l) \\
 & + \sum_{k=-P}^P \sum_{l=1}^P a(k, l) y(m - k, n - l) \\
 & + \sum_{k=1}^P a(k, 0) y(m - k, n)
 \end{aligned}$$

Simplified notation

$$y_s = \sum_r b_r x_{s-r} + \sum_{r > (0,0)} a_r y_{s-r}$$

- For nonsymmetric half plane with $N = 0$ and $P = 2$

$$\begin{array}{ccccc}
 \circ & \circ & \circ & \circ & \circ \\
 \circ & \circ & \circ & \circ & \circ \\
 \circ & \circ & \times & &
 \end{array}$$

- Number of multiplies per output point

$$\text{Multiplies} = \underbrace{(2N + 1)^2}_{\text{FIR Part}} + \underbrace{2(P + 1)P}_{\text{IIR Part}}$$

2-D IIR Filter Transfer Functions

- Transfer function in Z-transform domain is

$$H(z_1, z_2) = \frac{\sum_{k=-N}^N \sum_{l=-N}^N b(k, l) z_1^{-k} z_2^{-l}}{1 - \sum_{k=-P}^P \sum_{l=1}^P a(k, l) z_1^{-k} z_2^{-l} - \sum_{k=1}^P a(k, 0) z_1^{-k}}$$

- Transfer function in DSFT domain is

$$H(e^{j\mu}, e^{j\nu}) = \frac{\sum_{k=-N}^N \sum_{l=-N}^N b(k, l) e^{-j(k\mu + l\nu)}}{1 - \sum_{k=-P}^P \sum_{l=1}^P a(k, l) e^{-j(k\mu + l\nu)} - \sum_{k=1}^P a(k, 0) e^{-j(k\mu)}}$$

Example 3: 2-D IIR Filter

- Consider the difference equation

$$y(m, n) = x(m, n) + ay(m-1, n) + ay(m, n-1)$$

- Spatial dependencies - $\circ$ previous value; $\times$ current value

$$\begin{array}{c} \circ \\ \circ \times \end{array}$$

- Taking the Z-transform of the difference equation

$$Y(z_1, z_2) = X(z_1, z_2) + az_1^{-1}Y(z_1, z_2) + az_2^{-1}Y(z_1, z_2)$$

The transfer functions is then

$$H(z_1, z_2) = \frac{Y(z_1, z_2)}{X(z_1, z_2)} = \frac{1}{1 - az_1^{-1} - az_2^{-1}}$$

$$H(e^{j\mu}, e^{j\nu}) = \frac{1}{1 - ae^{-j\mu} - ae^{-j\nu}}$$

Example 3: 2-D IIR Filter in Space Domain

- For $a = 1/2$

$$y(m, n) = x(m, n) + \frac{1}{2}y(m-1, n) + \frac{1}{2}y(m, n-1)$$

- Looks like

$$\frac{1}{2} \times$$

- Apply filter in raster scan order.

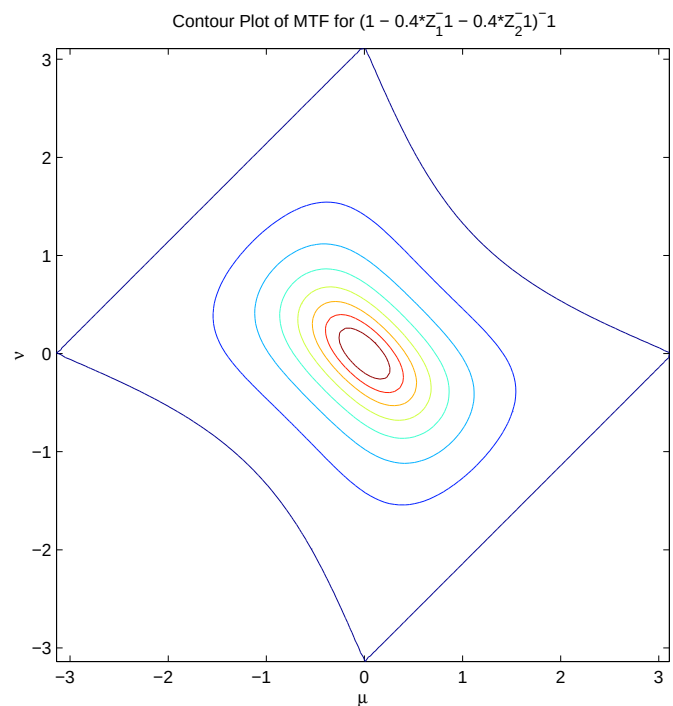
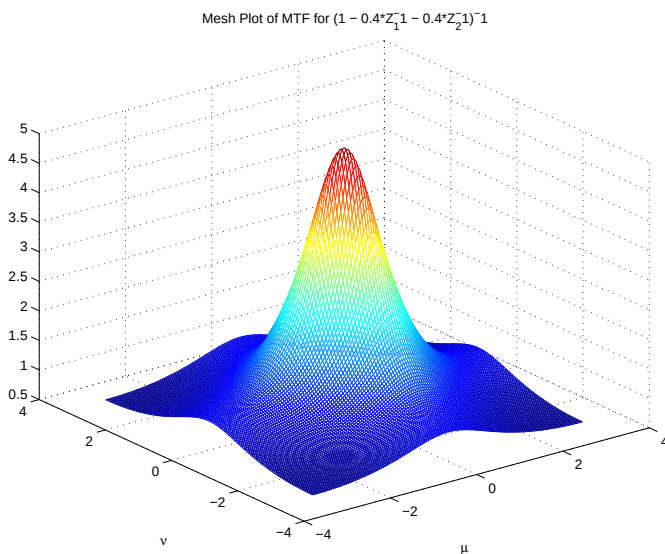
0 0 0 0 0 0 0		0 0 0 0 0 0 0
0 0 0 0 0 0 0		0 0 0 0 0 0 0
0 0 0 0 0 0 0		0 0 0 0 0 0 0
0 0 0 64 0 0 0	$\Rightarrow$	0 0 0 64 32 16 8
0 0 0 0 0 0 0		0 0 0 32 32 24 16
0 0 0 0 0 0 0		0 0 0 16 24 24 20
0 0 0 0 0 0 0		0 0 0 8 16 20 20
$\underbrace{\hspace{10em}}$ Input Image		$\underbrace{\hspace{10em}}$ Output Image

Example 3: Frequency Response of 2-D IIR Filter

- Plot of frequency response

$$H(z_1, z_2) = \frac{1}{1 - az_1^{-1} - az_2^{-1}}$$

for $a = 0.4$.



Example 4: 2-D IIR Filter

- Consider the difference equation

$$y(m, n) = x(m, n) + ay(m-1, n) + ay(m, n-1) + 2ay(m+1, n-1)$$

- Spatial dependencies - $\circ$ previous value; $\times$ current value

$$\begin{array}{cc} \circ & \circ \\ \circ & \times \end{array}$$

- The transfer functions is then

$$H(z_1, z_2) = \frac{1}{1 - az_1^{-1} - az_2^{-1} - 2az_1^{+1}z_2^{-1}}$$

$$H(e^{j\mu}, e^{j\nu}) = \frac{1}{1 - ae^{-j\mu} - ae^{-j\nu} - 2ae^{+j\mu-j\nu}}$$

Example 4: 2-D IIR Filter in Space Domain

- For $a = 1/4$

$$y(m, n) = x(m, n) + \frac{1}{4}y(m-1, n) + \frac{1}{4}y(m, n-1) + \frac{1}{2}y(m+1, n-1)$$

- Looks like

$$\begin{array}{cc} & 1/4 & 1/2 \\ 1/4 & & \times \end{array}$$

- Apply filter in raster scan order.

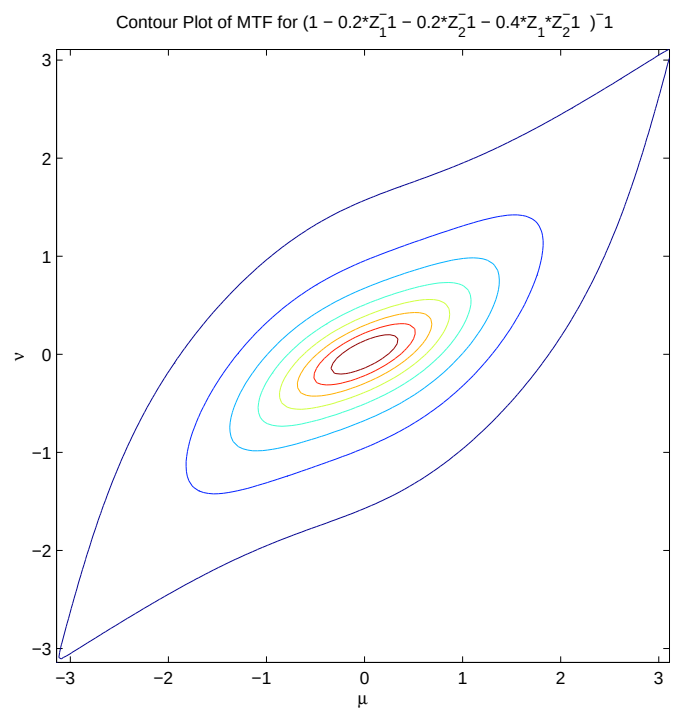
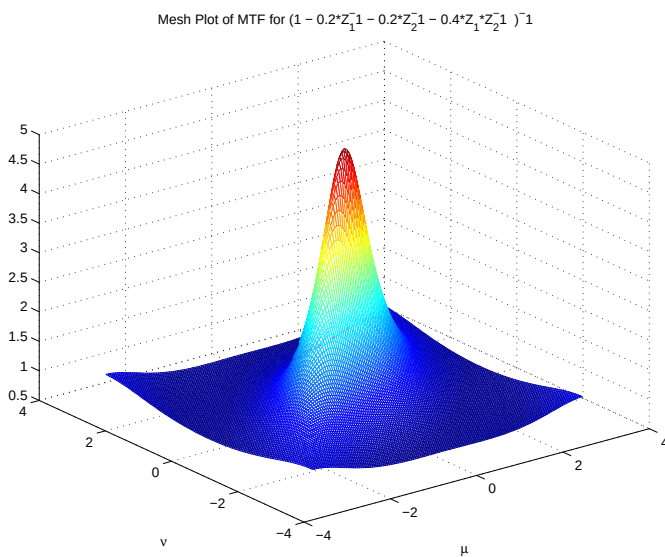
0 0 0 0 0 0 0		0 0 0 0 0 0 0
0 0 0 0 0 0 0		0 0 0 0 0 0 0
0 0 0 0 0 0 0		0 0 0 0 0 0 0
0 0 0 64 0 0 0	$\Rightarrow$	0 0 0 64 16 4 1
0 0 0 0 0 0 0		0 0 32 32 14 5 $1\frac{1}{2}$
0 0 0 0 0 0 0		0 16 28 22 $11\frac{1}{2}$ $4\frac{7}{8}$ $\frac{45}{32}$
$\underbrace{\hspace{10em}}$ Input Image		$\underbrace{\hspace{10em}}$ Output Image

Example 4: Frequency Response of 2-D IIR Filter

- Plot of frequency response

$$H(z_1, z_2) = \frac{1}{1 - az_1^{-1} - az_2^{-1} - 2az_1z_2^{-1}}$$

for $a = 0.2$.



- Notice that transfer function has a diagonal orientation.

Example 5: 2-D IIR Filter

- Consider the difference equation

$$y(m, n) = x(m, n) + ay(m-1, n) + ay(m, n-1) + ay(m+1, n) + ay(m, n+1)$$

- Spatial dependencies - $\circ$ previous value; $\times$ current value

$$\begin{array}{ccc} & \circ & \\ \circ & \times & \circ \\ & \circ & \end{array}$$

- Theoretically, the transfer functions is then

$$H(z_1, z_2) = \frac{1}{1 - az_1^{-1} - az_2^{-1} - az_1 - az_2}$$

$$H(e^{j\mu}, e^{j\nu}) = \frac{1}{1 - ae^{-j\mu} - ae^{-j\nu} - ae^{j\mu} - ae^{j\nu}}$$

- THIS DOESN'T WORK**