Digital Image Processing

COSC 6380/4393

Midterm Review
(Continued…)

Oct 30th, 2023

Mid Term Exam

- Syllabus:
  - Introduction
  - Binary Image Processing
  - Point Operations
  - Discrete Fourier Transform
  - Spatial Filtering

Discrete Fourier Transform

1. Compute the DFT of the following matrix

1	31
20	7

1. Prove that the DFT of a 2D matrix is Conjugate symmetric
    1. The magnitude of the DFT matrix is symmetric
2. Prove that the DFT of an image is periodic in nature (Periodic extension of DFT)

4. The IDFT is periodic in nature (Periodic extension of IDFT)

Example

$$I = \begin{array}{|c|c|}\hline 5 & 7 \\ \hline 8 & 3 \\ \hline \end{array}$$

$$\tilde{I}(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, e^{-\sqrt{-1}\frac{2\pi}{N}(ui+vj)}$$

$$\tilde{I}(0,0) = \sum_{i=0}^{2-1}\sum_{j=0}^{2-1} I(i,j)\, e^{-\sqrt{-1}\frac{2\pi}{2}(0*i+0*j)}$$

$$= \sum_{i=0}^{1}\sum_{j=0}^{1} I(i,j) = 21$$

$\tilde{I}(0,1) = 3.+0.j$

$\tilde{I}(1,0) = 1.+0.j$

$\tilde{I}(1,1) =$-7.+0.j

23	
	-7.+0.j

Example

$$I = \begin{array}{|c|c|}\hline 5 & 7 \\ \hline 8 & 3 \\ \hline \end{array}$$

$$\tilde{I}(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, e^{-\sqrt{-1}\frac{2\pi}{N}(ui+vj)}$$

$$\tilde{I}(0,0) = \sum_{i=0}^{2-1}\sum_{j=0}^{2-1} I(i,j)\, e^{-\sqrt{-1}\frac{2\pi}{2}(0*i+0*j)}$$

$$= \sum_{i=0}^{1}\sum_{j=0}^{1} I(i,j) = 21$$

$\tilde{I}(0,1) = 3.+0.j$

$\tilde{I}(1,0) = 1.+0.j$

$\tilde{I}(1,1) =$-7.+0.j

DFT =

23	3
1	-7

1. Compute the DFT of the following matrix

1	31
20	7

DFT =

59.+0.j	-17.+0.j
5.+0.j	-43.+0.j

2D Discrete Fourier Transform

- We will use the abbreviation

$$W_N = e^{-\sqrt{-1}\frac{2\pi}{N}} \Rightarrow \boldsymbol{W_N^{ui+vj}} = \boldsymbol{e^{-\sqrt{-1}\frac{2\pi}{N}(ui+vj)}}$$

- Then

$$\tilde{I}(u,v) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j)\, e^{-\sqrt{-1}\frac{2\pi}{N}(ui+vj)}$$

2D Discrete Fourier Transform

- We will use the abbreviation

$$W_N = e^{-\sqrt{-1}\frac{2\pi}{N}} \Rightarrow \boldsymbol{W_N^{ui+vj}} = \boldsymbol{e^{-\sqrt{-1}\frac{2\pi}{N}(ui+vj)}}$$

- Then

$$\tilde{I}(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, e^{-\sqrt{-1}\frac{2\pi}{N}(ui+vj)}$$

$$= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, \boldsymbol{W_N^{ui+vj}}$$

$$I(i,j) = \sum_{u=0}^{N-1}\sum_{u=0}^{N-1} \tilde{I}(u,v)\boldsymbol{W_N^{-(ui+vj)}}$$

Symmetry of DFT

$$\tilde{I}(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\,\boldsymbol{W_N^{ui+vj}}$$

$$\begin{aligned}
\tilde{I}(N-u, N-v) &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, W_N^{[(N-u)i+(N-v)j]} \\
&= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, W_N^{N(i+j)}\, W_N^{-(ui+vj)} \\
&= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\, \left[W_N^{(ui+vj)}\right]^* = \tilde{I}^*(u,v)
\end{aligned}$$

since

$$W_N^{N(i+j)} = e^{-\sqrt{-1}\,\frac{2\pi}{N}\cdot N(i+j)} = e^{-2\pi\sqrt{-1}\,(i+j)} = 1^{(i+j)} = 1 \text{ for any } i, j$$

and

$$W_N^{-(ui+vj)} = \left[W_N^{(ui+vj)}\right]^*.$$

The DFT of an image **I** is **conjugate symmetric**:

$$\begin{aligned}
\tilde{I}_{real}(N-u, N-v) &= \tilde{I}_{real}(u,v)\ ;\ 0 \le u, v \le N-1 \\
\tilde{I}_{imag}(N-u, N-v) &= -\tilde{I}_{imag}(u,v)\ ;\ 0 \le u, v \le N-1
\end{aligned}$$

Symmetry of DFT

$$\tilde{I}(N-u, N-v) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j)\, W_N^{[(N-u)i+(N-v)j]}$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j)\, W_N^{N(i+j)}\, W_N^{-(ui+vj)}$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j)\, [W_N^{(ui+vj)}]^* = \tilde{I}^*(u,v)$$

since

$$W_N^{N(i+j)} = e^{-\sqrt{-1}\,\frac{2\pi}{N}\cdot N(i+j)} = e^{-2\pi\sqrt{-1}\,(i+j)} = 1^{(i+j)} = 1 \text{ for any } i, j$$

and

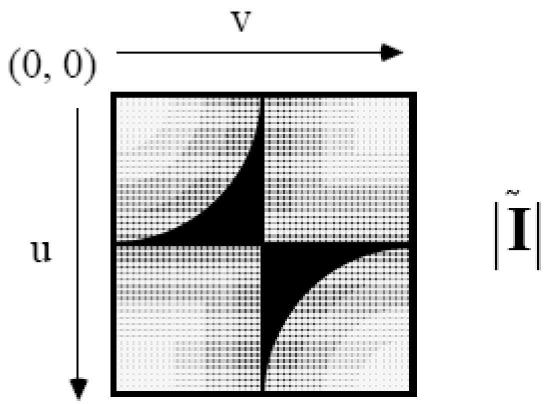
$$W_N^{-(ui+vj)} = [W_N^{(ui+vj)}]^*.$$

The DFT of an image **I** is **conjugate symmetric**:

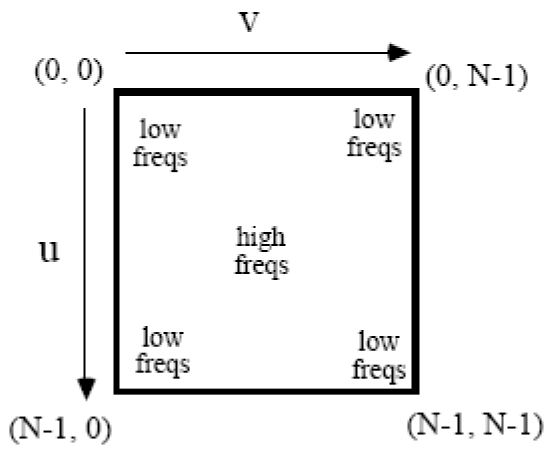
$|\tilde{I}(N-u, N-v)| = |\tilde{I}(u,v)|$ The magnitude DFT of an image **I** is **symmetric**:

Symmetry of DFT

- Depiction of the symmetry of the DFT (magnitude).



- The highest frequencies are represented near (u, v) = (N/2, N/2).



Periodicity of DFT

- We have defined the DFT matrix as **finite** in extent (N x N):

$$\tilde{\mathbf{I}} = \left[\tilde{I}(u, v)\ ;\ 0 \leq u, v \leq N\text{-}1\right]$$

- However, if the arguments are allowed to take values outside the range $0 \leq u, v \leq N\text{-}1$, we find that the DFT is periodic in both the u- and v-directions, with **period N**:
- For any integers m, n

$$\tilde{I}(u+nN, v+mN) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i, j)\ W_N^{[(u+nN)i+(v+mN)j]}$$

$$= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i, j)\ W_N^{N(ni+mj)}\ W_N^{(ui+vj)}$$

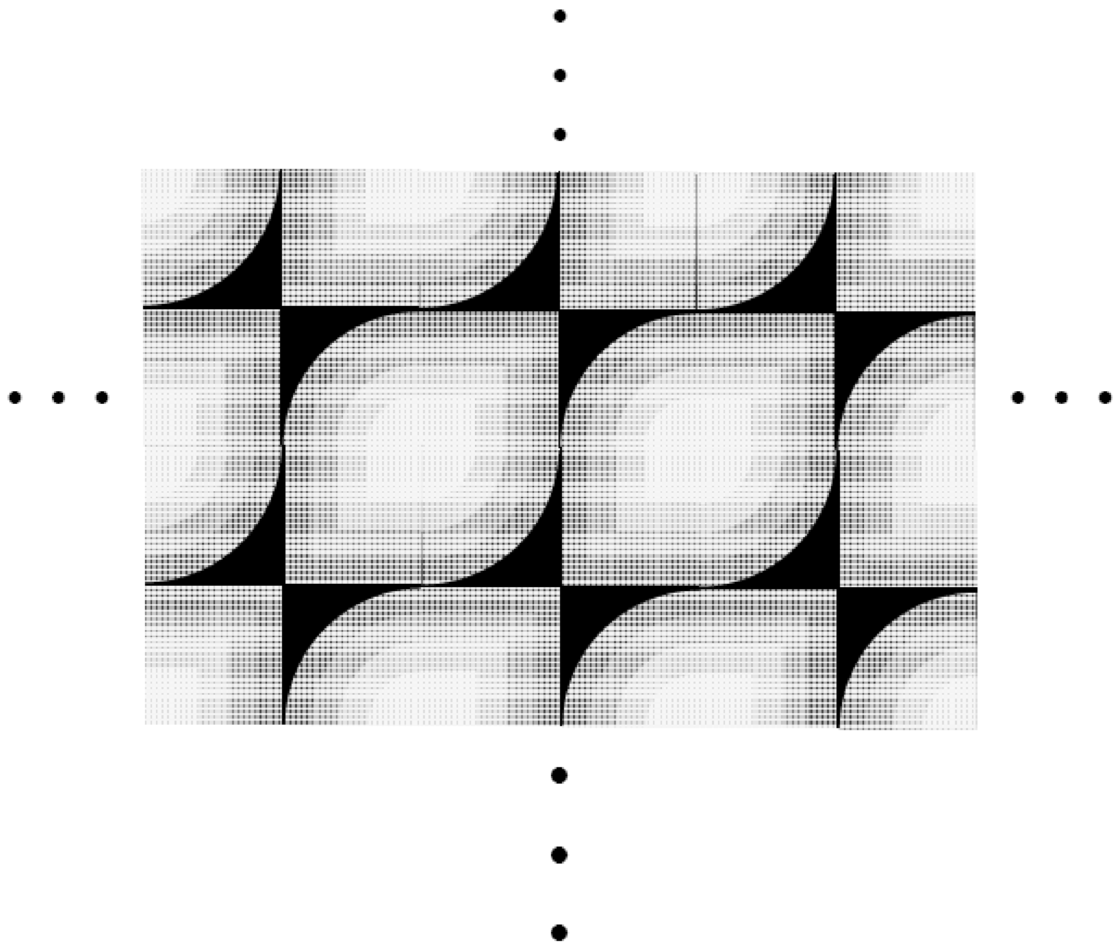
$$= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i, j)\ W_N^{(ui+vj)} = \tilde{I}(u, v)$$

since

$$W_N^{N(ni+mj)} = e^{-\sqrt{-1}\,\frac{2\pi}{N}\cdot N(ni+mj)} = e^{-2\pi\sqrt{-1}\,(ni+mj)} = 1^{(ni+mj)} = 1$$

- This is called the **periodic extension** of the DFT. It is defined for all integer frequencies u, v.

Periodic Extension of DFT



Periodic Extension of Image

- The IDFT equation $$I(i, j) = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \tilde{I}(u, v)\, W_N^{-(ui+vj)}$$

  implies the **periodic extension of the image I** as well (with period N), simply by letting the arguments (i, j) take any integer value.
- Note that for any integers n, m

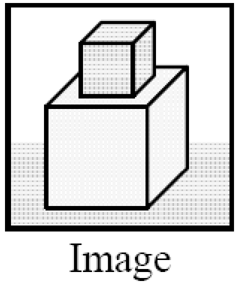
$$\begin{aligned} I(i+nN, j+mN) &= \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \tilde{I}(u, v)\, W_N^{-[u(i+nN)+v(j+mN)]} \\ &= \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \tilde{I}(u, v)\, W_N^{-(ui+vj)}\, W_N^{-N(nu+mv)} \\ &= \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \tilde{I}(u, v)\, W_N^{-(ui+vj)} = I(i, j) \end{aligned}$$

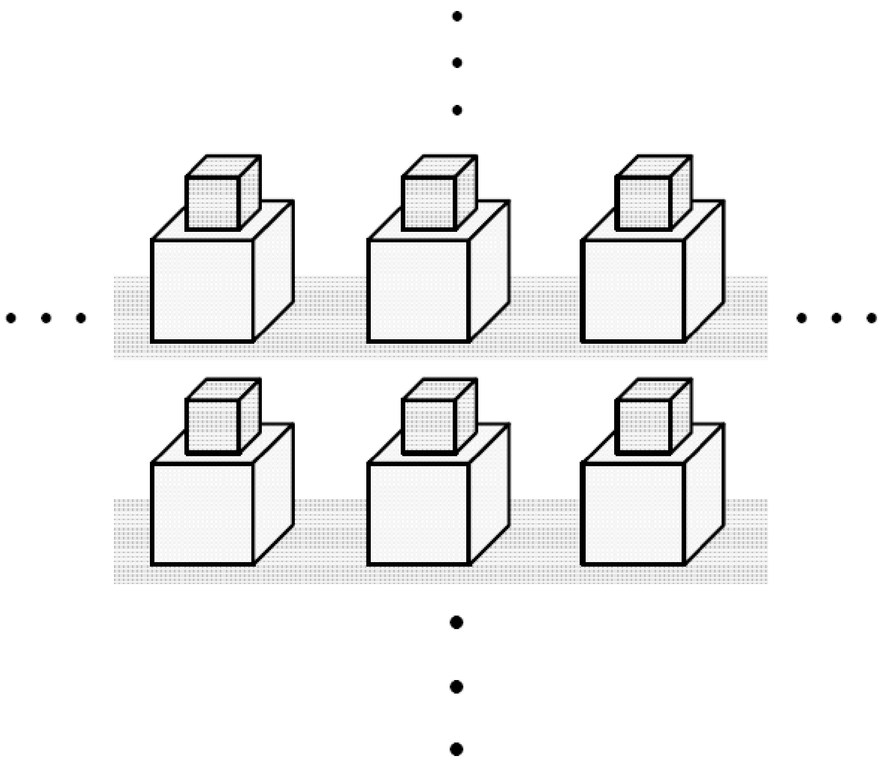
since

$$W_N^{-N(nu+mv)} = e^{-\sqrt{-1}\,\frac{2\pi}{N}\cdot N(nu+mv)} = e^{-2\pi\sqrt{-1}\,(nu+mv)} = 1^{(nu+mv)} = 1$$

- In a sense, the DFT **implies** that the image **I** is already periodic.
- This will be extremely important when we consider **convolution**

Periodic Extension of Image





Displaying the DFT

- Usually, the DFT is displayed with its center coordinate (u, v) = (0, 0) at the center of the image.
- This way, the lower frequency information (which usually dominates an image) is clustered together near the origin at the center of the display.

$$\tilde{I}(u - \tfrac{N}{2}, v - \tfrac{N}{2}) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i, j)\, W_N^{[(u-N/2)i+(v-N/2)j]}$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i, j)\, W_N^{(ui+vj)}\; W_N^{-N(i+j)/2}$$

$$W_N^{-N(i+j)/2} = e^{\sqrt{-1}\,\frac{2\pi}{N}\,N(i+j)/2} = e^{\sqrt{-1}\pi\,(i+j)} = (-1)^{i+j}$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i, j)\, (-1)^{i+j}\, W_N^{(ui+vj)}$$

$$= \mathrm{DFT}\left[(-1)^{i+j} I(i, j)\right]$$

- This can be accomplished in practice by taking the DFT of the alternating image (for display purposes only!)

$$[(-1)^{i+j} I(i,j)\ ;\ 0 \le\ i, j \le\ N-1]$$

Displaying the DFT

- Usually, the DFT is displayed with its center coordinate (u, v) = (0, 0) at the center of the image.
- This way, the lower frequency information (which usually dominates an image) is clustered together near the origin at the center of the display.
- This can be accomplished in practice by taking the DFT of the alternating image (for display purposes only!)

$$[(-1)^{i+j} I(i,j)\ ;\ 0 \le\ i, j \le\ N-1]$$

- Observe that

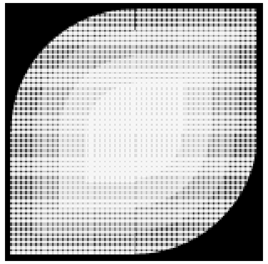
$$(-1)^{i+j} = e^{\sqrt{-1}\pi\,(i+j)} = e^{\sqrt{-1}\,\frac{2\pi}{N}\,N(i+j)/2} = W_N^{-N(i+j)/2}$$

so

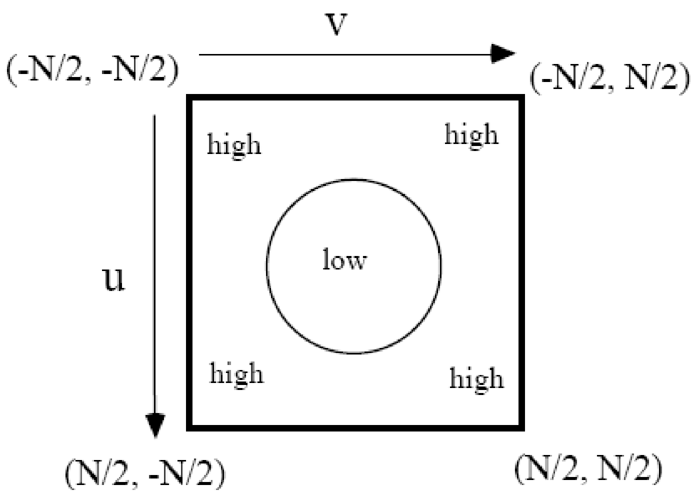
$$\begin{aligned} \mathrm{DFT}\left[(-1)^{i+j} I(i,j)\right] &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\,(-1)^{i+j}\,W_N^{(ui+vj)} \\ &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\,W_N^{(ui+vj)}\,W_N^{-N(i+j)/2} \\ &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\,W_N^{[(u-N/2)i+(v-N/2)j]} \\ &= \tilde{I}(u - \tfrac{N}{2}, v - \tfrac{N}{2}) \end{aligned}$$

- A simple shift of the DF

Centered DFT



Centered DFT



Spatial Filtering

Filtering

1. Apply convolution with filter w on image I
2. Apply a 3X3 spatial smoothing filter on the image I
3. Sharpen Image using a
   1. Laplacian filter (3X3 filter)
   2. Unsharp mask (3X3 filter)

$$\mathbf{I} = \begin{array}{|c|c|c|c|} \hline 1 & 1 & 3 & 4 \\ \hline 2 & 5 & 3 & 2 \\ \hline 8 & 1 & 8 & 2 \\ \hline 4 & 5 & 3 & 11 \\ \hline \end{array}$$

Spatial Convolution Operator

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

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

Smoothing Spatial Filters

Smoothing filters are used for blurring and for noise reduction

Blurring is used in removal of small details and bridging of small gaps in lines or curves

Smoothing spatial filters include linear filters and nonlinear filters.

Two Smoothing Averaging Filter Masks

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

a b

FIGURE 3.32 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to 1 divided by the sum of the values of its coefficients, as is required to compute an average.

Convolution

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 & 3 & 4 \\ 2 & 5 & 3 & 2 \\ 8 & 1 & 8 & 2 \\ 4 & 5 & 3 & 11 \end{bmatrix}$$

1. Apply a 3X3 spatial smoothing filter on the image I

   Solution: Convolution with a smoothing filter

Sum of Products

0	0	0	0	0	0
0	1	1	3	4	0
0	2	5	3	2	0
0	8	1	8	2	0
0	4	5	3	11	0
0	0	0	0	0	0

=

1			

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Rotated

Sum of Products

0	0	0	0	0	0
0	1	1	3	4	0
0	2	5	3	2	0
0	8	1	8	2	0
0	4	5	3	11	0
0	0	0	0	0	0

1	15/9		

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Rotated

Note: Return result without 0 padding

Sum of Products

0	0	0	0	0	0
0	1	1	3	4	0
0	2	5	3	2	0
0	8	1	8	2	0
0	4	5	3	11	0
0	0	0	0	0	0

=

1	15/9	18/9	

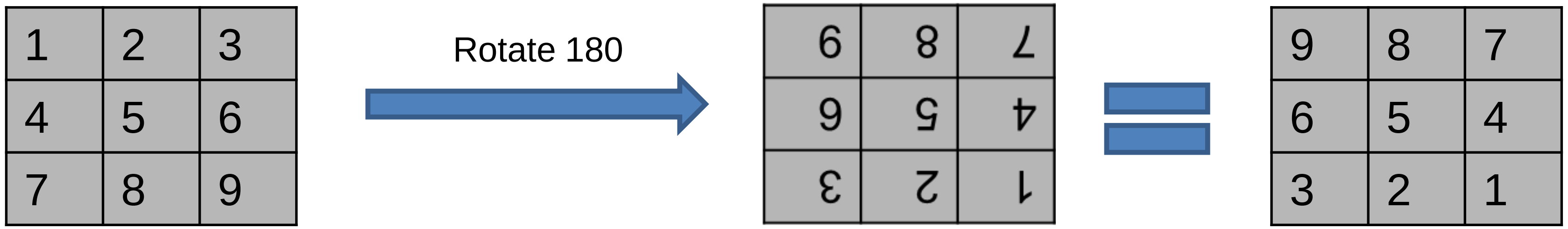
$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Rotated

Note: You need to rotate the filter by 180 degrees for convolution! It's the same if the filter is symmetric (e.g., smoothing filter)

Rotate Filter Example



Laplace Operator

(a)

0	1	0
1	−4	1
0	1	0

(b)

1	1	1
1	−8	1
1	1	1

(c)

0	−1	0
−1	4	−1
0	−1	0

(d)

−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.37
(a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

Demo Filters

Sharpening Spatial Filters: Laplace Operator

Image sharpening in the way of using the Laplacian:

$$g(x, y) = f(x, y) + c\left[\nabla^2 f(x, y)\right]$$

where,

$f(x, y)$ is input image,

$g(x, y)$ is sharpenend images,

$c =$ -1 if $\nabla^2 f(x, y)$ corresponding to Fig. 3.37(a) or (b)

and $c = 1$ if either of the other two filters is used.

Convolution

$$-1 \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 & 3 & 4 \\ 2 & 5 & 3 & 2 \\ 8 & 1 & 8 & 2 \\ 4 & 5 & 3 & 11 \end{bmatrix} + c \begin{bmatrix} 1 & 1 & 3 & 4 \\ 2 & 5 & 3 & 2 \\ 8 & 1 & 8 & 2 \\ 4 & 5 & 3 & 11 \end{bmatrix}$$

$$g(x,y) = f(x,y) + c\left[\nabla^2 f(x,y)\right]$$

Note: You need to remember the filters!

Unsharp Masking and Highboost Filtering

- **Unsharp masking**

  Sharpen images consists of subtracting an unsharp (smoothed) version of an image from the original image

  e.g., printing and publishing industry

- **Steps**

  1. Blur the original image
  2. Subtract the blurred image from the original
  3. Add the mask to the original

Let $\overline{f}(x, y)$ denote the blurred image, unsharp masking is

$$g_{mask}(x, y) = f(x, y) - \overline{f}(x, y)$$

Then add a weighted portion of the mask back to the original

$$g(x, y) = f(x, y) + k * g_{mask}(x, y) \qquad k \geq 0$$

when $k > 1$, the process is referred to as highboost filtering.