Lecture 9
Wavelet and Other Image Transforms

Dr. Xiran Cai

Email: caixr@shanghaitech.edu.cn

Office: 3-438 SIST

Tel: 20684431

ShanghaiTech University

上海科技大学 ShanghaiTech University

Outline

- **2D Unitary transform**
- **Frequency Domain Extension**
  - Discrete Cosine Transform（余弦变换）
  - Hadamard Transform （哈德马变换）
  - Discrete Wavelet Transform （小波变换）
- **Discrete Wavelet Transform (DWT)**
  - An example for 1D-DWT
  - Generalization of 1D-DWT
  - 2D-DWT

Unitary Transform

- **Forward Transform:**

$$t = Af$$

$$t[k] = \sum_{n=0}^{N-1} A[k,n] f[n]$$

- **Inverse Transform:**

$$f = A^H t \quad \text{if } A^H = (A^T)^* \text{ and } AA^H = I$$

Example for 1D Unitary Transform

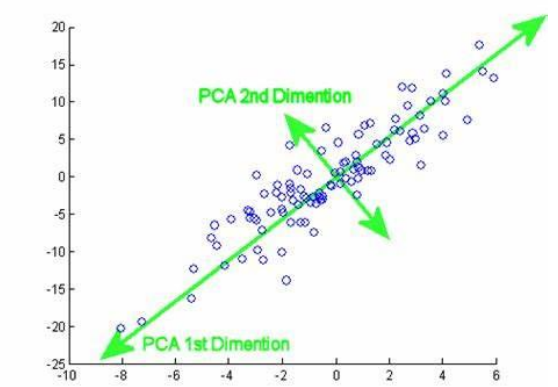
- **Image rotation:**

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix}$$

- **Principle Component Analysis (PCA):**

$Y = PX$ that satisfies $C = XX^T \;\; D = PCP^T$

and $PP^T = I$



Discrete Fourier Transform

- **Forward Transform:**

$$t = Af; \quad t[k] = \sum_{n=0}^{N-1} A[k,n]f[n]$$

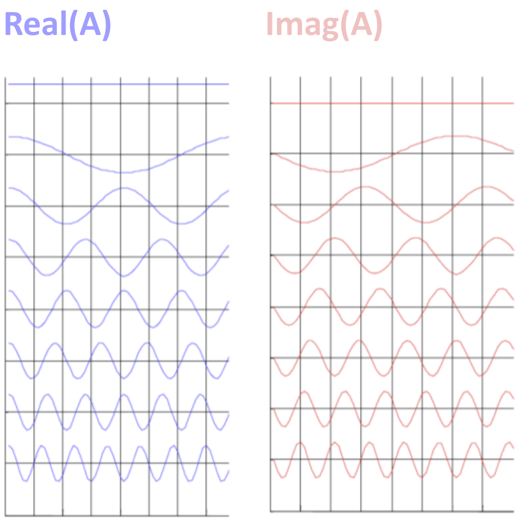
- **Inverse Transform:**

$$f = A^H t; \quad f[n] = \sum_{k=0}^{N-1} A^H[k,n]t[k]$$

- **1-D DFT**

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi kn}{N}}, (k = 1,2,\cdots,N)$$

$$A[k,n] = e^{-j\frac{2\pi kn}{N}} = cos(2\pi \frac{kn}{N})\text{-}jsin(2\pi \frac{kn}{N})$$



2D Unitary Transform

- Forward Transform:

$$F(u,v) = \sum_{x=0}^{M} \sum_{y=0}^{N} f[x,y]\, e^{-j\left(\frac{2\pi ux}{M} + \frac{2\pi vy}{N}\right)}$$

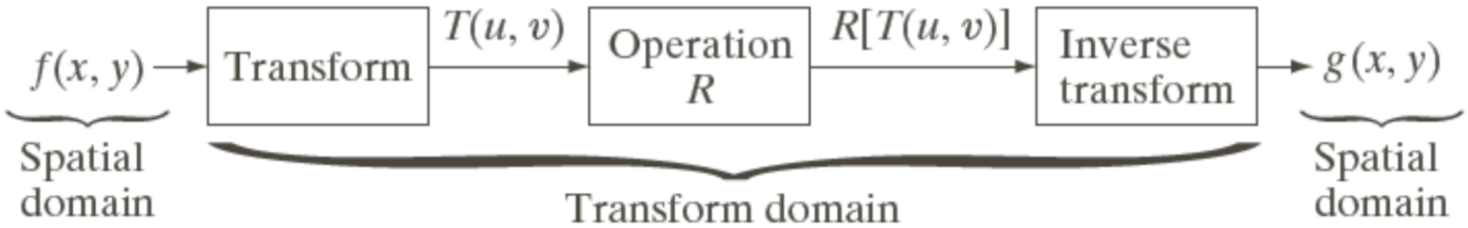
$$= A_M f A_N$$

- Inverse Transform

$$f = {A_M}^T F {A_N}^T \qquad AA^T = I$$

Image Transform

- The general approach for operating in linear transform domain



- The unitary transform satisfies

$$\sum_{x=0}^{M}\sum_{y=0}^{N}(\boldsymbol{f}[\boldsymbol{x},\boldsymbol{y}])^2 = \sum_{u=0}^{M}\sum_{v=0}^{N}(\boldsymbol{F}[\boldsymbol{u},\boldsymbol{v}])^2$$

  - i.e., the energy is preserved.

Good and Bad things about DFT

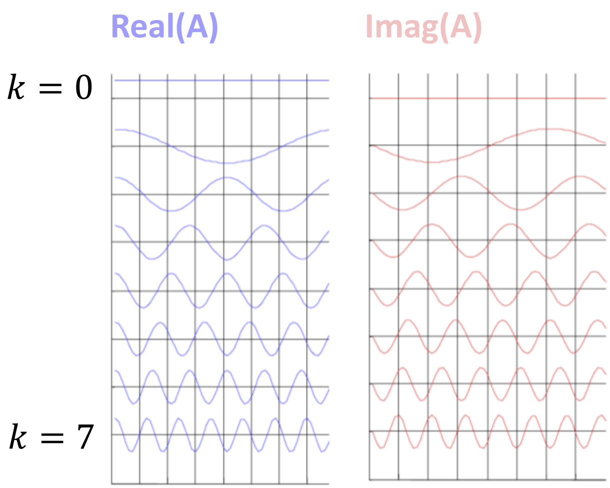
- **Positive:**
  - Energy is usually packed into low-frequency coefficients
  - Convolution property
  - Fast implementation
- **Negative:**
  - Transform is complex, even if image is real
  - The basis function span image height/width

DFT vs. DCT (Discrete Cosine Transform)

➤ **1D-DFT**

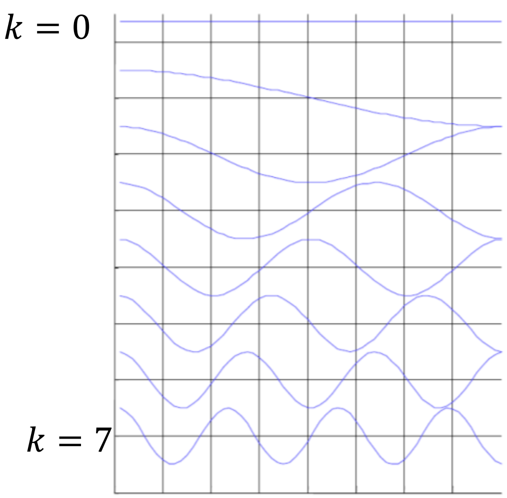
$$A[k,n] = e^{-j\frac{2\pi kn}{N}}$$

$$= cos\left(2\pi\frac{kn}{N}\right) + jsin\left(2\pi\frac{kn}{N}\right)$$



➤ **1D-DCT**

$$A[k,n] = \sqrt{\frac{2}{N}} cos \frac{\pi(2n+1)k}{2N}$$



What's the difference???

2D DCT

- Forward Transform

$$F(0,0) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

$$F(u,0) = \frac{\sqrt{2}}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{(2x+1)u\pi}{2N}$$

$$F(0,v) = \frac{\sqrt{2}}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{(2y+1)v\pi}{2N}$$

$$F(u,v) = \frac{2}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{(2x+1)u\pi}{2N}\cos\frac{(2y+1)v\pi}{2N}$$

2D IDCT

Inverse Transform

$$f(x,y) = \frac{1}{N} F(0,0)$$

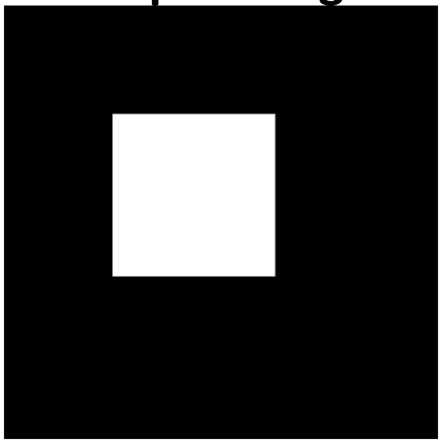
$$+ \frac{\sqrt{2}}{N} \sum_{u=1}^{N-1} F(u,0) \cos \frac{(2x+1)u\pi}{2N}$$

$$+ \frac{\sqrt{2}}{N} \sum_{v=1}^{N-1} F(0,v) \cos \frac{(2y+1)v\pi}{2N}$$

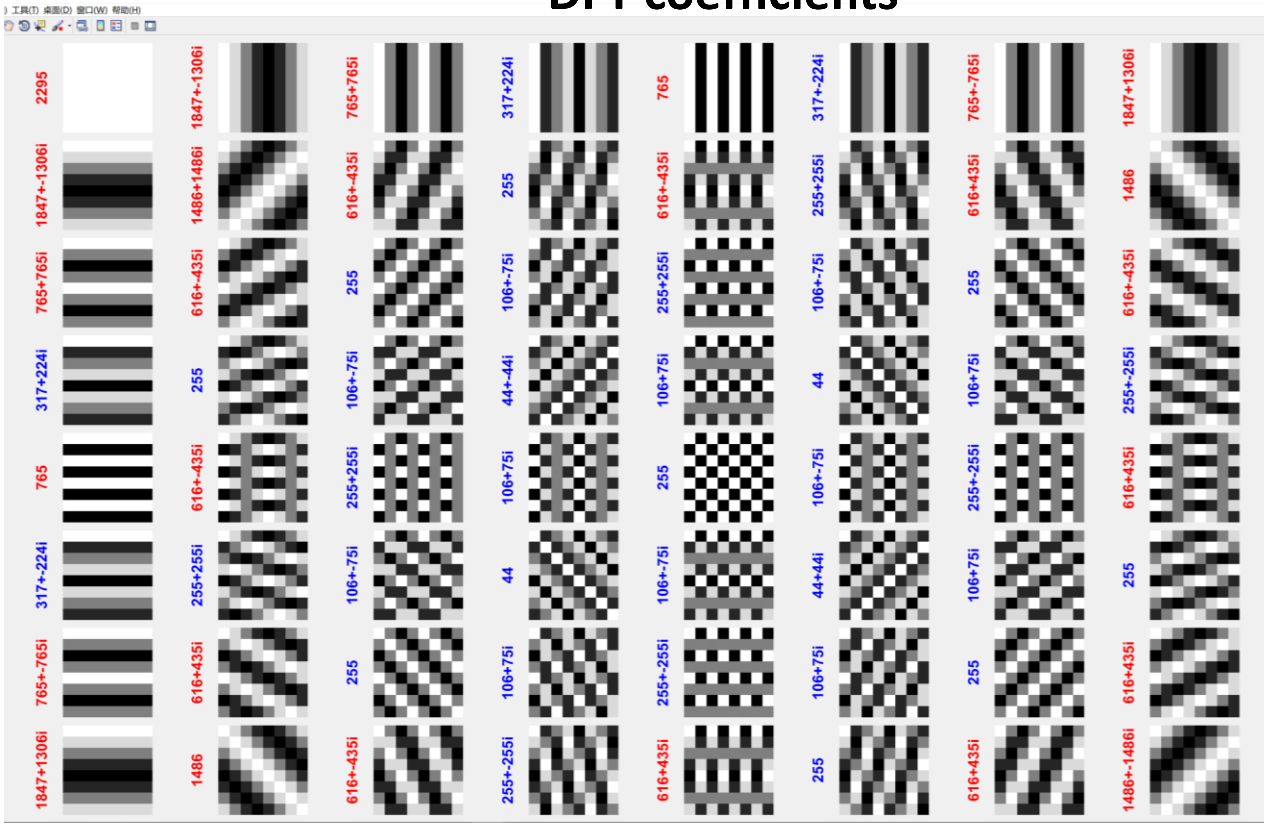
$$+ \frac{2}{N} \sum_{x=1}^{N-1} \sum_{y=1}^{N-1} F(u,v) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}$$

DFT example

Input image

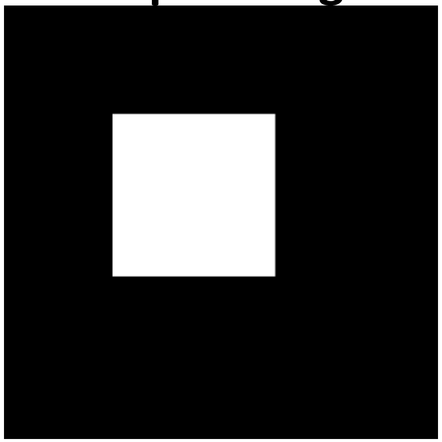


DFT coefficients

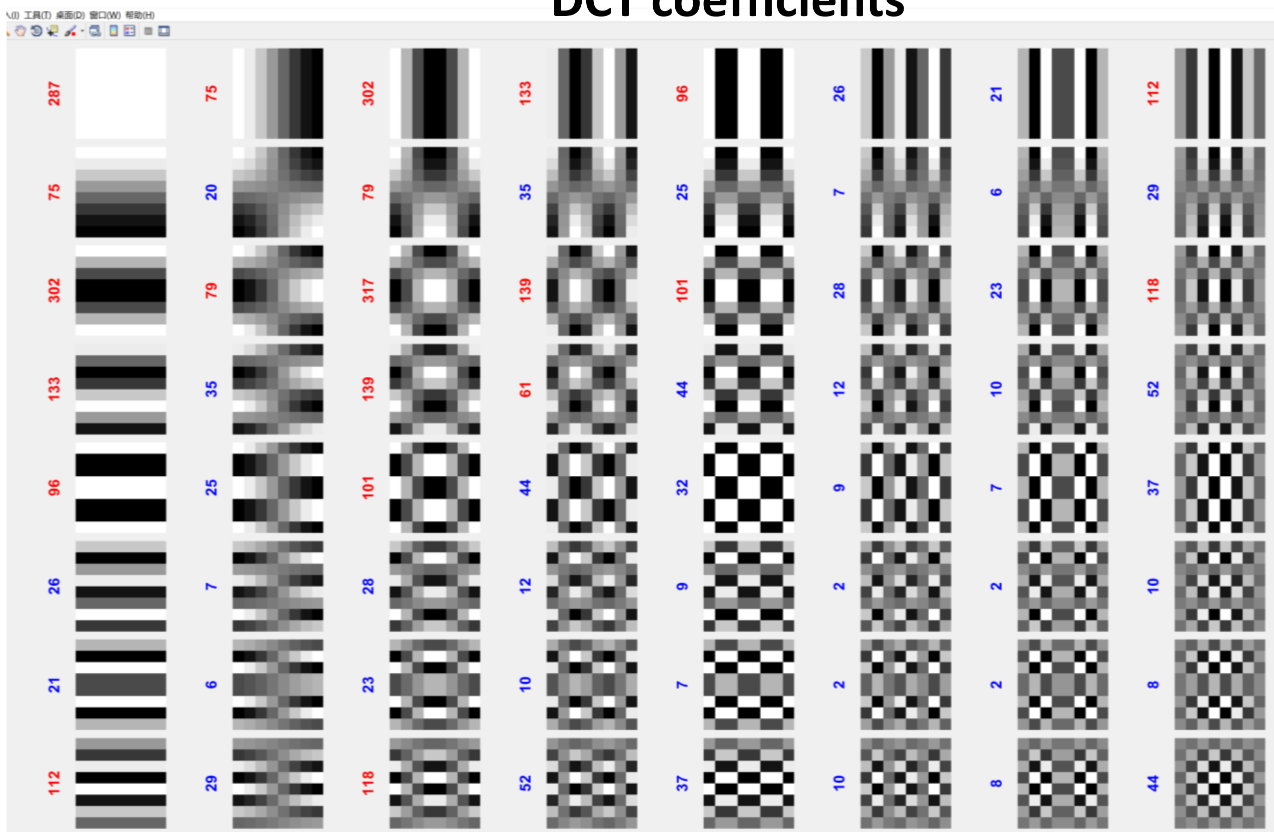


DCT example

Input image

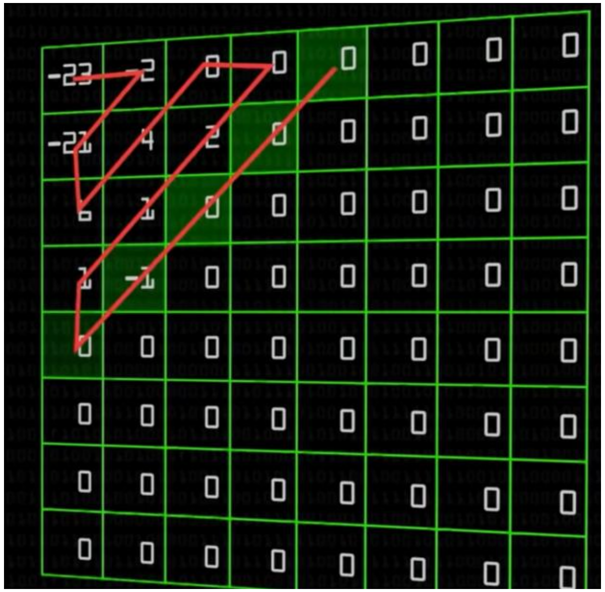


DCT coefficients



Good and bad things about DCT

- Positive
  - Transform is real, $C^{-1} = C^T$ (unitary transform).
  - Excelent energy compaction for nature images.
  - Fast transform.
  - JPEG algorithm.



Walsh Transform

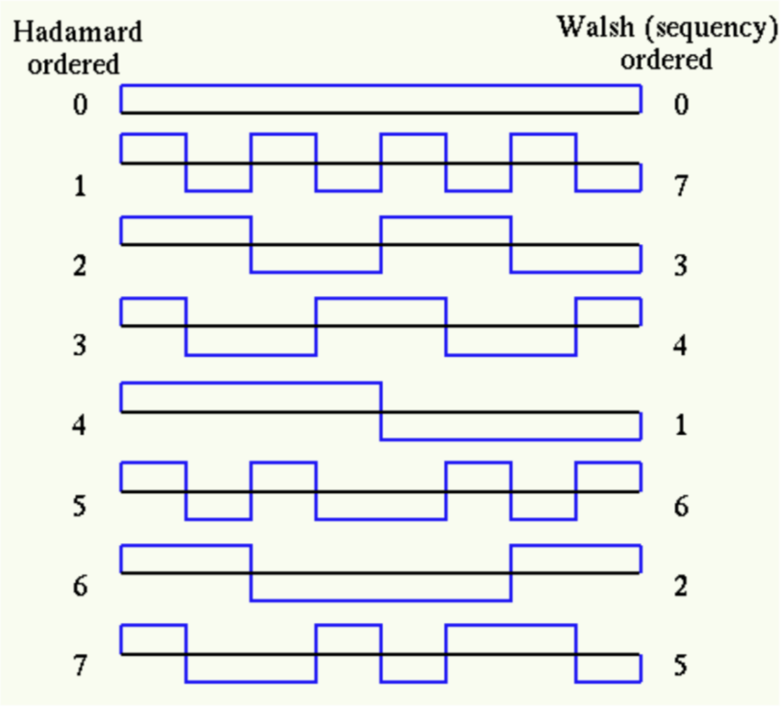
- Consist of ±1 arranged in a checkerboard pattern
- Transform:

$$W(i)=\frac{1}{N}\sum_{t=0}^{N-1} f(t)\cdot \mathrm{Wal}(i,t)$$

$$f(t)=\sum_{i=0}^{N-1} W(i)\cdot \mathrm{Wal}(i,t)$$

- Types of $\mathrm{Wal}(i,t)$
  - Walsh Ordering (沃尔什定序)
  - Paley Ordering （佩利定序）
  - Hadamard Matrix Ordering（哈达玛矩阵定序）

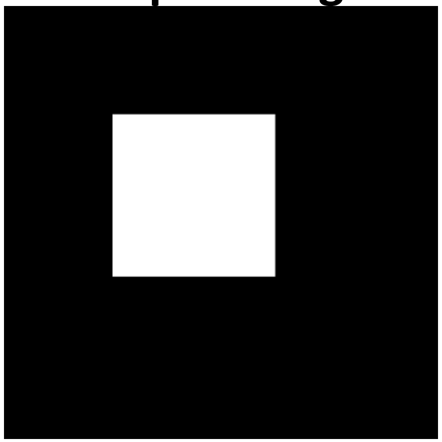
Hadamard Matrix Ordering



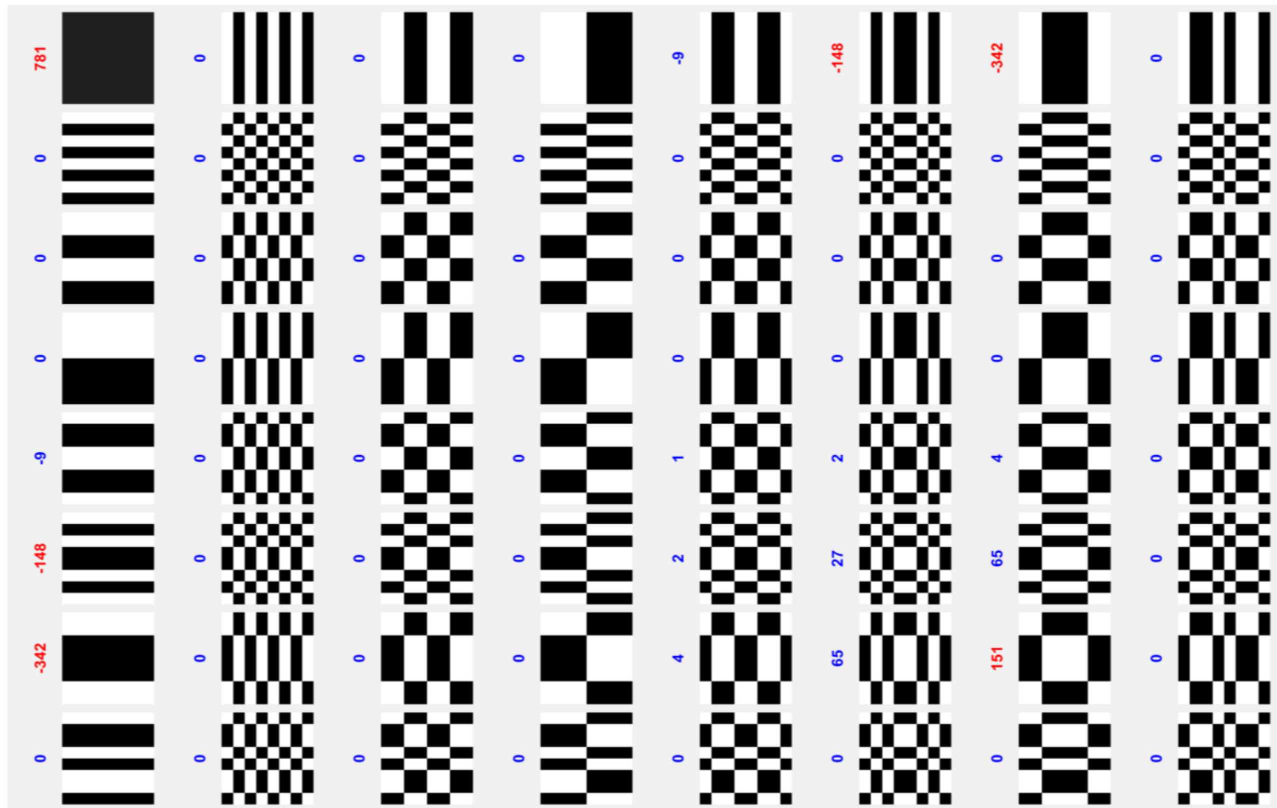
$$V_8 = \begin{pmatrix} W_4 & W_4 \\ W_4 & -W_4 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{pmatrix}$$

Hadarmad Transform

Input image

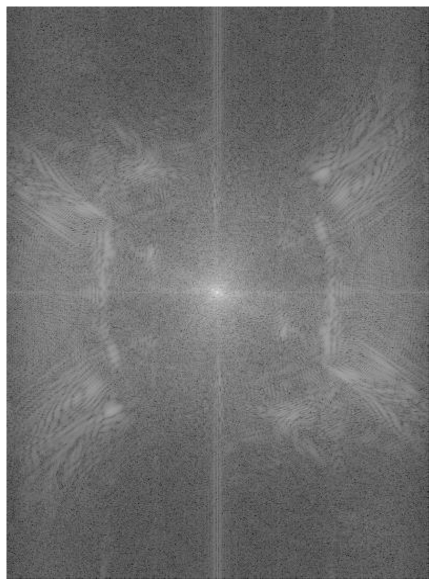


Hadarmad coefficients

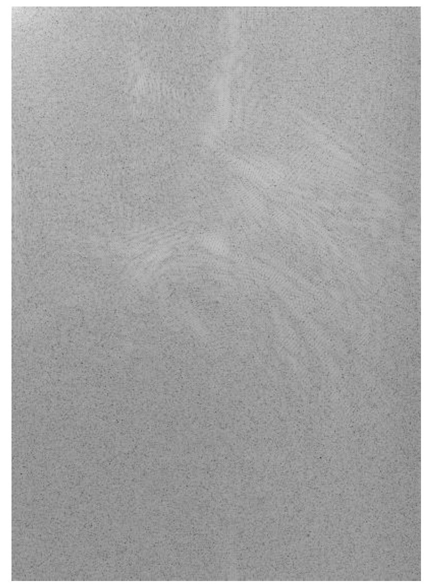


DFT, DCT, & Hadamard

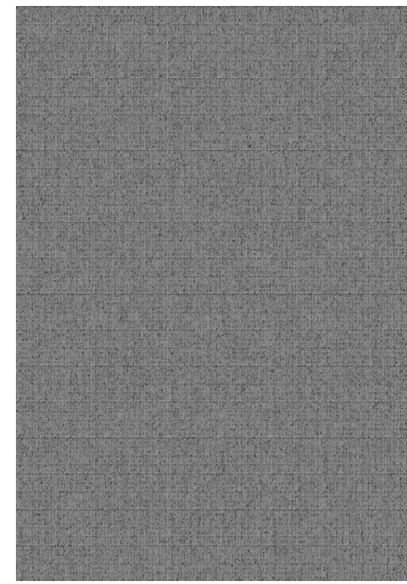
DFT



DCT



Hadamard



Any connection
between DFT and DCT?

Take home message

- The key idea for unitary transform is to find a proper basis for data decomposition.
- DCT provides better frequency consistency than DFT.
- Hadamard transform is able to present a simple image with simple coefficients. But can not keep energy compact for image full of details.

Wavelet transform Outline

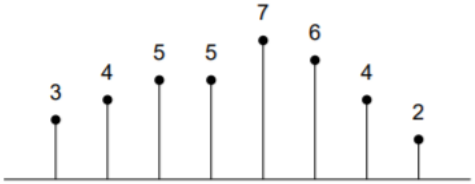
- Discrete Wavelet Transform (DWT)（小波变换）
  - An example for 1D-DWT
  - Generalization of 1D-DWT
  - 2D-DWT

Discrete Wavelet Transform (DWT)

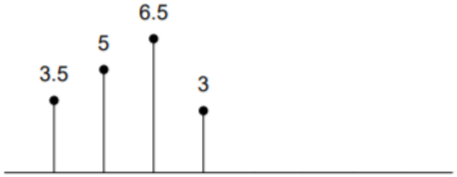
- Based on small waves called Wavelets-1) limited; 2) oscillation.
- Key idea: Translation & Scaling.
- Localized both time/space and frequency.
- Efficient for noise reduction and image compression.
- Two types of DWT one for image processing (easy invertible) and one for signal processing (invertible but computational expensive).

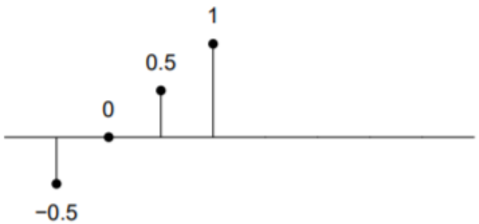
A simplest example

- ❑ **We can decompose an eight-point signal x(n):**



into two four-point signals:





$$c(n) = 0.5\,x(2n) + 0.5\,x(2n+1) \qquad d(n) = 0.5\,x(2n) - 0.5\,x(2n+1)$$

A simplest example

- The above process can be represented by a block diagram:

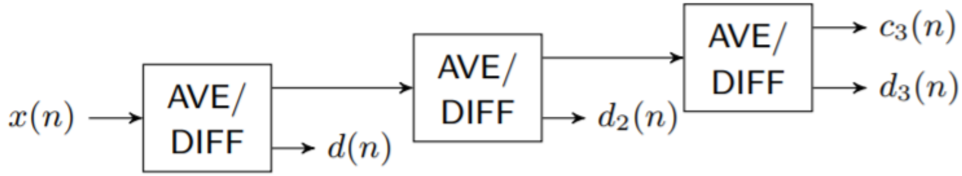
It is clear that this decomposition can be easily reversed:

$$y(2n) = c(n) + d(n)$$
$$y(2n+1) = c(n) - d(n)$$

Which is also represented by a block diagram:

A simplest example

- When we repeat the simple AVE/DIFF signal decomposition:



$$x(n) = [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8] = [3, 4, 5, 5, 7, 6, 4, 2]$$

Level 1

$$c_1 = \frac{1}{2}[x_1 + x_2, x_3 + x_4, x_5 + x_6, x_7 + x_8] \qquad d = d_1 = \frac{1}{2}[x_1 - x_2, x_3 - x_4, x_5 - x_6, x_7 - x_8]$$

Level 2

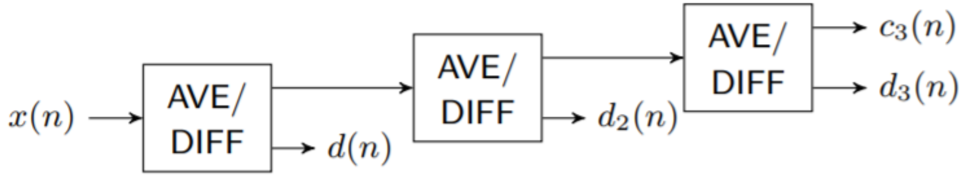
$$c_2 = \frac{1}{4}[x_1 + x_2 + x_3 + x_4, x_5 + x_6 + x_7 + x_8] \qquad d_2 = \frac{1}{4}[x_1 - x_3 + x_2 - x_4, x_5 - x_7 + x_6 - x_8]$$

Level 3

$$c_3 = \frac{1}{8}[x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8] \qquad d_3 = \frac{1}{8}[x_1 + x_2 + x_3 + x_4 - x_5 - x_6 - x_7 - x_8]$$

A simplest example

- When we repeat the simple AVE/DIFF signal decomposition:



The Haar wavelet representation of the eight-point signal x[n] is simply the set of four output signals produced by this three-level operation :

$$
\begin{aligned}
c_3 &= [4.5] \\
d_3 &= [-0.25] \\
d_2 &= [-0.75, \quad 1.75] \\
d &= [-0.5, \quad 0, \quad 0.5, \quad 1]
\end{aligned}
$$

$$
\mathbf{H}_8 = \frac{1}{\sqrt{8}} \begin{array}{cc}
\left[\begin{array}{cccccccc}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
\sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\
2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & -2
\end{array}\right] &
\begin{array}{c}
c_3 \\ d_3 \\ d_2(1) \\ d_2(2) \\ d(1) \\ d(2) \\ d(3) \\ d(4)
\end{array}
\end{array}
$$

上海科技大学
ShanghaiTech University

Haar Transform matrix

- When N=2 we have:

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

- When N=4 we have:

$$\mathbf{H}_4 = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

- When N=8 we have:

$$\mathbf{H}_8 = \frac{1}{\sqrt{8}}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix}$$

Haar Transform matrix

- The family of N Haar functions $h_u(x)$, $(u = 0, \ldots, N-1)$ are defined on the interval $0 \le x \le 1$. The shape of the specific function $h_u(x)$ of a given index u depends on two parameters p and q:

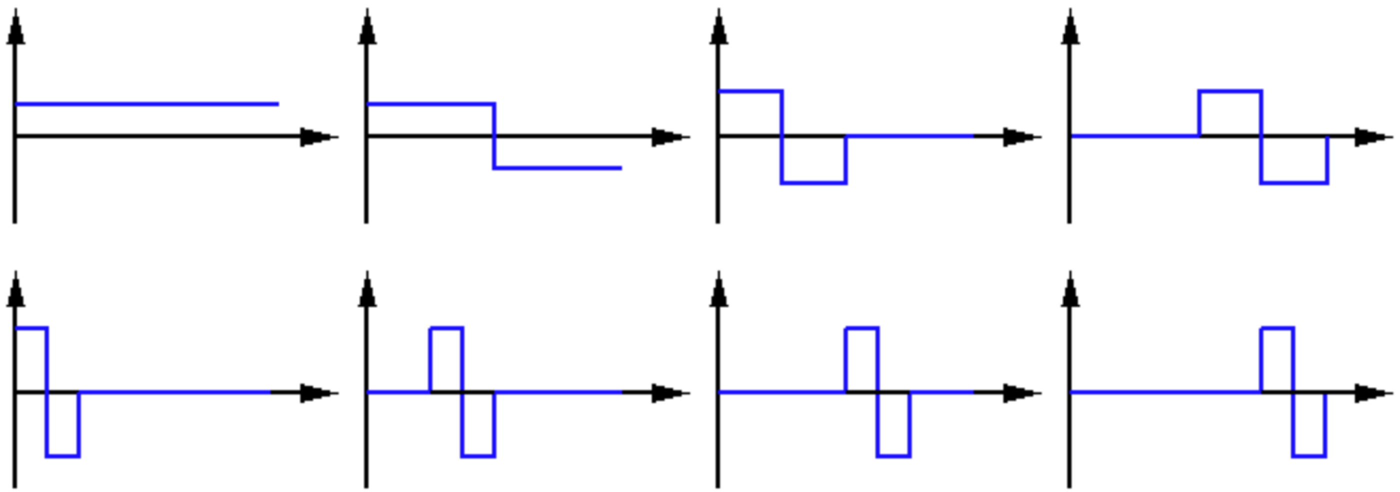
$$u = 2^p + q$$

u	p	q
1	0	0
2	1	0
3	1	1

- The Haar basis functions are defined by:

$$h_u(x) = \begin{cases} 1 & u = 0 \text{ and } 0 \le x < 1 \\ 2^{p/2} & u > 0 \text{ and } q/2^p \le x < (q+0.5)/2^p \\ -2^{p/2} & u > 0 \text{ and } (q+0.5)/2^p \le x < (q+1)/2^p \\ 0 & \text{otherwise} \end{cases}$$

Haar Transform matrix



$$\mathbf{H}_8 = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix}$$

Generalization of 1D-DWT

- **Discrete Wavelet Transform (DWT):**

$$W_\varphi(j_0,k)=\frac{1}{\sqrt{M}}\sum_n f(n)\,\varphi_{j_0,k}(n)$$

$$W_\psi(j,k)=\frac{1}{\sqrt{M}}\sum_n f(n)\,\psi_{j,k}(n)\quad j\geq j_0$$

- **Inverse Discrete Wavelet Transform (IDWT):**

$$f(n)=\frac{1}{\sqrt{M}}\sum_k W_\varphi(j_0,k)\,\varphi_{j_0,k}(n)+\frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_k W_\psi(j,k)\,\psi_{j,k}(n)$$

Where

$\varphi_{j_0,k}(n)$: scaling function (尺度函数)

$\psi_{j,k}(n)$: Wavelet (小波)

$W_\varphi(j_0,k)$: Approximation coefficients (近似系数)

$W_\psi(j,k)$: detail coefficients (细节系数)

2D-DWT

- **Define 2D wavelet function: Directionally sensitive wavelet**

$$\psi^H(x,y)=\psi(x)\varphi(y) \qquad \psi^V(x,y)=\varphi(x)\psi(y) \qquad \psi^D(x,y)=\psi(x)\psi(y)$$

- **2D-DWT**

$$W_\varphi(j_0,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\,\varphi_{j_0,m,n}(x,y)$$

$$W_\psi(j,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\,\psi^i_{j,m,n}(x,y) \qquad i=\{H,V,D\}$$

- **2D-IDWT**

$$f(x,y)=\frac{1}{\sqrt{MN}}\sum_m\sum_n W_\varphi(j_0,m,n)\,\varphi_{j_0,m,n}(x,y)$$

$$+\frac{1}{\sqrt{MN}}\sum_{i=\{H,V,D\}}\sum_{j=j_0}^{\infty}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}W_\psi(j,m,n)\,\psi^i_{j,m,n}(x,y)$$

Haar Transform matrix

Input: image size 8X8 I_{in}

Generate a Haar matrix of 8X8 as shown right
Then clip it into 4 part:

$$H_{L1}(dim = 4 * 8); H_{L2}(dim = 2 * 8);$$
$$H_{L3}(dim = 1 * 8); L_{L3}(dim = 1 * 8);.$$

$$\mathbf{H}_8 = \frac{1}{\sqrt{8}} \left[\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \hline \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ \hline 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{array}\right] \begin{array}{c} \\ H_{L3} \\ H_{L2} \\ \\ \\ \\ H_{L1} \\ \\ \end{array}$$

For computing ***level 1*** components:

LL_1 is down-sample of I_{in} on both X and Y direction	$HL_1 = H_{L1} * I_{in}$ + downsample on Y direction
$LH_1 = I_{in} * H_{L1}$ + downsample on X direction	$HH_1 = H_{L1} * I_{in} * H_{L1}$

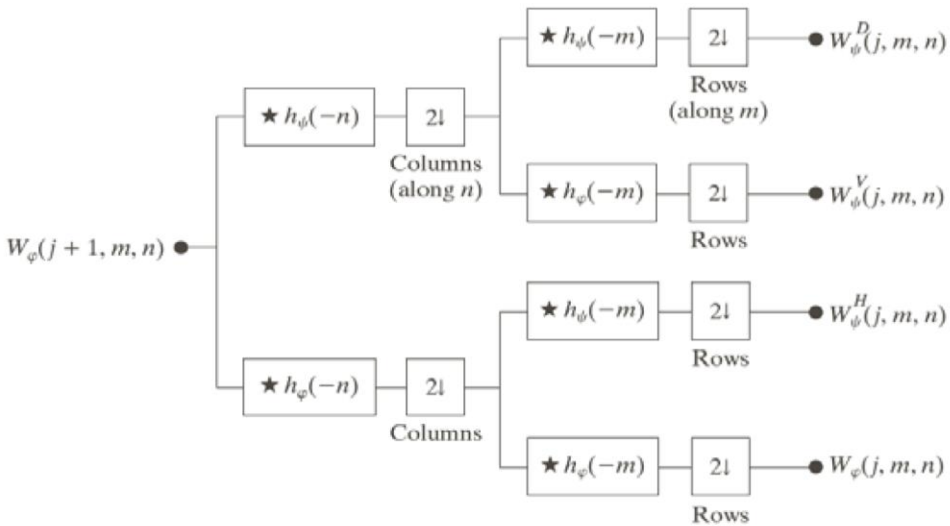
For computing ***level 2*** components:

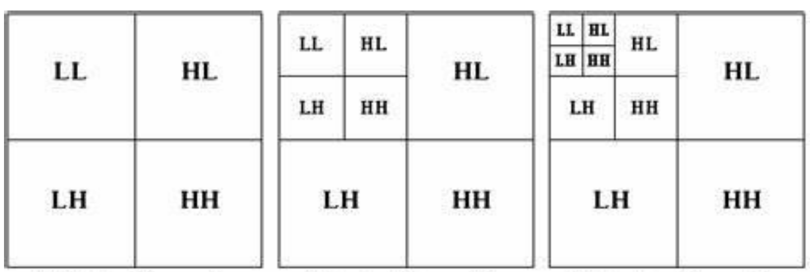
LL_2 is down-sample of LL_1 on both X and Y direction	$HL_2 = H_{L2} * I_{in}$ + downsample **twice** on Y direction
$LH_2 = I_{in} * H_{L2}$ + downsample **twice** on X direction	$HH_2 = H_{L2} * I_{in} * H_{L2}$

For computing ***level 3*** components:

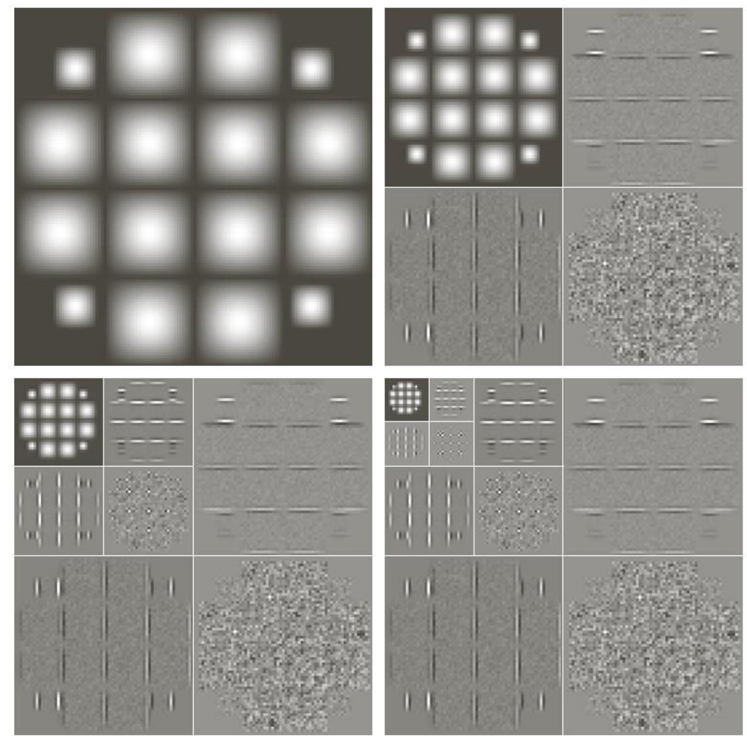
LL_3 is down-sample of LL_2 on both X and Y direction	$HL_3 = H_{L3} * I_{in}$ + downsample **3 times** on Y direction
$LH_3 = I_{in} * H_{L3}$ + downsample **3 times** on X direction	$HH_3 = H_{L3} * I_{in} * H_{L3}$

2D-DWT

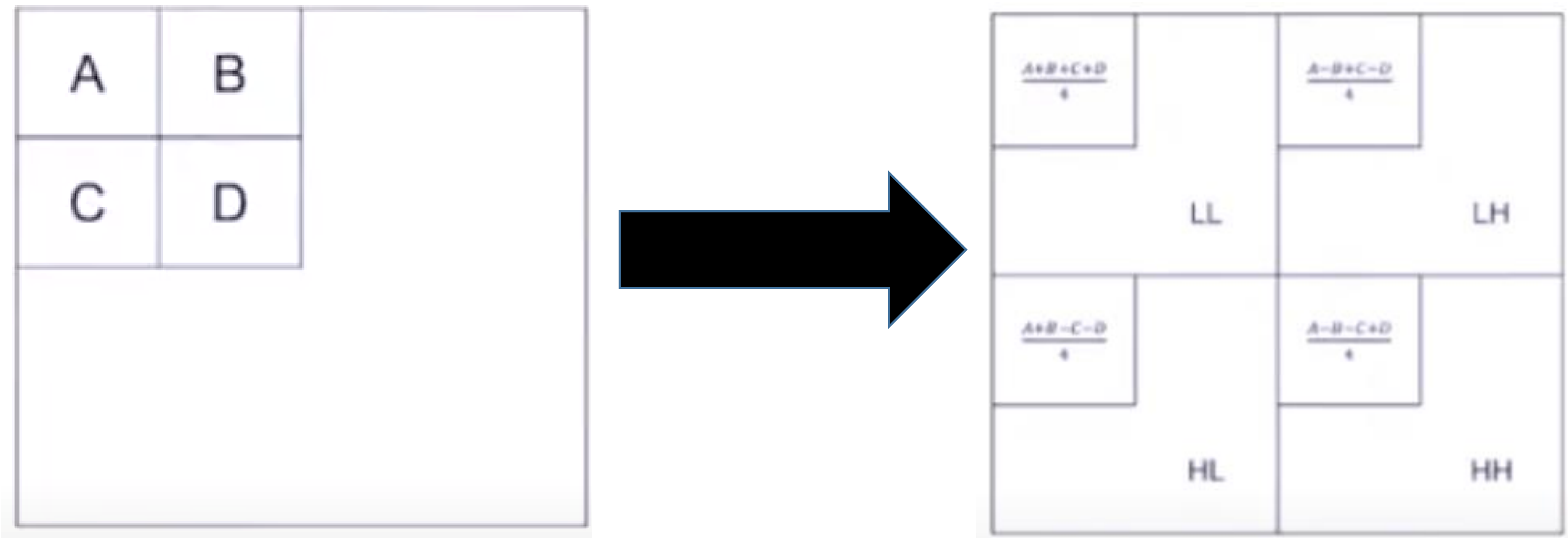




(a) Single Level Decomposition (b) Two Level Decomposition (c) Three Level Decomposition



2D Haar Transform



2D Haar Transform

$$A = \begin{pmatrix} 88 & 88 & 89 & 90 & 92 & 94 & 96 & 97 \\ 90 & 90 & 91 & 92 & 93 & 95 & 97 & 97 \\ 92 & 92 & 93 & 94 & 95 & 96 & 97 & 97 \\ 93 & 93 & 94 & 95 & 96 & 96 & 96 & 96 \\ 92 & 93 & 95 & 96 & 96 & 96 & 96 & 95 \\ 92 & 94 & 96 & 98 & 99 & 99 & 98 & 97 \\ 94 & 96 & 99 & 101 & 103 & 103 & 102 & 101 \\ 95 & 97 & 101 & 104 & 106 & 106 & 105 & 105 \end{pmatrix}$$

3 level Haar Transform for the first row

$$r_1 = (88 \quad 88 \quad 89 \quad 90 \quad 92 \quad 94 \quad 96 \quad 97)$$

Group r_1 in pair $[88, 88], [89, 90], [92, 94], [96, 97]$

$$r_1 h_1 = (\underbrace{88 \quad 89.5 \quad 93 \quad 96.5}_{\text{Approximation coefficients}} \quad \underbrace{0 \quad -0.5 \quad -1 \quad -0.5}_{\text{Detail coefficients}})$$

Group the first 4 columns in pair $[88, 89.5], [93, 96.5]$

$$r_1 h_1 h_2 = (88.75 \quad 94.75 \quad -0.75 \quad -1.75 \quad 0 \quad -0.5 \quad -1 \quad -0.5)$$

Group the first 2 columns in pair $[88, 94.75]$

$$r_1 h_1 h_2 h_3 = (91.75 \quad -3 \quad -0.75 \quad -1.75 \quad 0 \quad -0.5 \quad -1 \quad -0.5)$$

2D Haar Transform

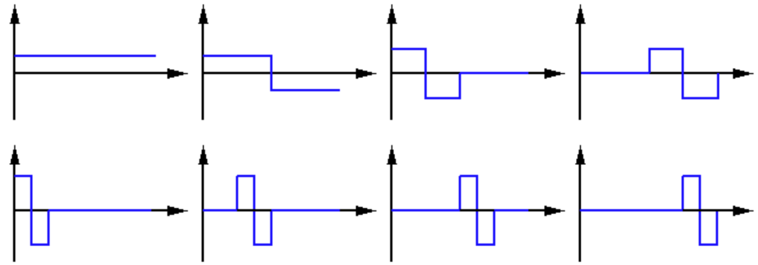
Repeat the same processing for all the columns and for the rows of the resulting matrix, we get

$$\begin{pmatrix} 96 & -2.0312 & -1.5312 & -0.2188 & -0.4375 & -0.75 & -0.3125 & 0.125 \\ -2.4375 & -0.0312 & 0.7812 & -0.7812 & 0.4375 & 0.25 & -0.3125 & -0.25 \\ -1.125 & -0.625 & 0 & -0.625 & 0 & 0 & -0.375 & -0.125 \\ -2.6875 & 0.75 & 0.5625 & -0.0625 & 0.125 & 0.25 & 0 & 0.125 \\ -0.6875 & -0.3125 & 0 & -0.125 & 0 & 0 & 0 & -0.25 \\ -0.1875 & -0.3125 & 0 & -0.375 & 0 & 0 & -0.25 & 0 \\ -0.875 & 0.375 & 0.25 & -0.25 & 0.25 & 0.25 & 0 & 0 \\ -1.25 & 0.375 & 0.375 & 0.125 & 0 & 0.25 & 0 & 0.25 \end{pmatrix}$$

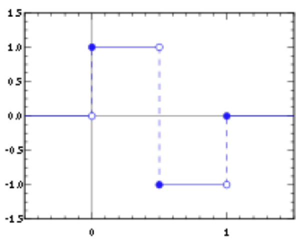
Mother Wavelet (母小波)

➢ **Mother Wavelet should satisfy:**

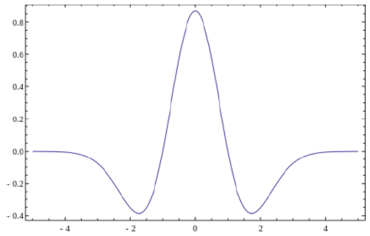
- $\int_{-\infty}^{\infty} |\psi(t)|^2 dt = 1$
- $\int_{-\infty}^{\infty} |\psi(t)| dt < \infty$
- $\int_{-\infty}^{\infty} \psi(t) dt = 0$



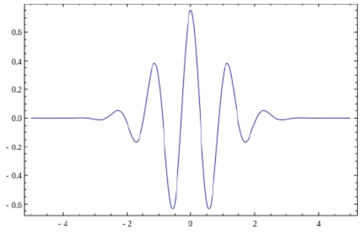
Haar



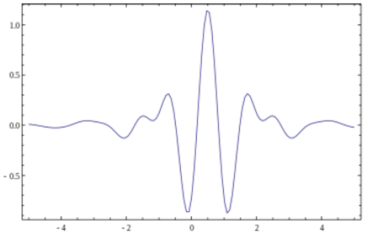
Mexican Hat



Morlet



Meyer



Take home message

- Based on small waves called Wavelets-1) limited; 2) oscillation.
- Key idea: Translation & Scaling.
- Localized both time/space and frequency.
- Efficient for noise reduction and image compression.
- JPEG2000, FBI finger printing databased.