Digital Image Processing

Chapter 4
Filtering in the Frequency Domain

Yasin Fakhar

Complex numbers

A complex number, C, is defined as $R + jI$ where R and I are real numbers and $j = \sqrt{-1}$

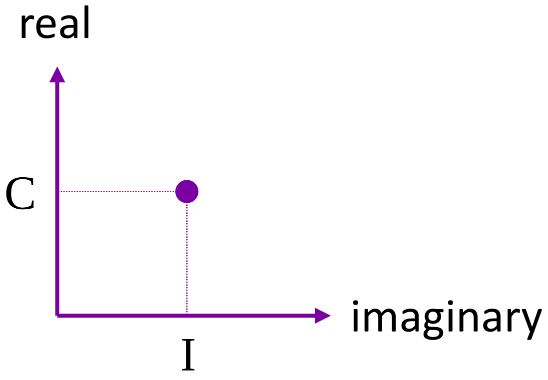
Real part Imaginary part

The conjugate of a complex number C, denoted C*, is defined as $C^* = R - jI$

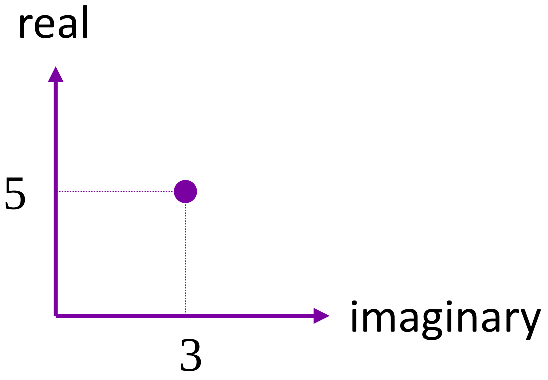
Complex numbers

Complex numbers can be viewed geometrically as points on a plane (called the complex plane)

$C + jI$



Example $5 + 3j$



Complex numbers

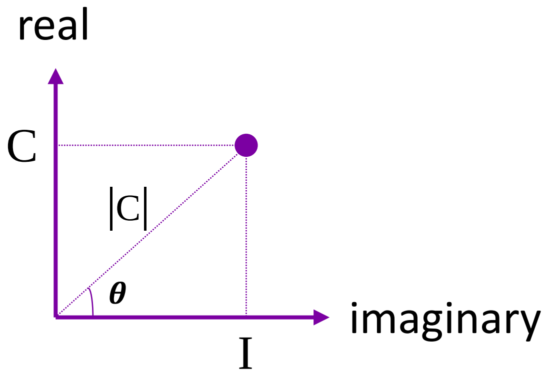
Sometimes it is useful to represent complex numbers in polar coordinates

$$C = |C|(\cos\theta + j\sin\theta)$$

where

$$|C| = \sqrt{R^2 + I^2}$$

$$\theta = \arctan(\frac{I}{R})$$



The arctan function returns angles in the range [-∏/2 , ∏/2]. But, because I and R can be positive and negative independently, we need to be able to obtain angles in the full range [-∏, ∏] We do this by keeping track of the sign of I and R when computing θ.

Euler's formula

$e^{j\theta} = \cos\theta + j\sin\theta$ where $e = 2.71828\ldots$

gives the following familiar representation of complex numbers in polar coordinates

$C = |C|\, e^{j\theta}$

For example, the polar representation of the complex number 1+2j is $\sqrt{5}e^{63.4j}$, where θ = 63.4° or 1.1 radians.

Complex number representation

We can show a complex number in three format :

$$C = \mathrm{R} + \mathrm{jI}$$

$$C = |C|(\cos\theta + j\sin\theta)$$

$$C = |C|\, e^{j\theta}$$

We can look at an image from another aspect

$$\begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} = 1 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + 0 \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + 2 \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + 1 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} = 1 \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} + 0.5 \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} - 0.5 \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} + 0 \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Basis

Note : if basis are orthonormal (orthogonal + normal) then the computation of coefficients would be very much cheaper

Dot product = 0

$||.|| = 1$

How to compute coefficients

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} = 1.5 \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 0.5 \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

f(x) s(x,u=0) s(x,u=1)

$$\begin{bmatrix} 1.5 \\ -0.5 \end{bmatrix}$$

T(u)

Basis : s(x,u) for all u : orthogonal, why?
Coefficients : T(u)

$$T(u=i) = \frac{<f(x), s(x,u=i)>}{||s(x,u=i)||2}$$

If basis are orthonormal :

$$f(x) = \sum_{u=0}^{N-1} T(u) s(x,u)$$

$$T(u) = \sum_{x=0}^{N-1} f(x) r(x,u)$$

$$T(u) = \langle f(x), s(x,u) \rangle$$

$$r(x,u) = s^*(x,u)$$

If s() is not complex then r = s

$$\underset{f(x,y)}{\begin{bmatrix}1 & 0\\ 2 & 1\end{bmatrix}} = 1\underset{s(x,y,u=0,v=0)}{\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}} + 0.5\underset{s(x,y,u=1,v=0)}{\begin{bmatrix}1 & -1\\ 1 & -1\end{bmatrix}} - 0.5\underset{s(x,y,u=0,v=1)}{\begin{bmatrix}1 & 1\\ -1 & -1\end{bmatrix}} + 0\underset{s(x,y,u=1,v=1)}{\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}}$$

orthogonal

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} T(u,v)s(x,y,u,v)$$

$$T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r(x,y,u,v)$$

T(u,v)

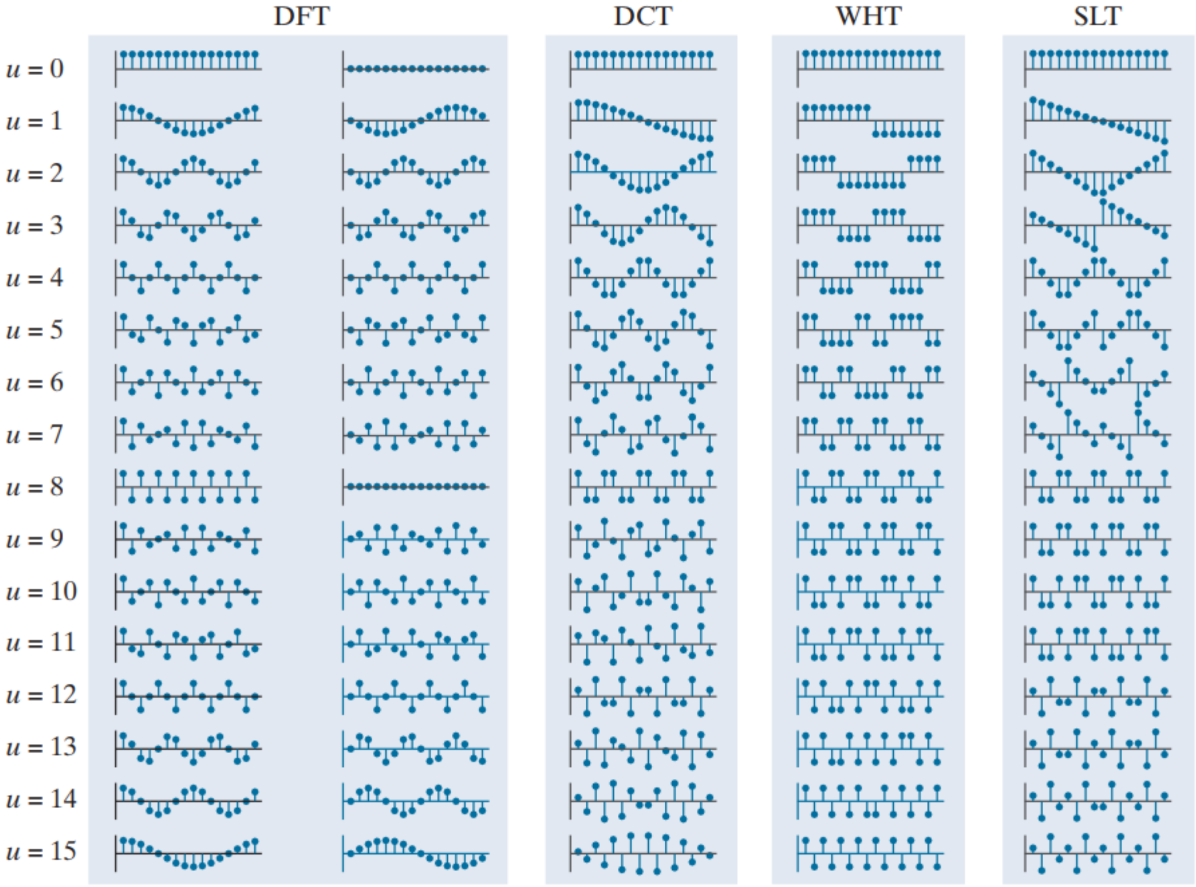
1	0.5
-0.5	0

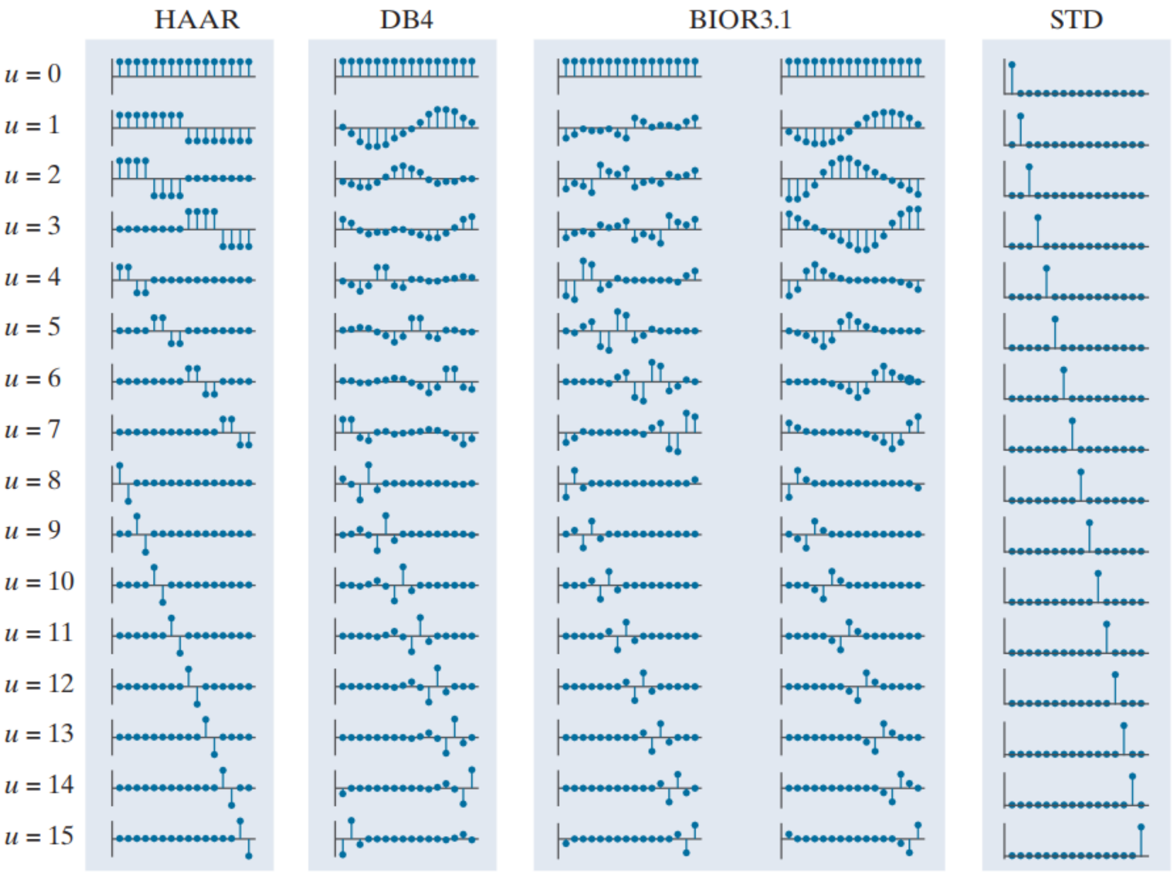
NOTE

Basis are not normal, so we need to make them normalize
Divide by $||s||^2$

How to choose basis

All of the transformations must be the inverse





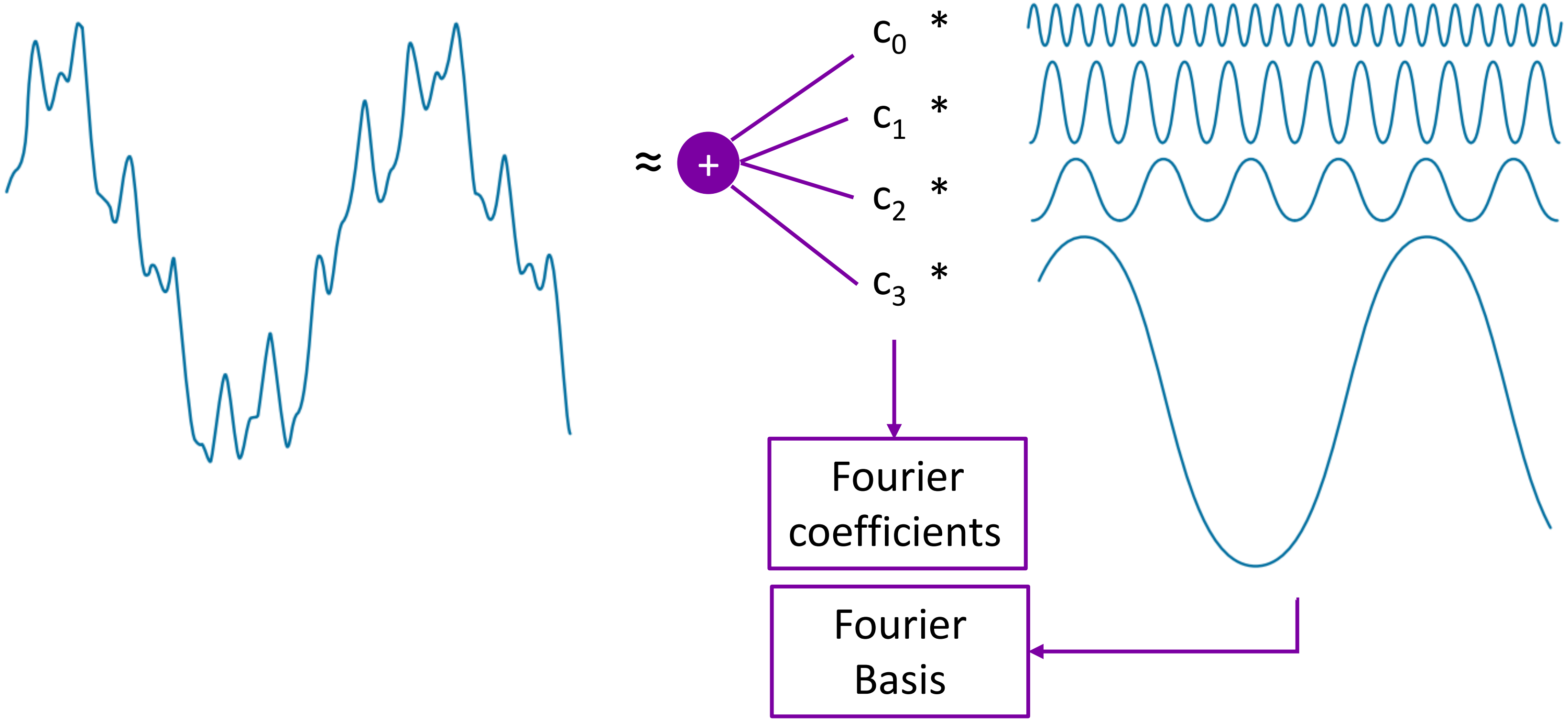
Spatial domain

Fourier serries and transform

Fourier's contribution in this field states that any periodic function can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient (we now call this sum a Fourier series)

It does not matter how complicated the function is; if it is periodic and satisfies some mild mathematical conditions, it can be represented by such a sum

Functions that are not periodic (but whose area under the curve is finite) can be expressed as the integral of sines and/or cosines multiplied by a weighting function. The formulation in this case is the Fourier transform


≈
+
c_0 *
c_1 *
c_2 *
c_3 *
Fourier coefficients
Fourier Basis

Symmetric and separable basis

Separable : $r(x,y,u,v) = r_1(x,u)\ r_2(y,v)$

Simplicity and fast in computation

Symmetric and separable: $r(x,y,u,v) = r_1(x,u)\ r_1(y,v)$

e.g : Fourier transform $r(x,y,u,v) = e^{-j2\pi(\frac{ux}{N}+\frac{vy}{N})}$

$$= e^{-j2\pi(\frac{ux}{N})}\ e^{-j2\pi(\frac{vy}{N})} \qquad \Rightarrow r_1(x,u) = e^{-j2\pi(\frac{ux}{N})}$$

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} T(u,v)s(x,y,u,v)$$

$$T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r(x,y,u,v)$$

$$T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r_1(x,u)r_2(y,v)$$

$$T(u,v) = \sum_{x=0}^{M-1} r_1(x,u)\sum_{y=0}^{N-1} f(x,y)r_2(y,v)$$

Fourier in x dimension then in y dimension or vise versa

Example of Fourier transform

compute the Fourier transform of below 2D signal. Which one are zero? Why?

2	2	2	2
1	1	1	1

$$\mathrm{T}(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r(x,y,u,v) \;, M = 2, N = 4$$

$$r(x,y,u,v) = e^{-j2\pi\left(\frac{ux}{N}+\frac{vy}{N}\right)}$$

$$\mathrm{T}(0,0) = \sum_{x=0}^{1}\sum_{y=0}^{3} f(x,y)r(x,y,0,0) = \sum_{x=0}^{1}\sum_{y=0}^{3} f(x,y)e^{0} = 12$$

$$\mathrm{T}(0,1) = \sum_{x=0}^{1}\sum_{y=0}^{3} f(x,y)r(x,y,0,1) = \sum_{x=0}^{1}\sum_{y=0}^{3} f(x,y)e^{-j2\pi\left(\frac{y}{4}\right)} =$$

$$2 \times (1-j-1+j) + 1 \times (1-j-1+j) = 0$$

```python
import numpy as np

signal = np.array([[2,2,2,2,],
                   [1,1,1,1,]])

res = np.fft.fft2(signal)
print(res)
```

```
[[12.+0.j  0.+0.j  0.+0.j  0.+0.j]
 [ 4.+0.j  0.+0.j  0.+0.j  0.+0.j]]
```

Basis Image

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} T(u,v)s(x,y,u,v)$$

$$\mathbf{F} = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} T(u,v)\boldsymbol{S}(u,v)$$

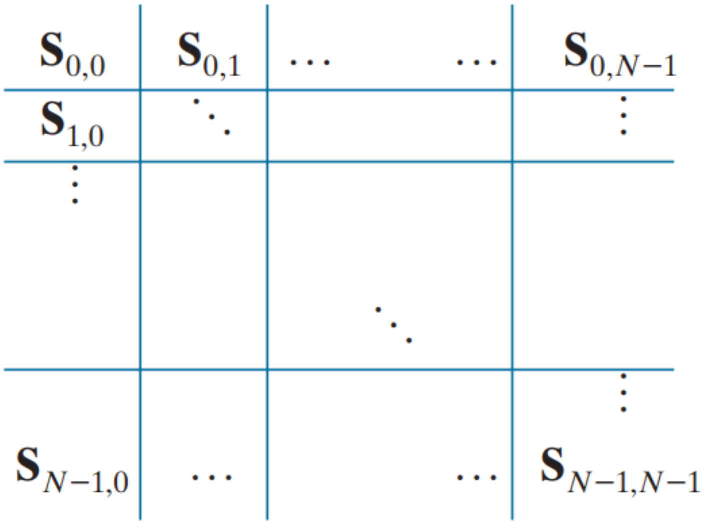
F		**S**(u=0,v=0)		**S**(u=1,v=0)		**S**(u=0,v=1)		**S**(u=1,v=1)
f(x,y)		s(x,y,u=0,v=0)		s(x,y,u=1,v=0)		s(x,y,u=0,v=1)		s(x,y,u=1,v=1)

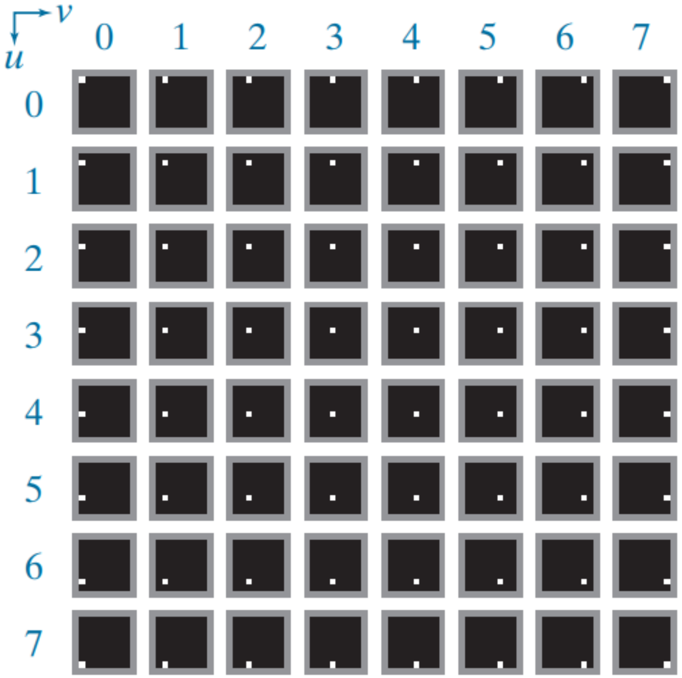
$$\begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} = 1 \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} + 0.5 \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} - 0.5 \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} + 0 \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Basis Image

$$\begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} = 1\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + 0\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + 2\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + 1\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{F} = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} T(u,v)\boldsymbol{S}(u,v)$$



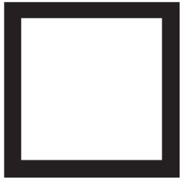


Yasin Fakhar

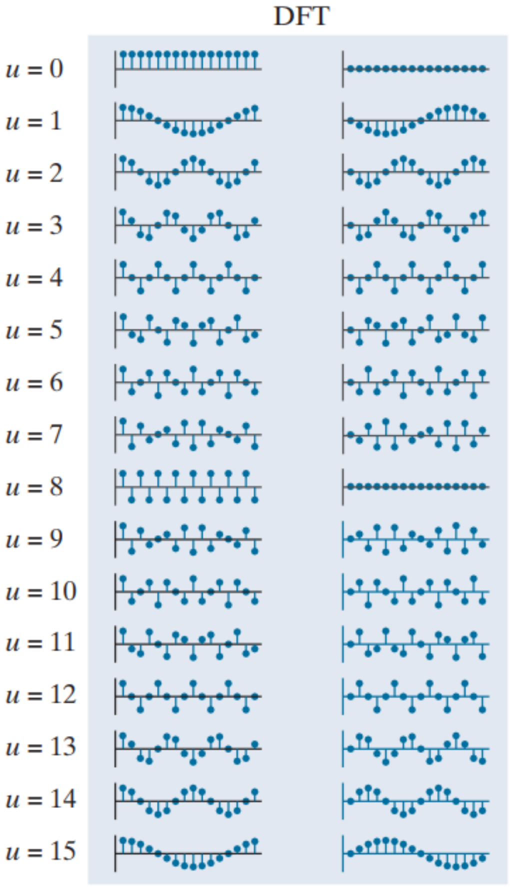
Basis Image of Fourier

$$s(x,y,u,v) = e^{+j2\pi(\frac{ux}{N}+\frac{vy}{N})} \begin{cases} \text{Real} : \cos(2\pi(\frac{ux}{N} + \frac{vy}{N})) \\ \text{Imaginary} : i\sin(2\pi(\frac{ux}{N} + \frac{vy}{N})) \end{cases}$$

Real of s(x,y,u=0,v=0)

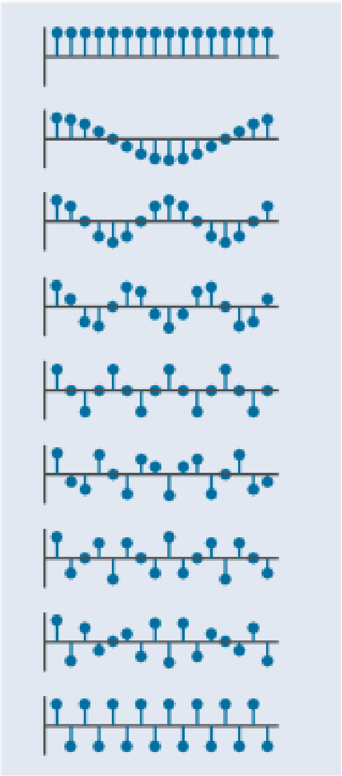


Real of s(x,y,u=0,v=1)



1-D

2-D (v = 0)

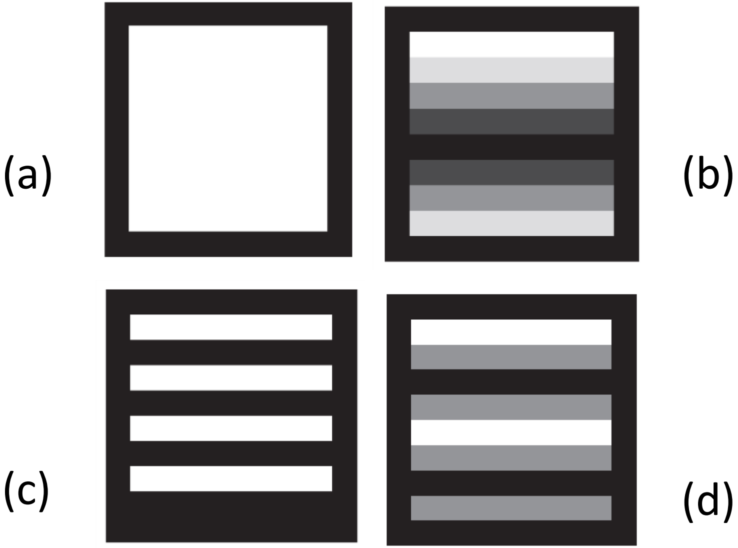


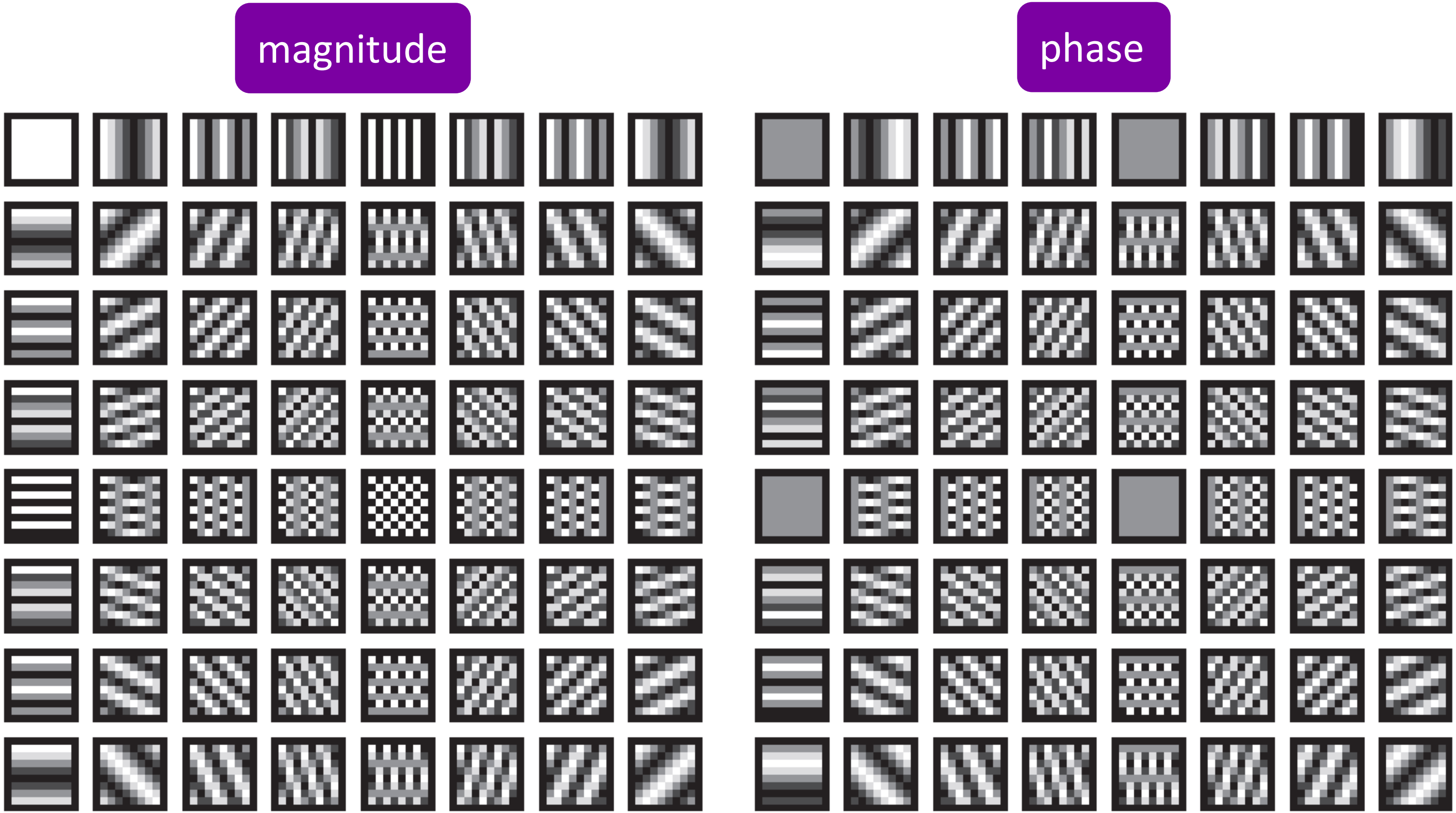
(a)

(b)

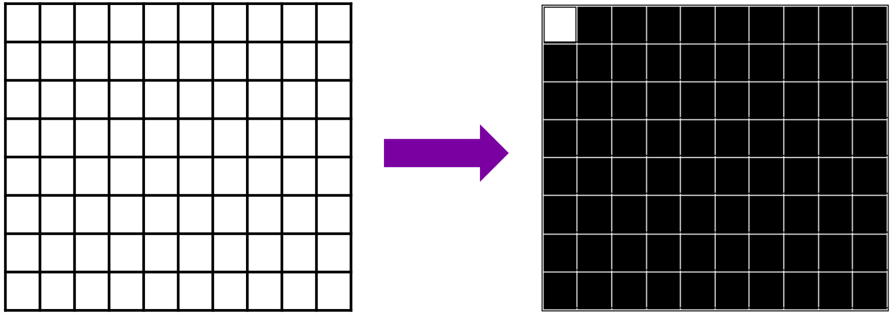
(d)

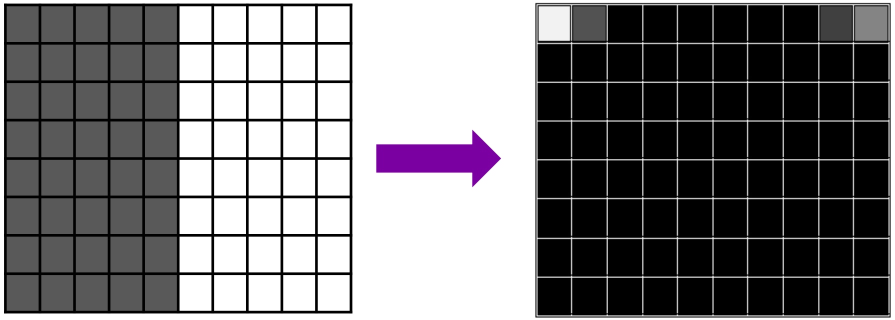
(c)

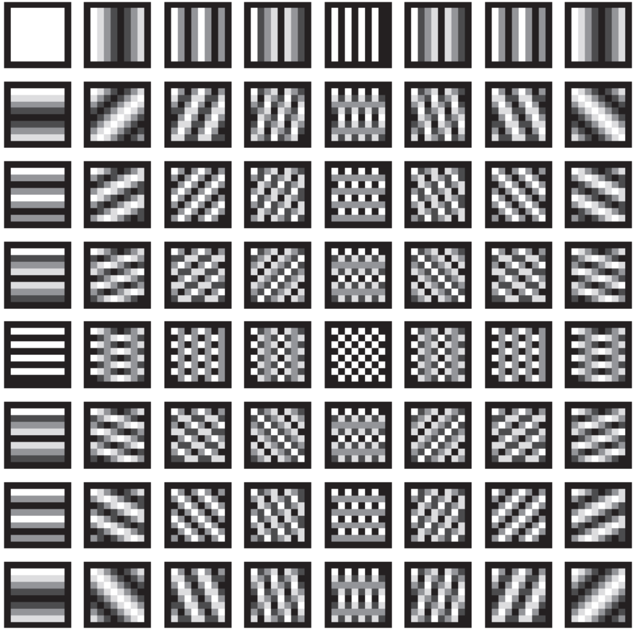



magnitude
phase

- So we can show every images by a linear combination of basis image
- If we change one of the transform coefficient and then come back to the original space (spatial) whole image will change.
- The greatest frequency in image occurs at $v = N/2$ and $u = N/2$ where phase will be 0 and the magnitude will be like a chess board
- The lowest frequency would be zero



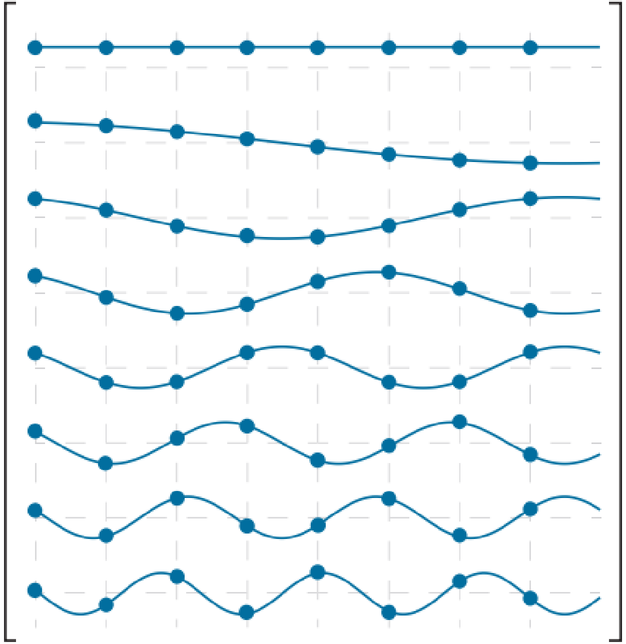


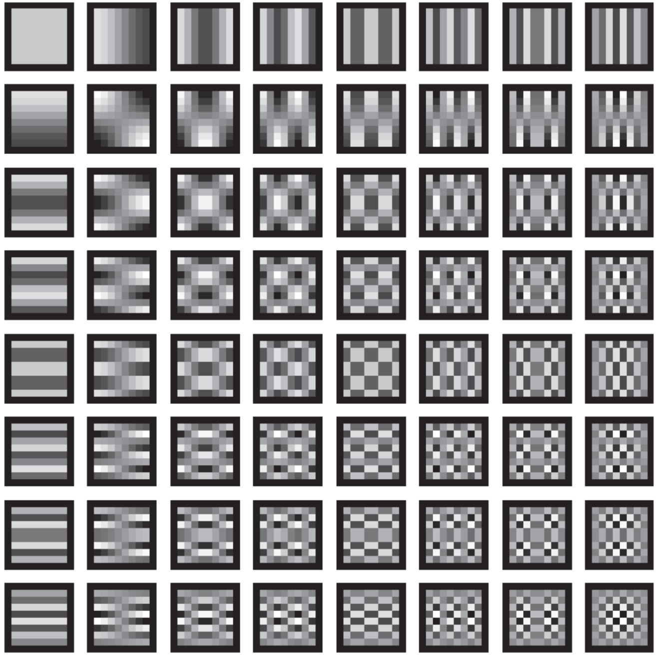


DCT transform

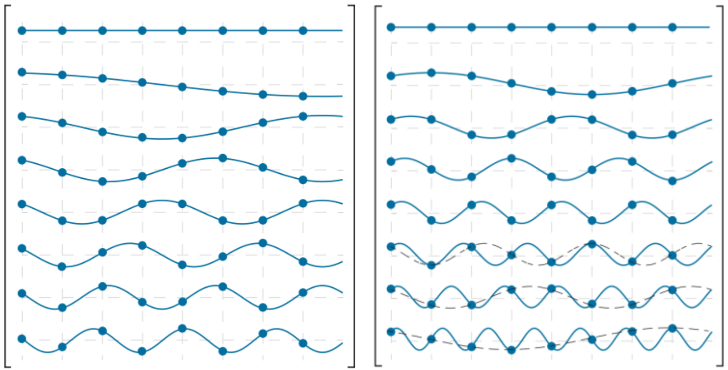
$$s(x,u) = \alpha(u)\cos\left(\frac{(2x+1)u\pi}{2N}\right) \qquad s(x,y,u,v) = \alpha(u)\alpha(v)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } u = 0 \\ \sqrt{\frac{2}{N}} & \text{for } u = 1, 2, \ldots, N-1 \end{cases}$$



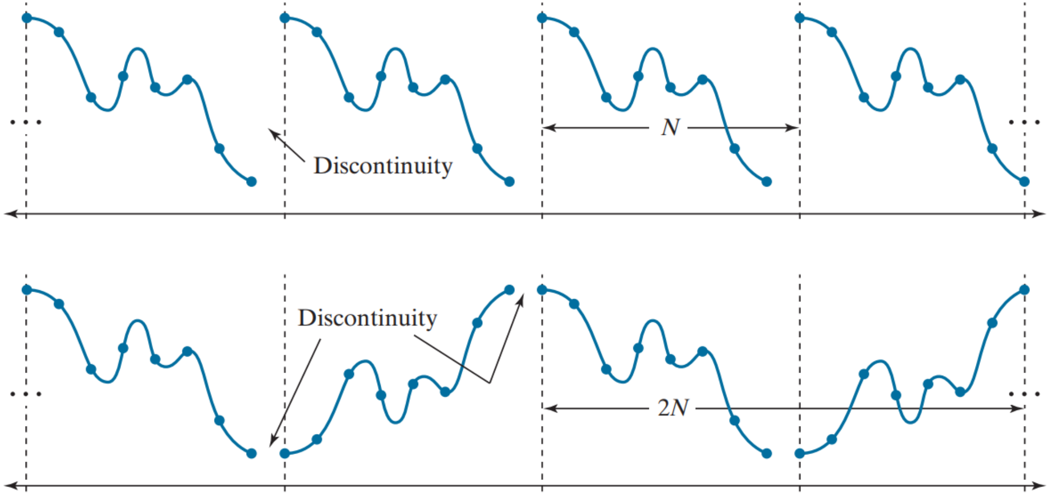


DCT vs DFT



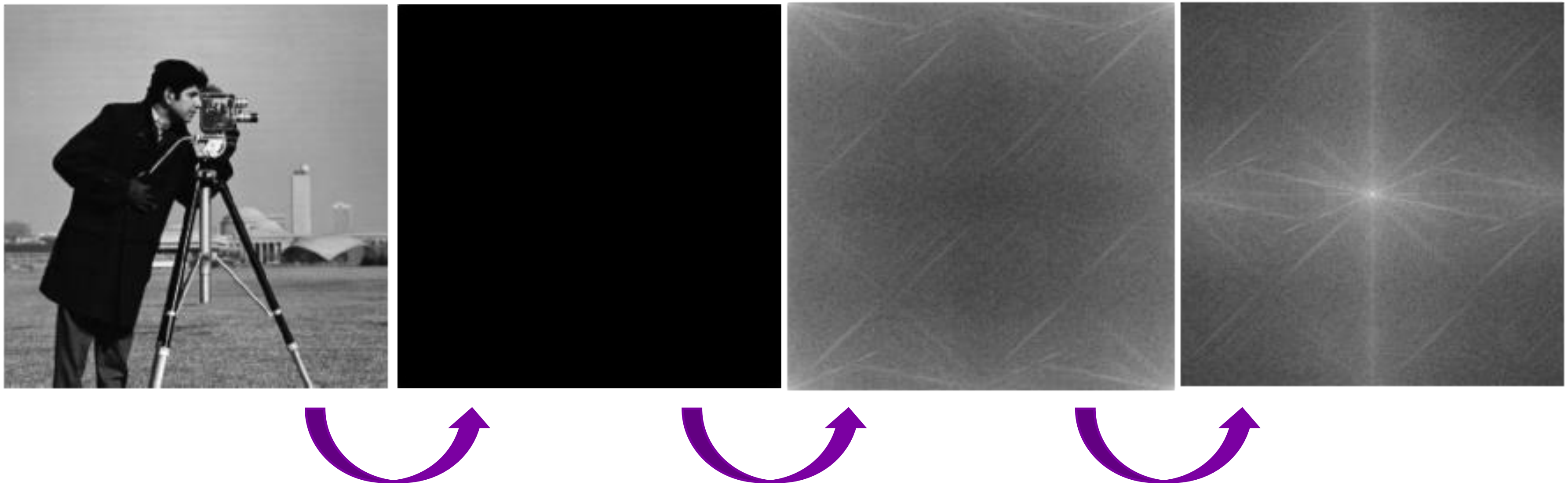
DCT

DFT



- DCT is good for image compression because it does not estimate high frequency for end of signal

DFT transform

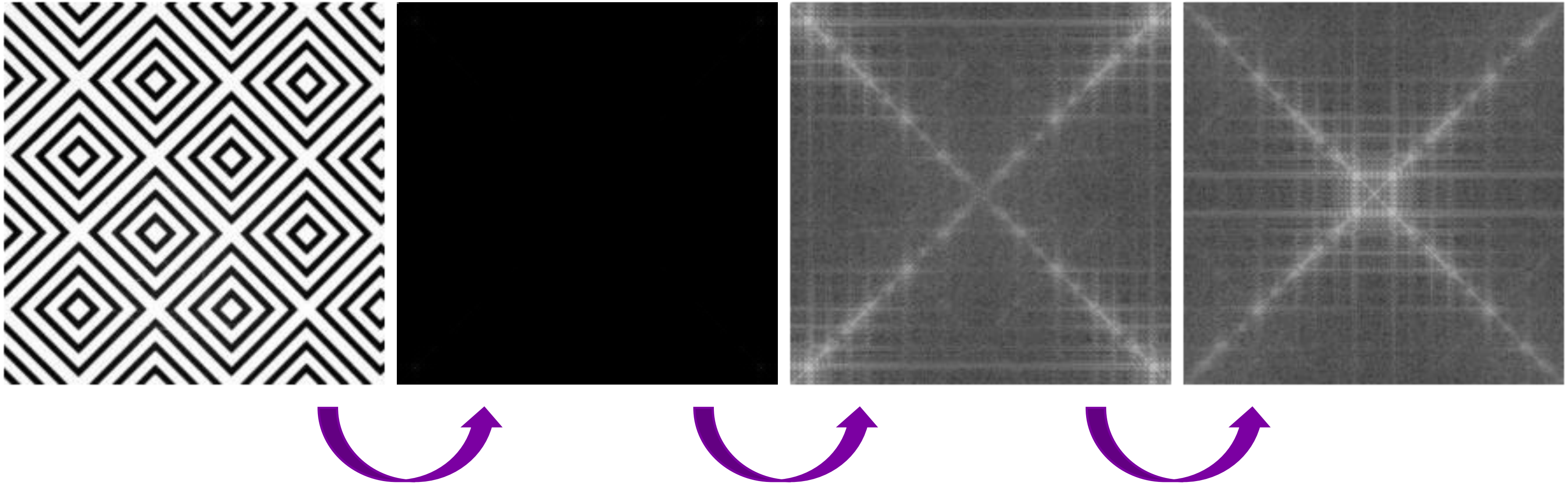


FFT and abs

20 * log(.)

shift

DFT transform

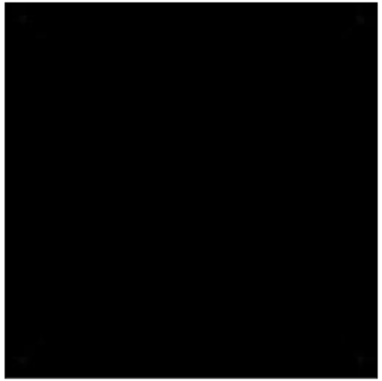


FFT and abs

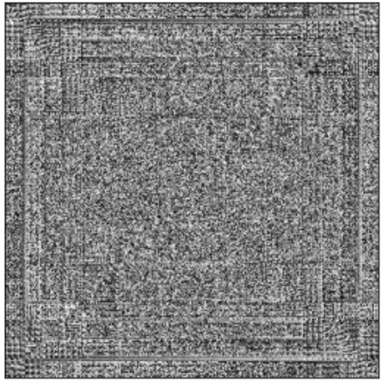
20 * log(.)

shift

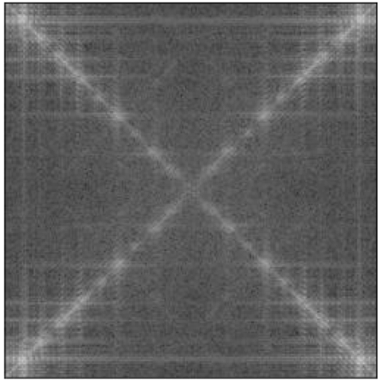
real



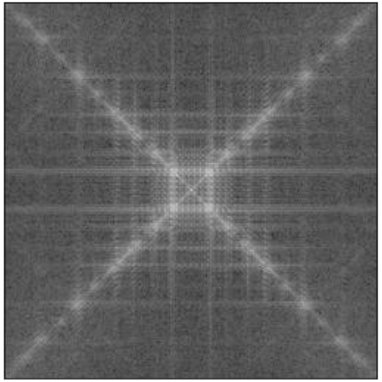
imaginary



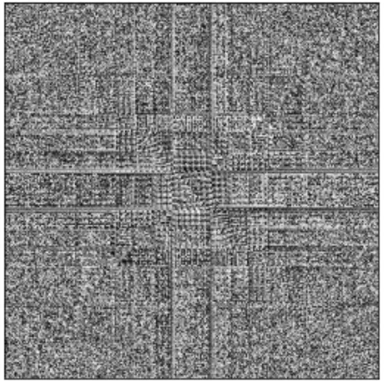
magnitude real



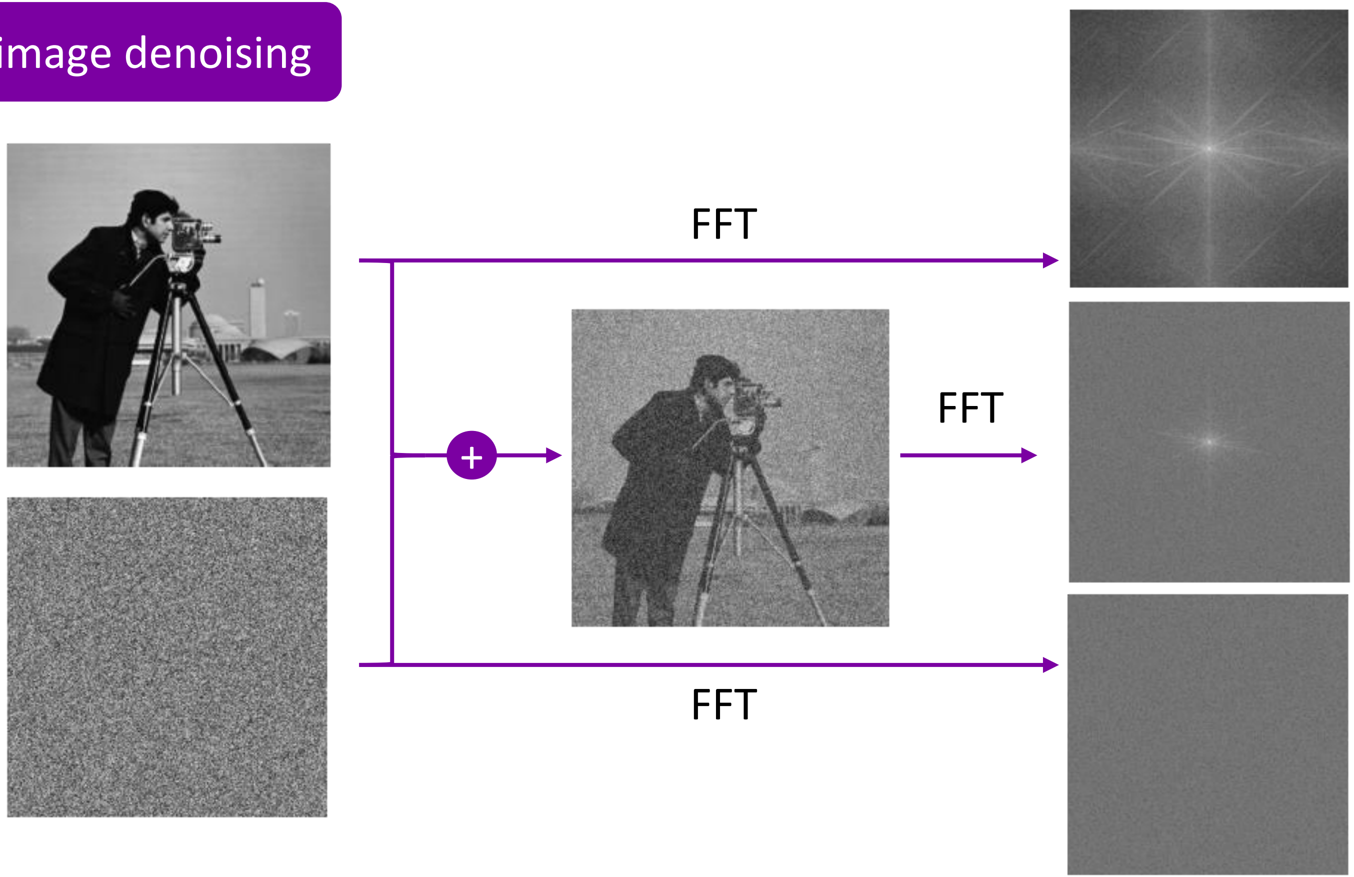
after shift real



after shift imaginary

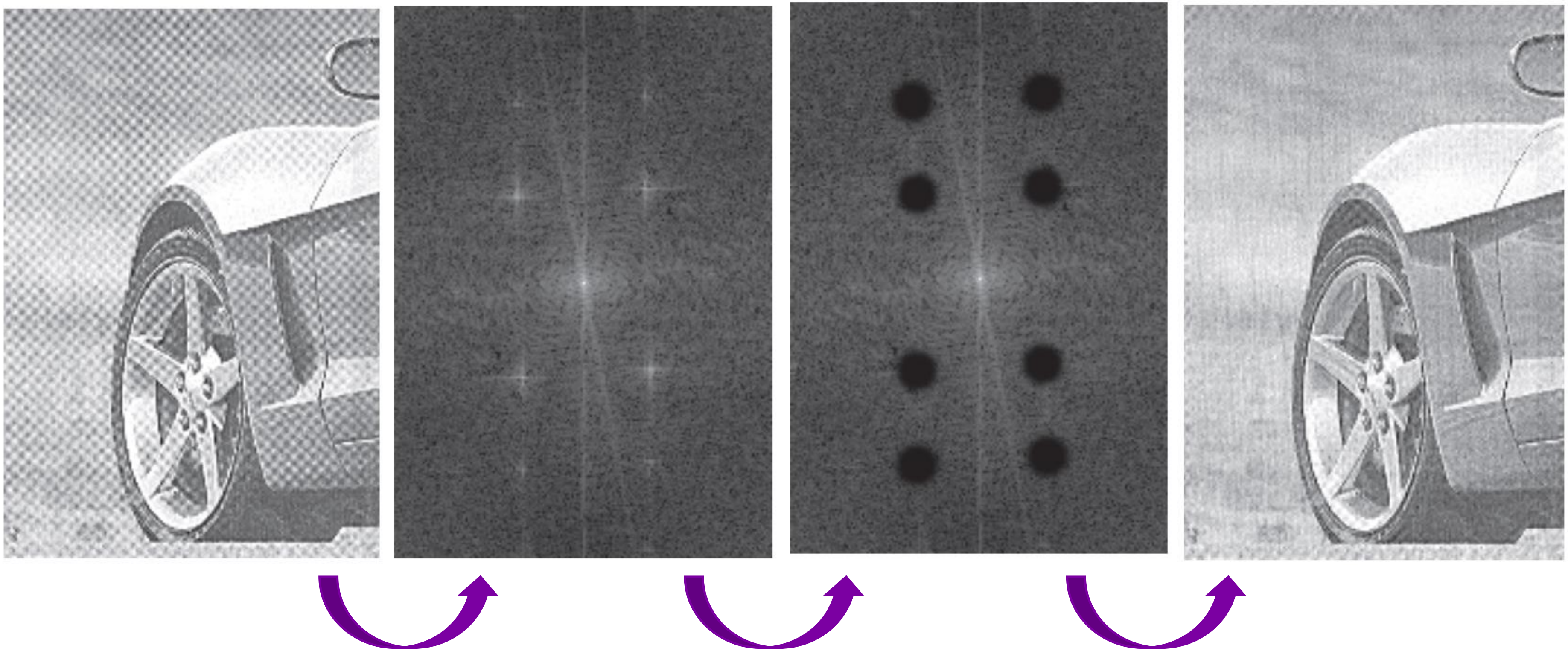


Noise reduction / image denoising



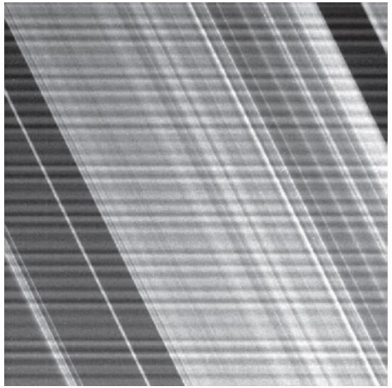
NOTE

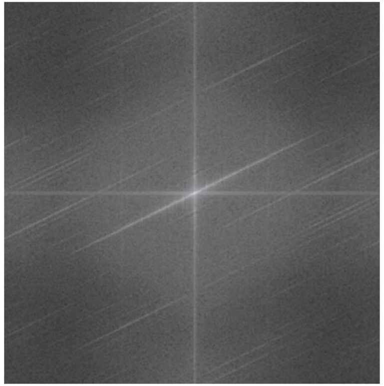
White noise that has value for all comprised frequency

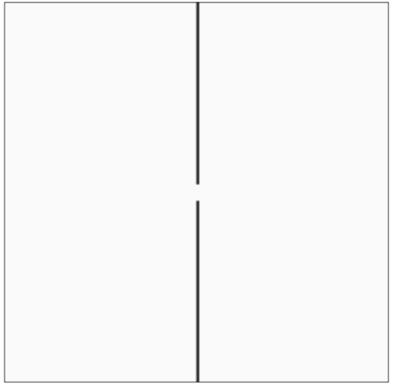


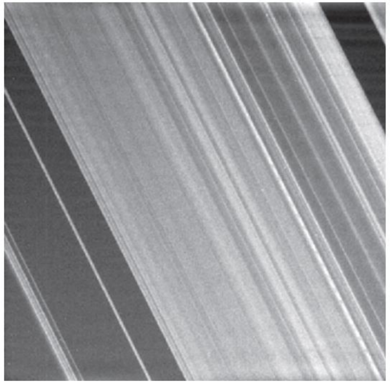
FFT Low pass filter Fourier inverse

Low pass filter

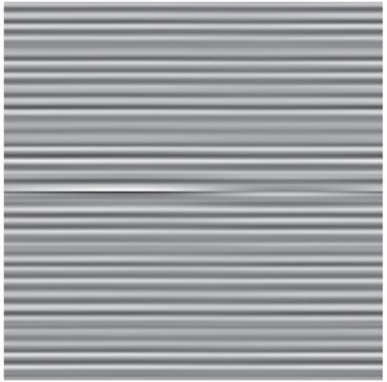








High pass filter



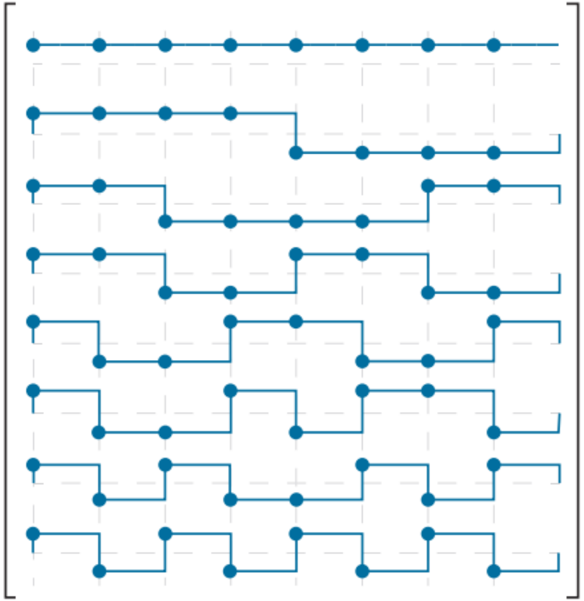
Walsh-Hadamard Transforms (WHT)

$$p_0(u) = b_{n-1}(u)$$
$$p_1(u) = b_{n-1}(u) + b_{n-2}(u)$$
$$p_2(u) = b_{n-2}(u) + b_{n-3}(u)$$
$$\vdots$$
$$p_{n-1}(u) = b_1(u) + b_0(u)$$

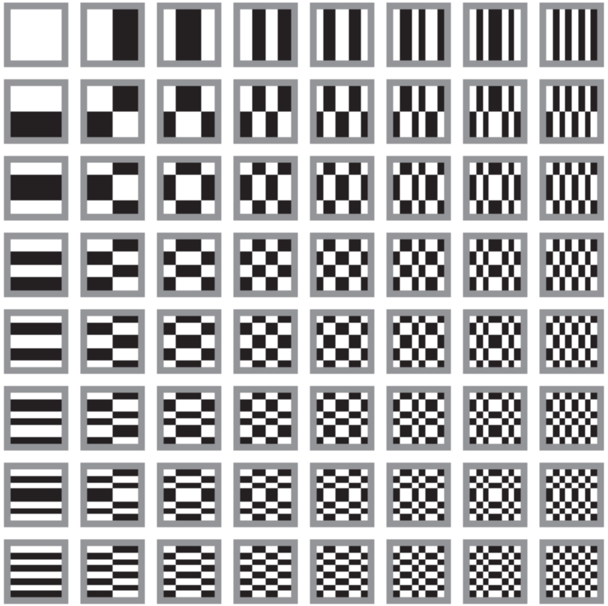
$N = 2^n$

Binary form of u : _ _ _

$b_2(u)$ $b_1(u)$ $b_0(u)$

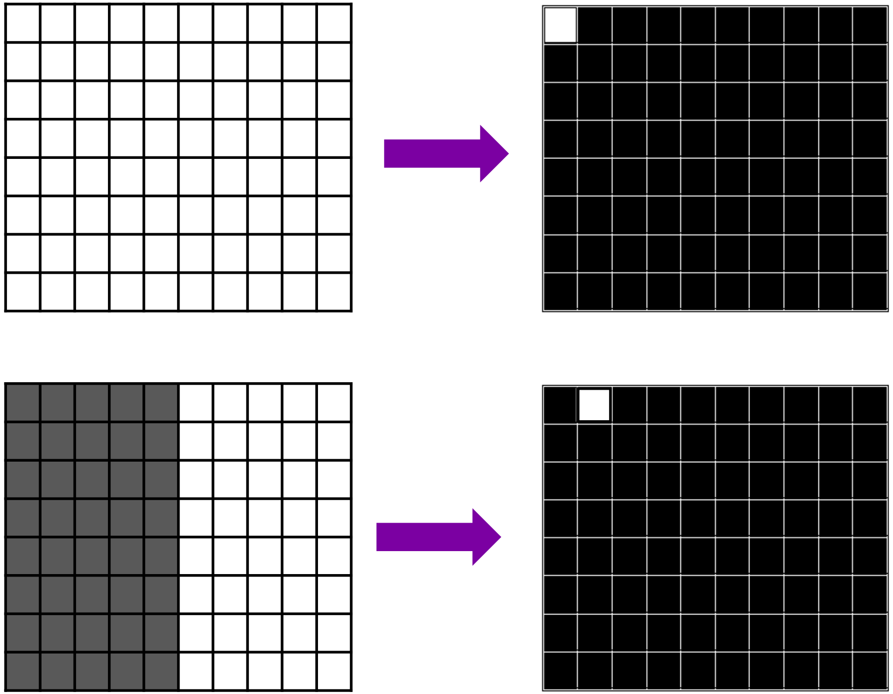


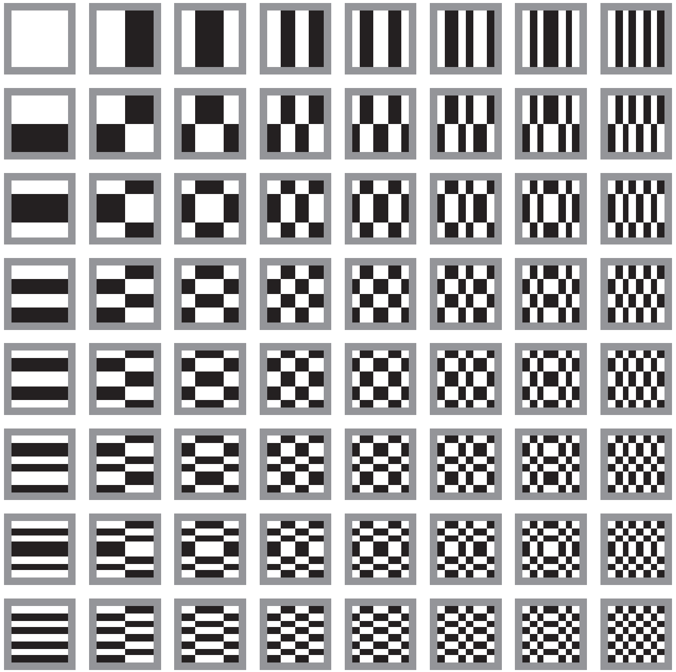
$$s(x,u) = \frac{1}{\sqrt{N}}(-1)^{\sum_{i=0}^{n-1} b_i(x)p_i(u)}$$



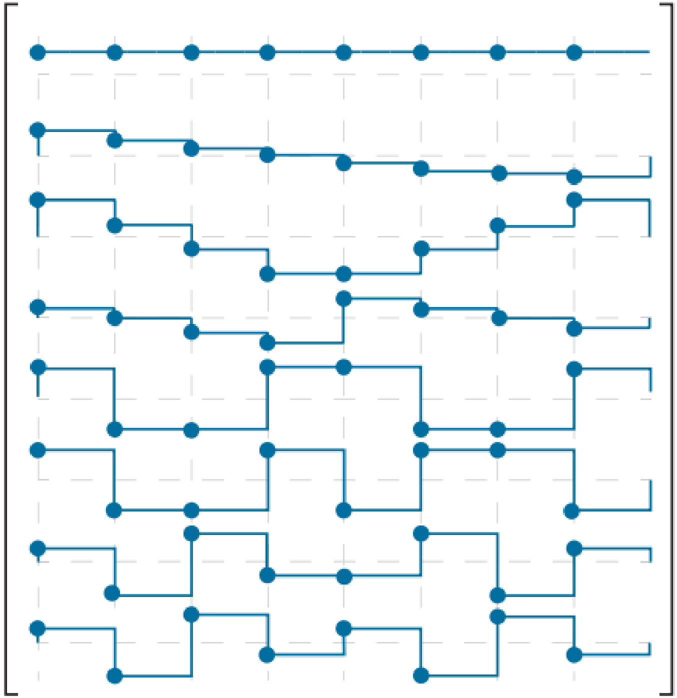
$$s(x,y,u,v) = \frac{1}{N}(-1)^{\sum_{i=0}^{n-1}[b_i(x)p_i(u)+b_i(y)p_i(v)]}$$

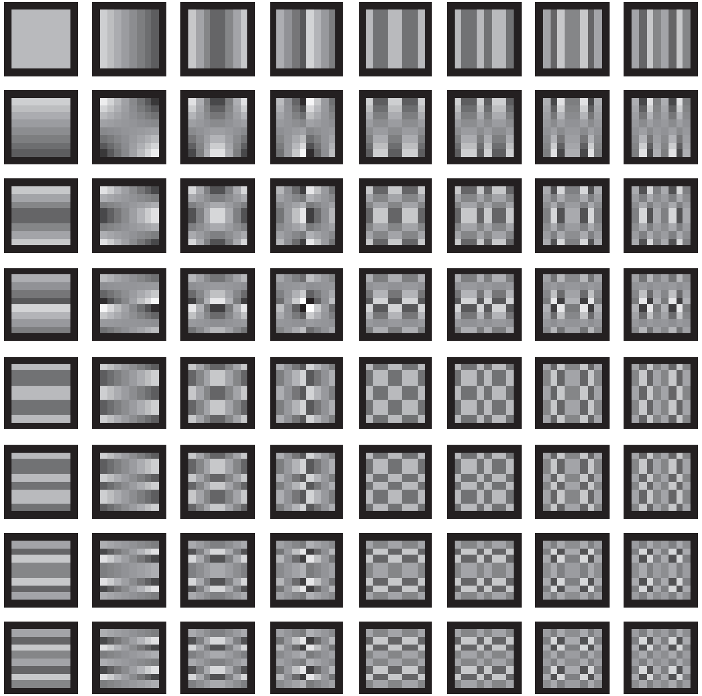
Walsh-Hadamard Transforms (WHT)



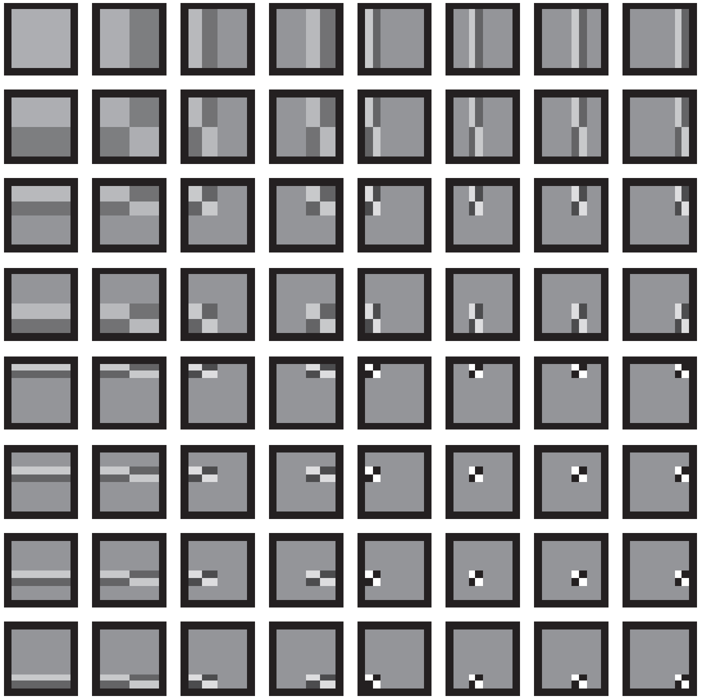


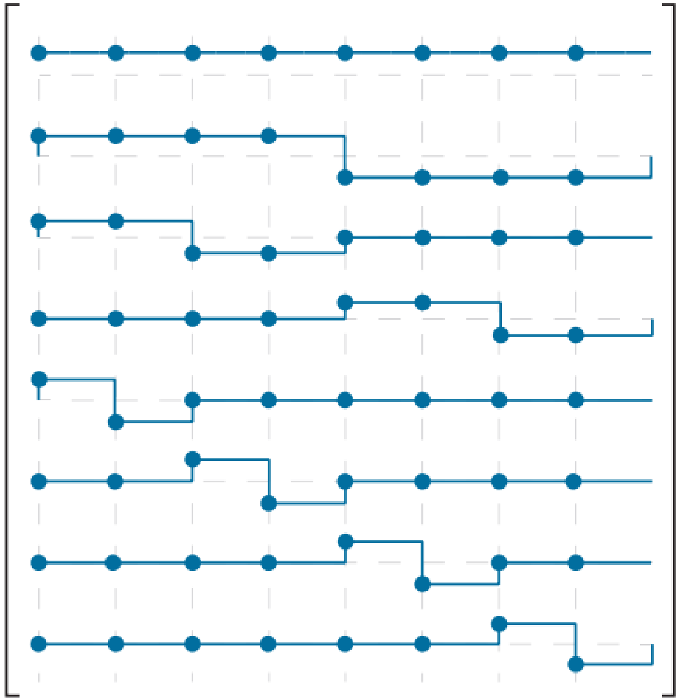
SLANT Transform

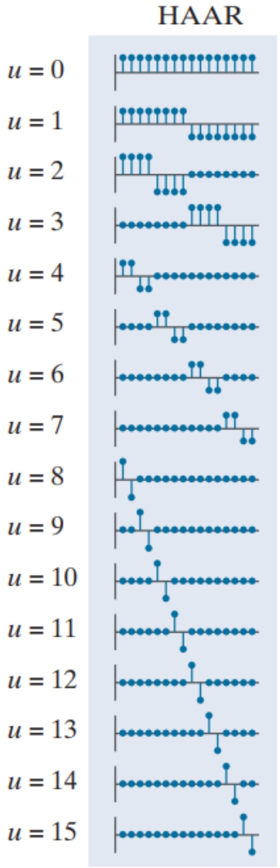




HAAR Transform





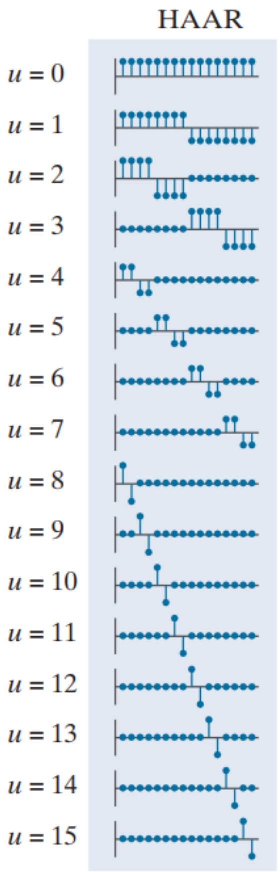


HAAR Transform

- Model the local changes in image
- It’s fast

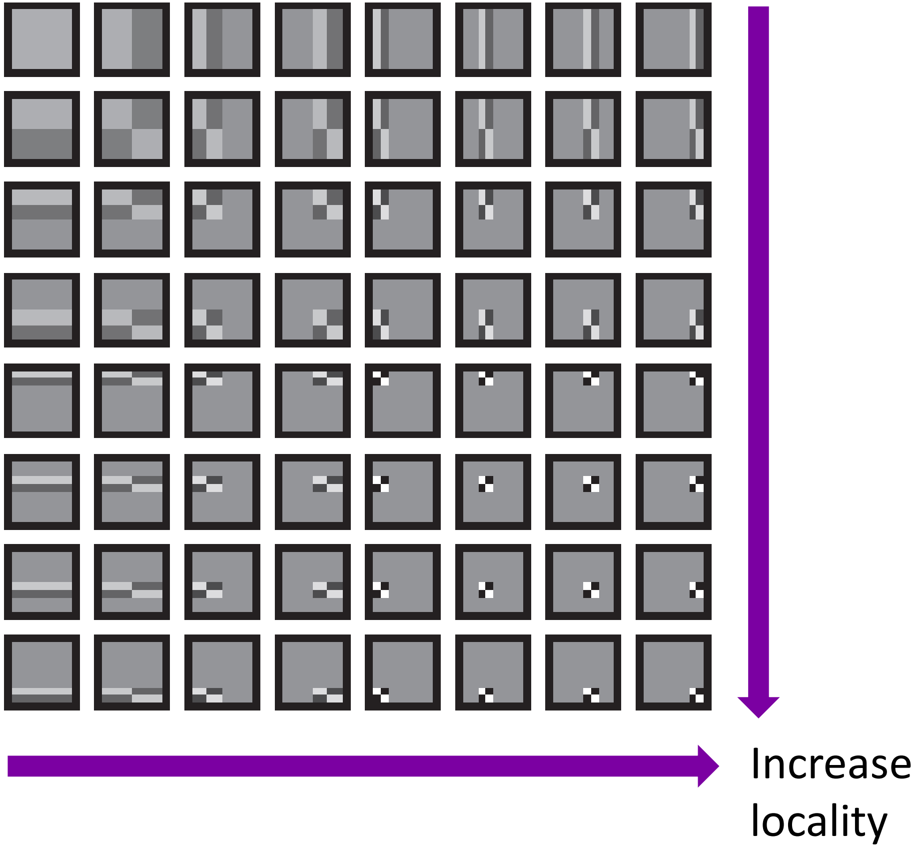
$$h_u(x) = \begin{cases} 1 & u = 0 \text{ and } 0 \le x < 1 \\ 2^{p/2} & u > 0 \text{ and } q/2^p \le x < (q+0.5)/2^p \\ -2^{p/2} & u > 0 \text{ and } (q+0.5)/2^p \le x < (q+1)/2^p \\ 0 & \text{otherwise} \end{cases}$$

$$u = 2^p + q$$



$$s(x,u) = \frac{1}{\sqrt{N}} h_u\left(\frac{x}{N}\right)$$

$$s(x,y,u,v) = \frac{1}{N} h_u\left(\frac{x}{N}\right) h_v\left(\frac{y}{N}\right)$$



Wavelet Transform

In 1987, wavelets were shown to be the foundation of a powerful new approach to signal processing and analysis called multiresolution theory

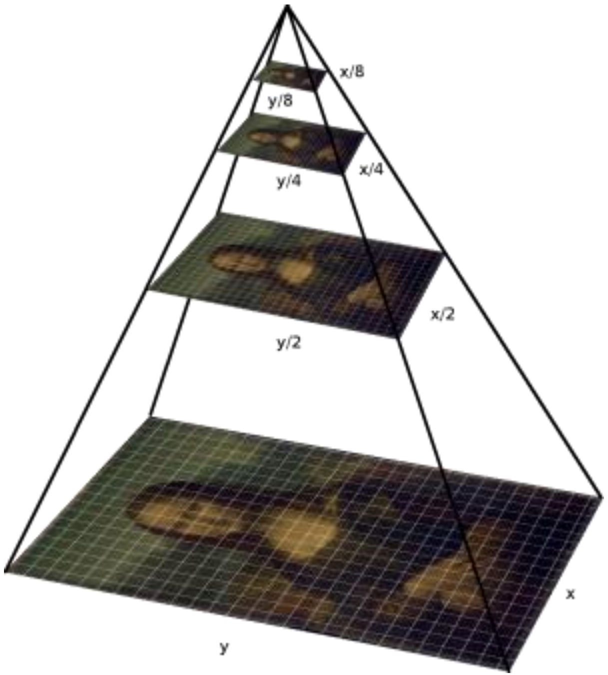
A scaling function is used to create a series of approximations of a function or image, each differing by a factor of 2 in resolution from its nearest neighboring approximations, and complementary functions, called wavelets, are used to encode the differences between adjacent approximations

discrete wavelet transform (DWT) uses those wavelets, together with a single scaling function, to represent a function or image as a linear combination of the wavelets and scaling function.

Wavelet Transform

- Scaling Function : generating approximation of signal (image)

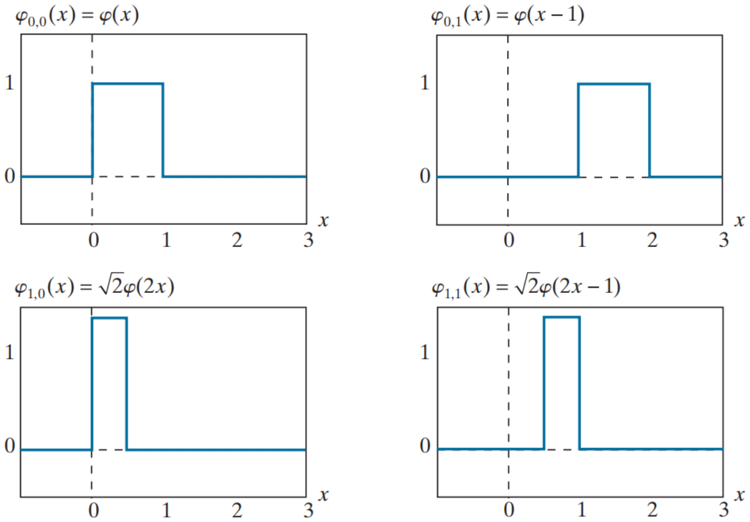
- Wavelet Function : generating differences between approximations



Scaling Function

Set of functions to approximate a signal

Consider the unit-height, unit-width scaling function :



NOTE

Scaling functions should be orthogonal and expand the whole space

Haar scaling function

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k)$$

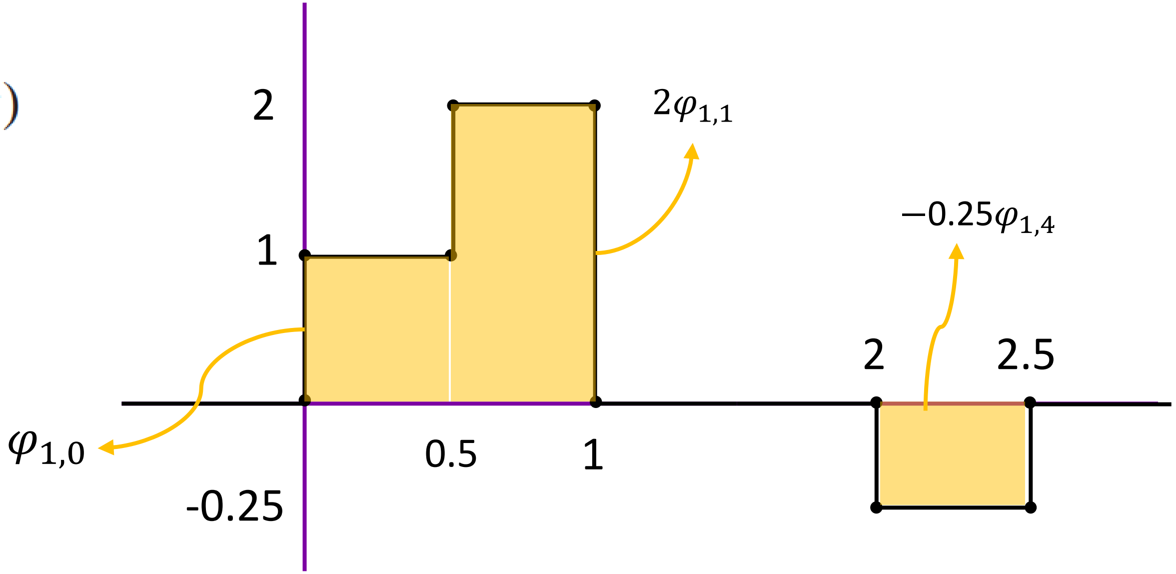
$$\varphi(x) = \begin{cases} 1 & 0 \le x < 1 \\ 0 & \text{otherwise} \end{cases}$$

Scaling Function

$$f(x) = \varphi_{1,0} + 2\varphi_{1,1} - 0.25\varphi_{1,4}$$

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k)$$

J = 1

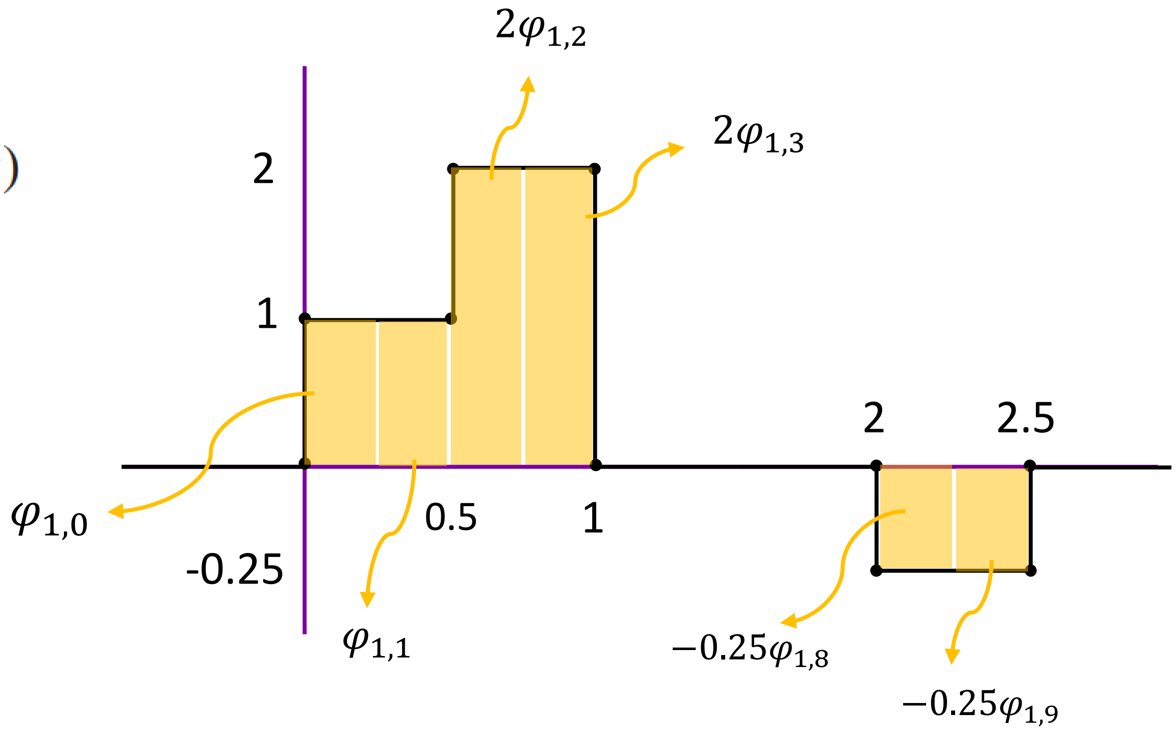


Scaling Function

$$f(x) = \varphi_{1,0} + \varphi_{1,1} + 2\varphi_{1,2} + 2\varphi_{1,3} - 0.25\varphi_{1,8} - 0.25\varphi_{1,9}$$

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k)$$

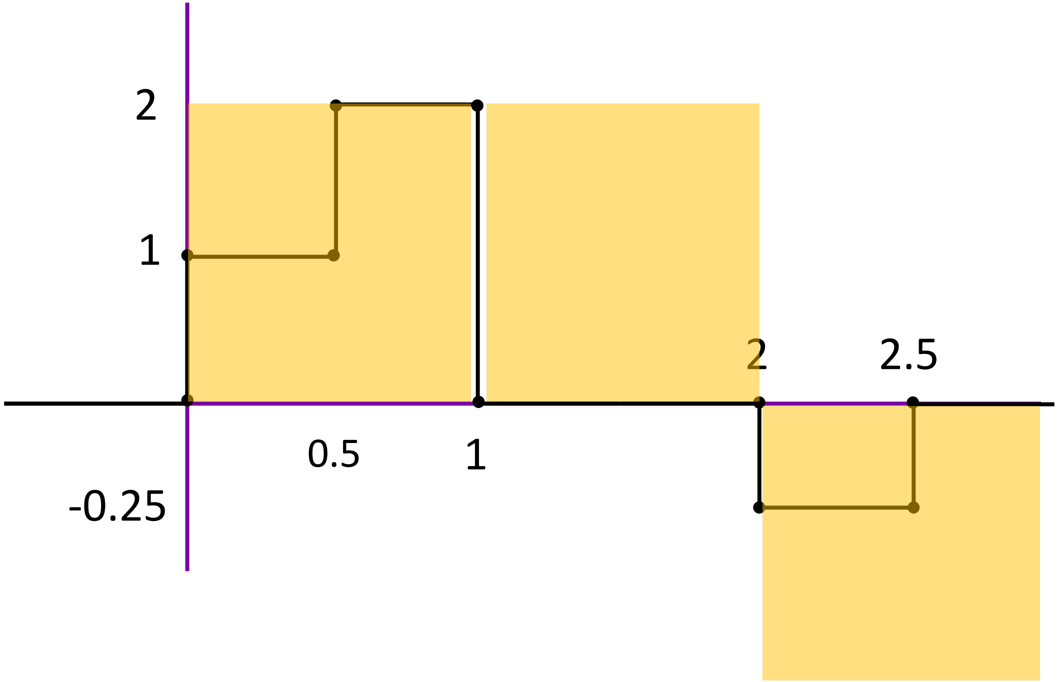
J = 2



Scaling Function

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k)$$

J = 0



Scaling Function

we can not estimate a function with $\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k)$ for any ‘ j ’ values

If we can estimate a function with j , we can estimate it with j+1 , j+2, …

If we can estimate a function with set of $\varphi_{j0,k}$, we call these functions V_{j0} that is
$f(x) \in Vj_0$

The function spaces spanned by the scaling function at low scales are nested within those spanned at higher scales. That is

$$V_{-\infty} \subset \ldots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \ldots \subset V_{\infty}$$

Scaling Function

we can estimate any function with V_∞

Base on previous notes , $\varphi(x)$ can be expressed as a linear combination of double-resolution copies of itself:

$$\underbrace{\varphi(x)}_{\varphi_{0,0}} = \sum_{k \in \mathbf{Z}} h_\varphi(k) \underbrace{\sqrt{2}\varphi(2x-k)}_{\varphi_{1,k}}$$

$dot\ (\varphi_{1,k})$

$$h_\varphi(k) = \frac{\sqrt{2}\,\langle \varphi(x), \varphi(2x-k) \rangle}{2\,\langle \varphi(2x-k), \varphi(2x-k) \rangle}$$

orthonormal

$$h_\varphi(k) = \left\langle \varphi(x), \sqrt{2}\varphi(2x-k) \right\rangle$$

scaling function coefficients

Example

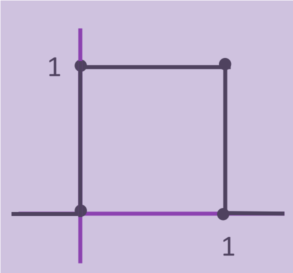
Compute scaling function coefficients for HAAR scaling function (j = 0)

Answer:

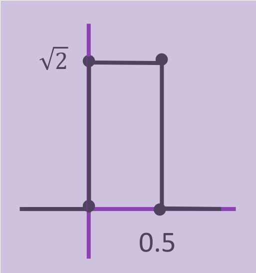
$$h_\varphi(k) = \left\langle \varphi(x), \sqrt{2}\varphi(2x-k) \right\rangle$$

k = 0

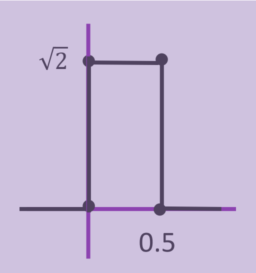
$h_\varphi(0)$ = <



,



> = $\int$

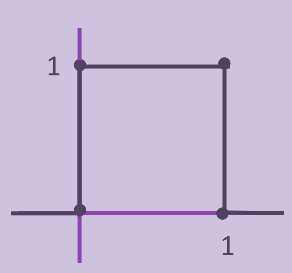


$= \dfrac{\sqrt{2}}{2}$

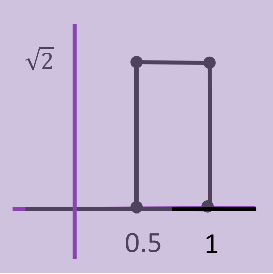
Example

k = 1

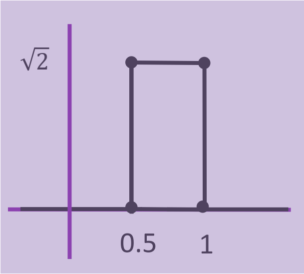
$h_\varphi(1) = <$



,



$> \; = \int$



$= \dfrac{\sqrt{2}}{2}$

$$\{h_\varphi(n) | n = 0, 1\} = \{1/\sqrt{2}, 1/\sqrt{2}\}$$

$$\begin{aligned}\varphi(x) &= \frac{1}{\sqrt{2}}\left[\sqrt{2}\varphi(2x)\right] + \frac{1}{\sqrt{2}}\left[\sqrt{2}\varphi(2x-1)\right] \\ &= \varphi(2x) + \varphi(2x-1)\end{aligned}$$

Wavelet Function

father scaling function φ(x) : that is, the set of scaled and translated functions $\varphi_{j,k}$(x)

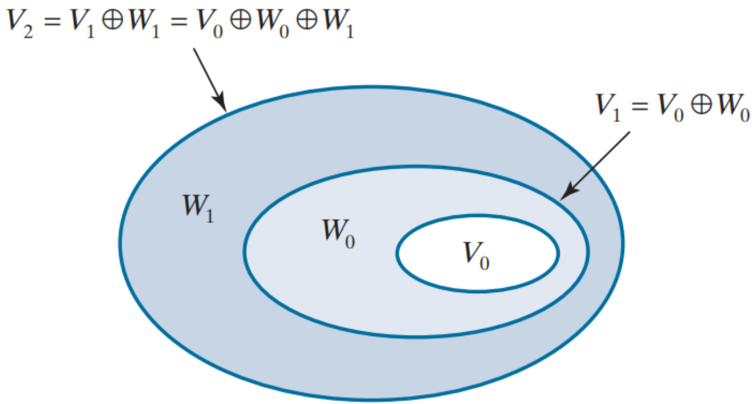
Given a father scaling function that meets the previous requirements, there exists a mother wavelet function Ψ(x)

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$$

We denote W_{j0} as Set of functions that can be represents $\Psi_{j0,k}$(x)

Wavelet Function

$\Psi_{j,k}(x)$ span the difference between any two adjacent scaling spaces



$$V_{j_0+1} = V_{j_0} \oplus W_{j_0}$$

detailed approximation

General approximation

$\oplus$ denotes the union of function spaces

Wavelet Function

$$< \varphi_{j,k}(\mathrm{x}), \varphi_{j,l}(\mathrm{x}) > = 0 \quad \mathrm{k} \neq l$$

$$< \varphi_{j,k}(\mathrm{x}), \Psi_{j,l}(\mathrm{x}) > = 0 \quad \mathrm{k} \neq l$$

$$\varphi(x) = \sum_{k \in \mathbf{Z}} h_{\varphi}(k)\sqrt{2}\varphi(2x - k)$$

$$\psi(x) = \sum_{k} h_{\psi}(k)\sqrt{2}\varphi(2x - k)$$

$$h_{\varphi}(k) = \frac{\sqrt{2}\,\langle\varphi(x), \varphi(2x - k)\rangle}{2\,\langle\varphi(2x - k), \varphi(2x - k)\rangle}$$

$$h_{\psi}(k) = \frac{\sqrt{2}\,\langle\psi(x), \varphi(2x - k)\rangle}{2\,\langle\varphi(2x - k), \varphi(2x - k)\rangle}$$

wavelet function coefficients

Wavelet Function

It can be demonstrated that : $h_\psi(k) = (-1)^k h_\varphi(1-k)$

$\varphi(x) \rightarrow h_\varphi(x) \rightarrow h_\Psi(x) \rightarrow \Psi(x)$

$$h_\varphi(k) = \frac{\sqrt{2}\,\langle \varphi(x), \varphi(2x-k)\rangle}{2\,\langle \varphi(2x-k), \varphi(2x-k)\rangle}$$

$$h_\psi(k) = (-1)^k h_\varphi(1-k)$$

$$\psi(x) = \sum_k h_\psi(k)\sqrt{2}\varphi(2x-k)$$

Example

Compute wavelet function coefficients for HAAR scaling function (j = 0)

$$\{h_\varphi(n)|n=0,1\}=\{1/\sqrt{2},1/\sqrt{2}\}$$

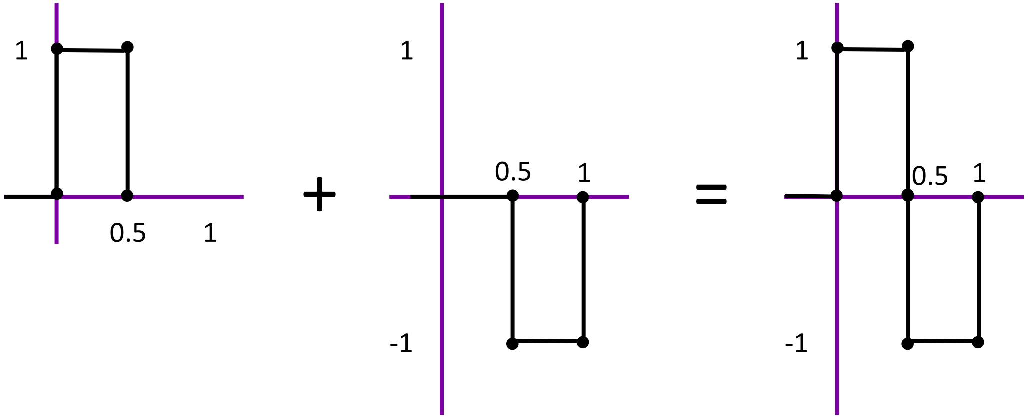
$$\varphi(x) = \frac{1}{\sqrt{2}}\left[\sqrt{2}\varphi(2x)\right]+\frac{1}{\sqrt{2}}\left[\sqrt{2}\varphi(2x-1)\right]$$
$$= \varphi(2x)+\varphi(2x-1)$$

$$h_\psi(k)=(-1)^k h_\varphi(1-k) \longrightarrow \begin{array}{l} h_\psi(0)=(-1)^0 h_\varphi(1-0)=1/\sqrt{2} \\ h_\psi(1)=(-1)^1 h_\varphi(1-1)=-1/\sqrt{2} \end{array} \longrightarrow \{h_\psi(n)|n=0,1\}=\{1/\sqrt{2},-1/\sqrt{2}\}$$

$$\psi(x)=\sum_k h_\psi(k)\sqrt{2}\varphi(2x-k) \longrightarrow \psi(x)=\varphi(2x)-\varphi(2x-1).$$

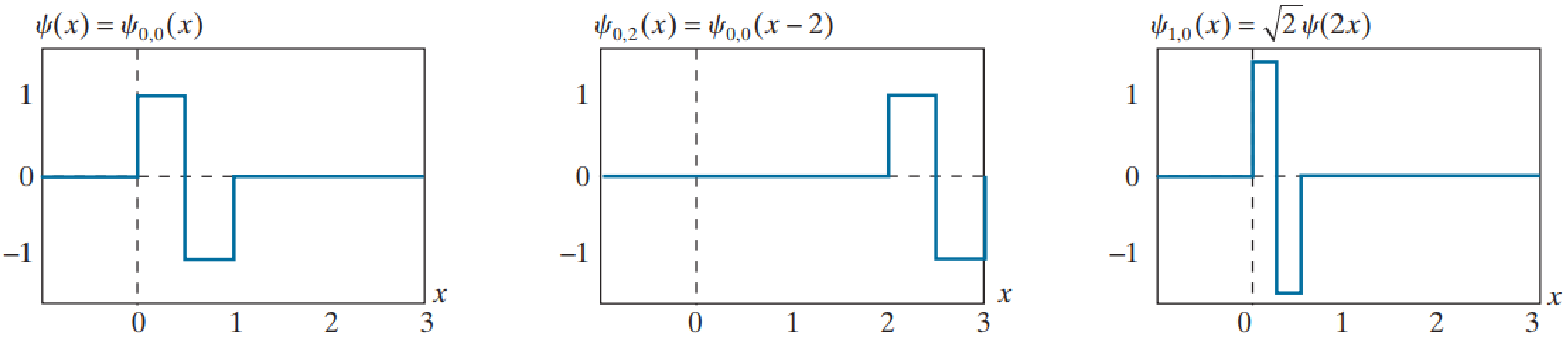
Example

$$\psi(x) = \varphi(2x) - \varphi(2x-1),$$

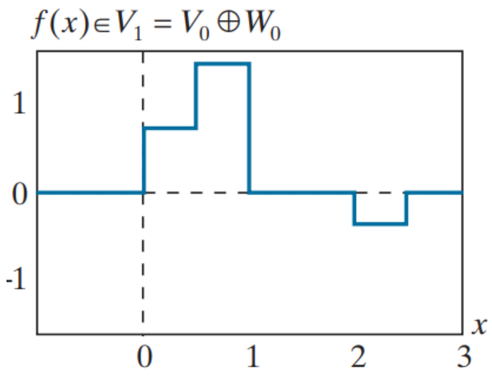


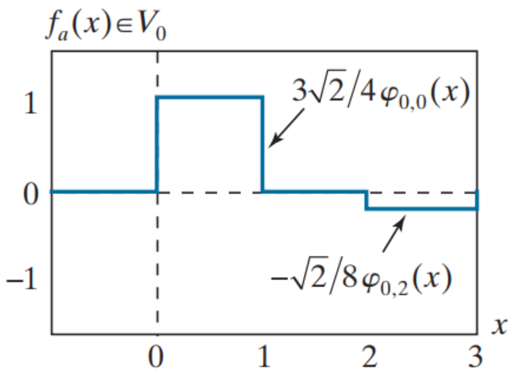
$$\psi(x) = \begin{cases} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x < 1 \\ 0 & \text{elsewhere} \end{cases}$$

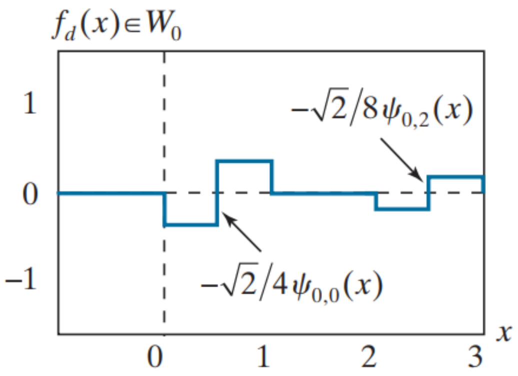
Wavelet Function



Example





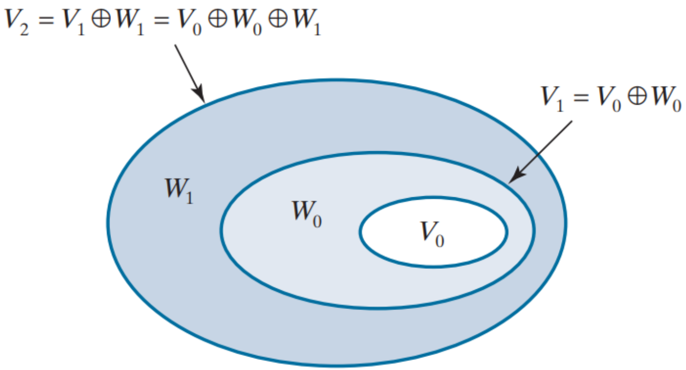


$$f(x) = f_a(x) + f_d(x)$$

$$f_a(x) = \frac{3\sqrt{2}}{4}\varphi_{0,0}(x) - \frac{\sqrt{2}}{8}\varphi_{0,2}(x)$$

$$f_d(x) = \frac{-\sqrt{2}}{4}\psi_{0,0}(x) - \frac{\sqrt{2}}{8}\psi_{0,2}(x)$$

Wavelet Series



$$L^2(\mathbf{R}) = V_{j_0} \oplus W_{j_0} \oplus W_{j_0+1} \oplus \ldots$$

$$f(x) = \sum_k c_{j_0}(k)\varphi_{j_0,k}(x) + \sum_{j=j_0}^{\infty} \sum_k d_j(k)\psi_{j,k}(x)$$

c_{j0} and d_j for $j \geq j_0$ are called approximation and detail coefficients, respectively.

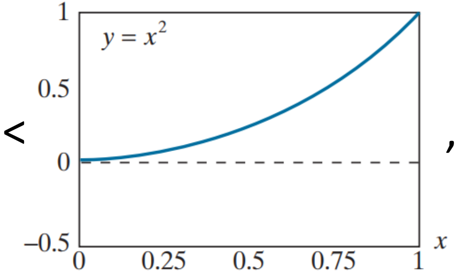
$$c_{j_0} = \langle f(x), \varphi_{j_0,k}(x) \rangle \qquad d_j = \langle f(x), \psi_{j,k}(x) \rangle$$

Example

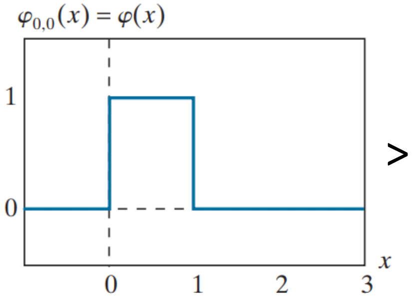
By consider the simple function below , compute approximation and detail coefficients (start with $j_0 = 0$)

$$y = \begin{cases} x^2 & 0 \le x \le 1 \\ 0 & \text{otherwise} \end{cases}$$

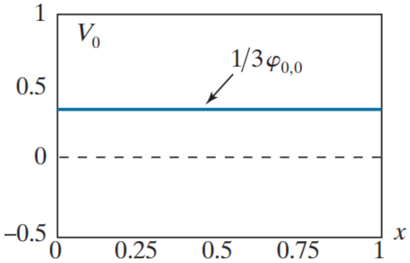
<



,



> =



$$c_{j_0} = \langle f(x), \varphi_{j_0,k}(x) \rangle \qquad c_0(0) = \int_0^1 x^2 \varphi_{0,0}(x)\,dx = \int_0^1 x^2\,dx = \left.\frac{x^3}{3}\right|_0^1 = \frac{1}{3}$$

Example

$$d_j = \left\langle f(x),\, \psi_{j,k}(x) \right\rangle$$

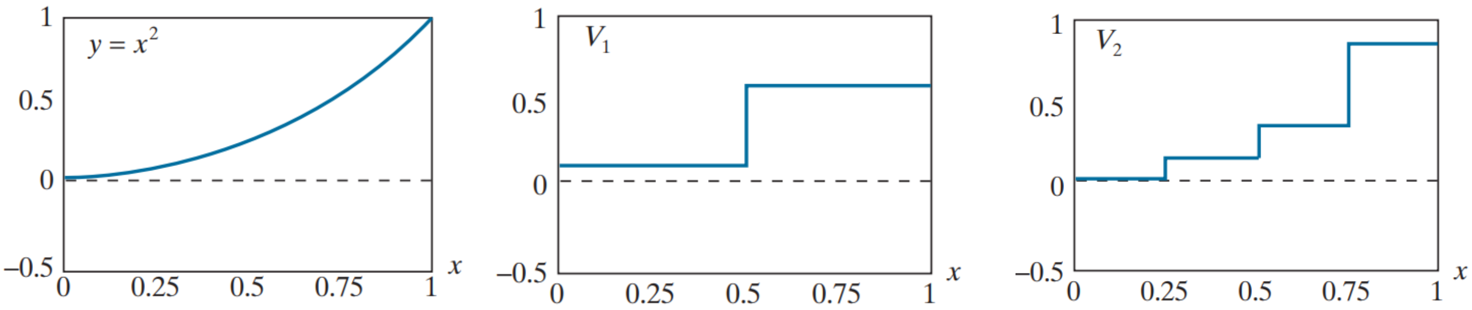
$$d_0(0) = \int_0^1 x^2 \psi_{0,0}(x)\,dx = \int_0^{0.5} x^2 dx - \int_{0.5}^1 x^2 dx = -\frac{1}{4}$$

$$d_1(0) = \int_0^1 x^2 \psi_{1,0}(x)\,dx = \int_0^{0.25} x^2 \sqrt{2} dx - \int_{0.25}^{0.5} x^2 \sqrt{2} dx = -\frac{\sqrt{2}}{32}$$

$$d_1(1) = \int_0^1 x^2 \psi_{1,1}(x)\,dx = \int_{0.5}^{0.75} x^2 \sqrt{2} dx - \int_{0.75}^1 x^2 \sqrt{2} dx = -\frac{3\sqrt{2}}{32}$$

Example

$$\underbrace{\underbrace{\underbrace{y = \frac{1}{3}\varphi_{0,0}(x)}_{V_0} + \underbrace{\left[-\frac{1}{4}\psi_{0,0}(x)\right]}_{W_0}}_{V_1 = V_0 \oplus W_0} + \underbrace{\left[-\frac{\sqrt{2}}{32}\psi_{1,0}(x) - \frac{3\sqrt{2}}{32}\psi_{1,1}(x)\right]}_{W_1}}_{V_2 = V_1 \oplus W_1 = V_0 \oplus W_0 \oplus W_1} + \cdots$$



Discrete Wavelet Transform

Like a Fourier series expansion, the wavelet series expansion of the previous section maps a function of a single continuous variable into a sequence of discrete coefficients. If the function being expanded is discrete, the coefficients of the expansion are its discrete wavelet transform (DWT)

$$f(x) = \frac{1}{\sqrt{N}}\left[T_\varphi(0,0)\varphi(x) + \sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1} T_\psi(j,k)\psi_{j,k}(x) \right]$$

$$T_\varphi(0,0) = \left\langle f(x), \varphi_{0,0}(x) \right\rangle = \left\langle f(x), \varphi(x) \right\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x)\varphi^*(x)$$

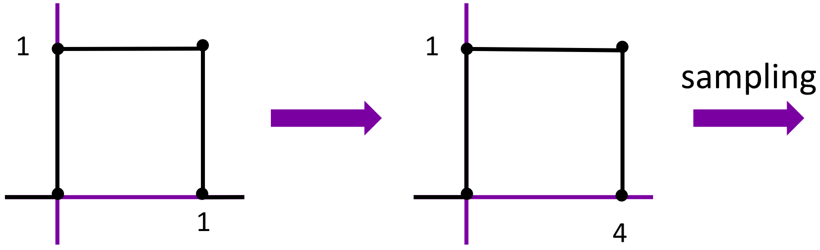
$$T_\psi(j,k) = \left\langle f(x), \psi_{j,k}(x) \right\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x)\psi_{j,k}^*(x)$$

Example

$$f(x)=\frac{1}{\sqrt{N}}\left[T_{\varphi}(0,0)\varphi(x)+\sum_{j=0}^{J-1}\sum_{k=0}^{2^{j}-1}T_{\psi}(j,k)\psi_{j,k}(x)\right]$$

F(x)

1
4
-3
0



φ(x)

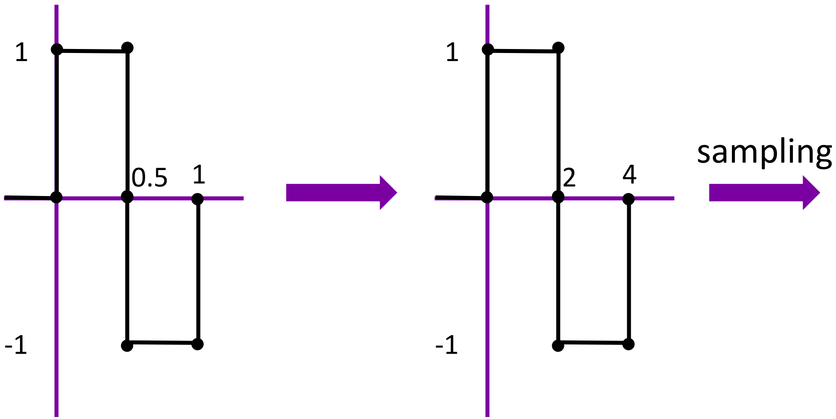
1
1
1
1

Example

$$f(x)=\frac{1}{\sqrt{N}}\left[T_{\varphi}(0,0)\varphi(x)+\sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1}T_{\psi}(j,k)\psi_{j,k}(x)\right]$$

F(x)

1
4
-3
0



$\Psi_{0,0}(x)$

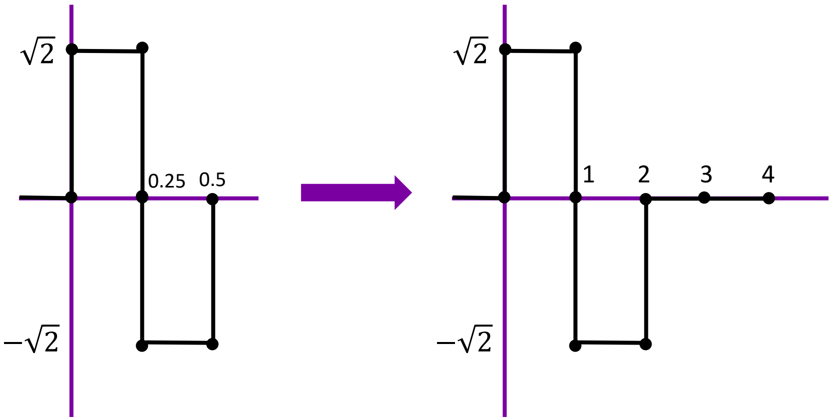
1
1
-1
-1

Example

$$f(x)=\frac{1}{\sqrt{N}}\left[T_{\varphi}(0,0)\varphi(x)+\sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1}T_{\psi}(j,k)\psi_{j,k}(x)\right]$$

F(x)

1
4
-3
0



sampling

$\Psi_{1,0}(x)$

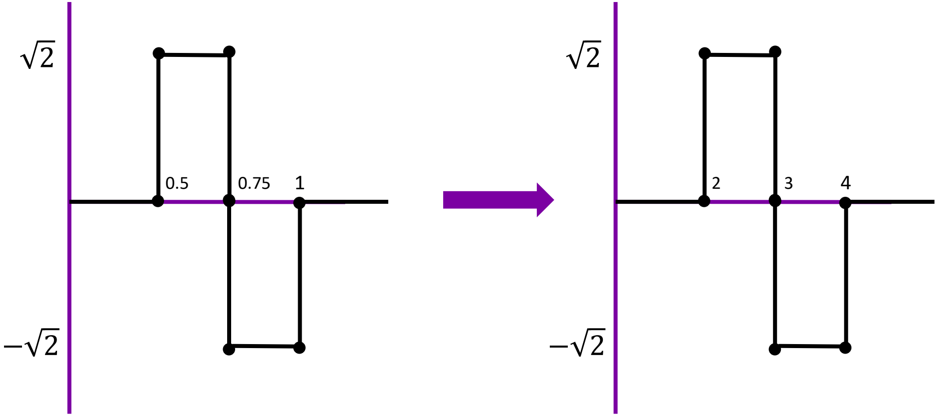
$\sqrt{2}$
$-\sqrt{2}$
0
0

Example

$$f(x)=\frac{1}{\sqrt{N}}\left[T_{\varphi}(0,0)\varphi(x)+\sum_{j=0}^{J-1}\sum_{k=0}^{2^{j}-1}T_{\psi}(j,k)\psi_{j,k}(x)\right]$$

F(x)

1
4
-3
0



sampling →

$\Psi_{1,1}(x)$

0
0
$\sqrt{2}$
$-\sqrt{2}$

Example

$$f(x)=\frac{1}{\sqrt{N}}\left[T_{\varphi}(0,0)\varphi(x)+\sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1}T_{\psi}(j,k)\psi_{j,k}(x)\right]$$

$$\underset{F(x)}{\begin{bmatrix}1\\4\\-3\\0\end{bmatrix}} = 1\underset{\varphi(x)}{\begin{bmatrix}1\\1\\1\\1\end{bmatrix}} +4\underset{\Psi_{0,0}(x)}{\begin{bmatrix}1\\1\\-1\\-1\end{bmatrix}} -1.5\sqrt{2}\underset{\Psi_{1,0}(x)}{\begin{bmatrix}\sqrt{2}\\-\sqrt{2}\\0\\0\end{bmatrix}} -1.5\sqrt{2}\underset{\Psi_{1,1}(x)}{\begin{bmatrix}0\\0\\\sqrt{2}\\-\sqrt{2}\end{bmatrix}}$$

2D Wavelet Transform

The 1-D wavelet transform of the previous section is easily extended to 2-D functions such as images.

$$\varphi(x, y) = \varphi(x)\varphi(y)$$

Linear Combination of two φ

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

Horizontal

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

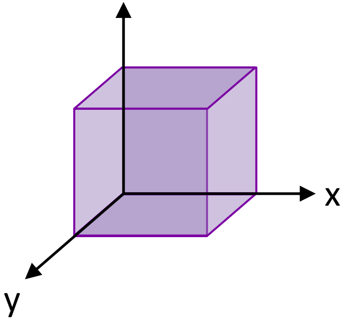
Vertical

$$\psi^D(x, y) = \psi(x)\psi(y)$$

Diagonal

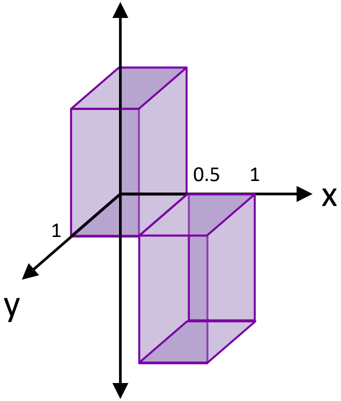
2D Wavelet Transform

$\varphi(x,y)$



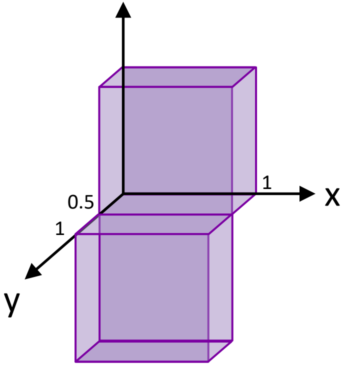
1	1
1	1

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



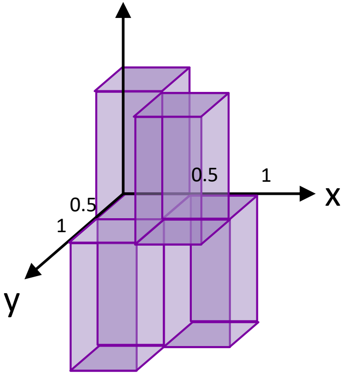
1	-1
1	-1

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



1	1
-1	-1

$\psi^D(x,y) = \psi(x)\psi(y)$

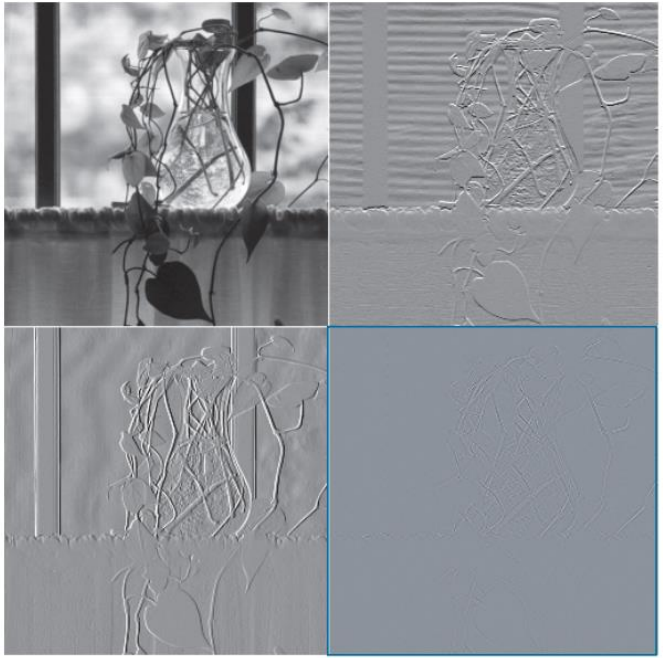


1	-1
-1	1

2D Wavelet Transform

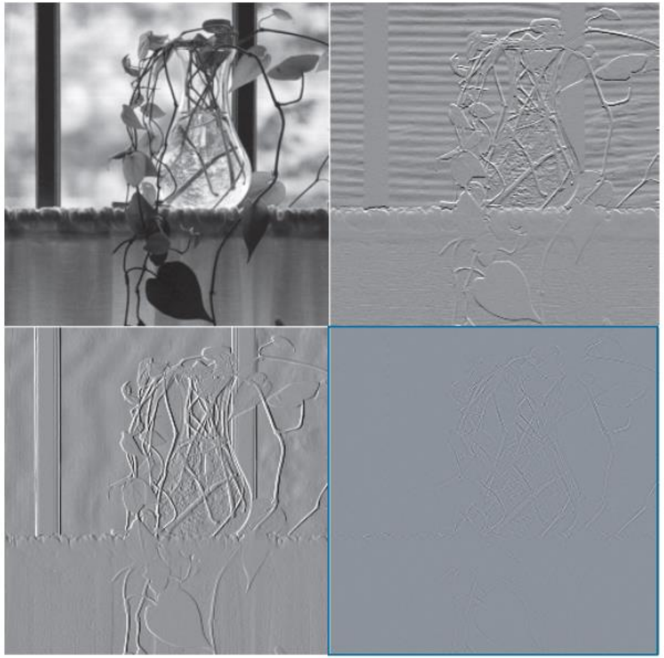
V_j

$V_{j-1} + W_{j-1}^{H} + W_{j-1}^{V} + W_{j-1}^{O}$

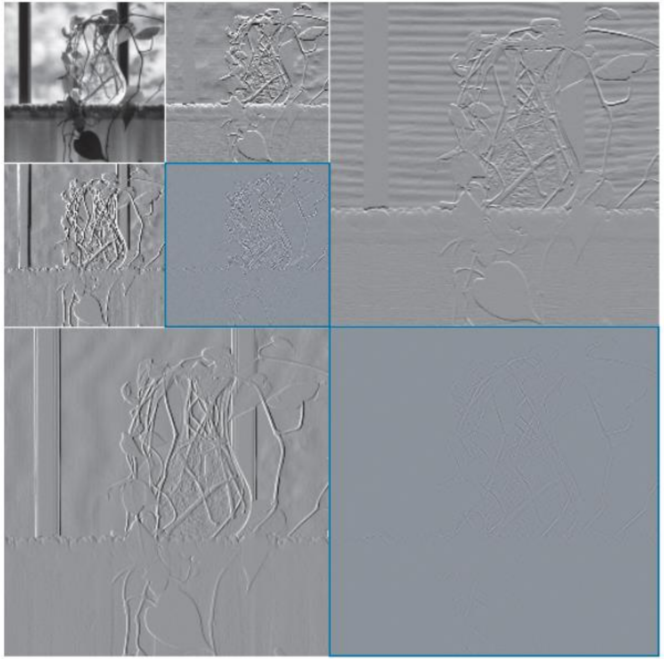


2D Wavelet Transform

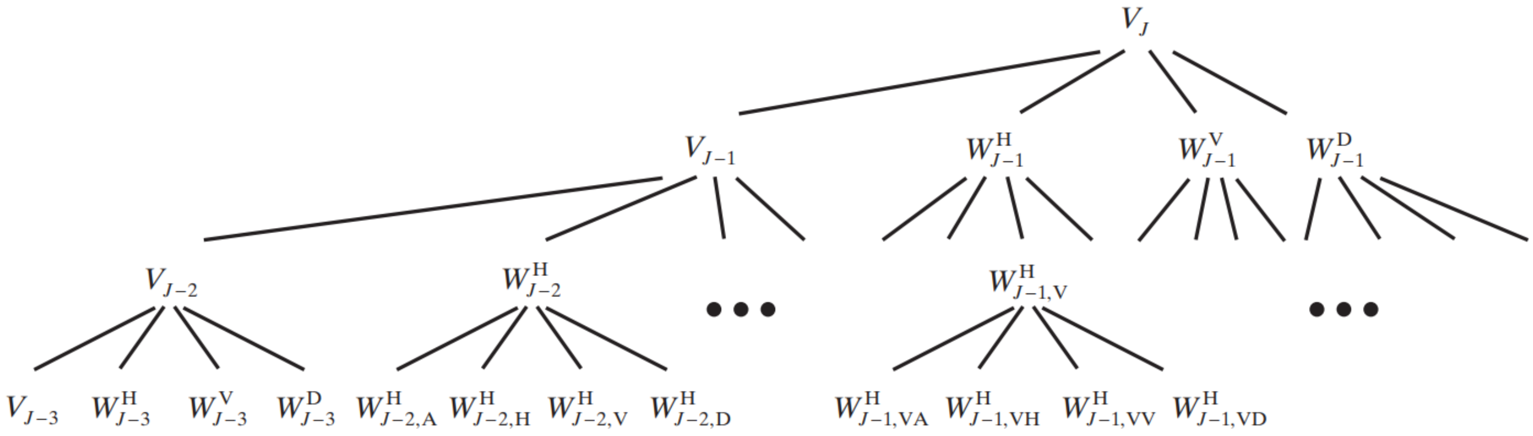
$$V_{j-1} + W_{j-1}^{H} + W_{j-1}^{V} + W_{j-1}^{O}$$

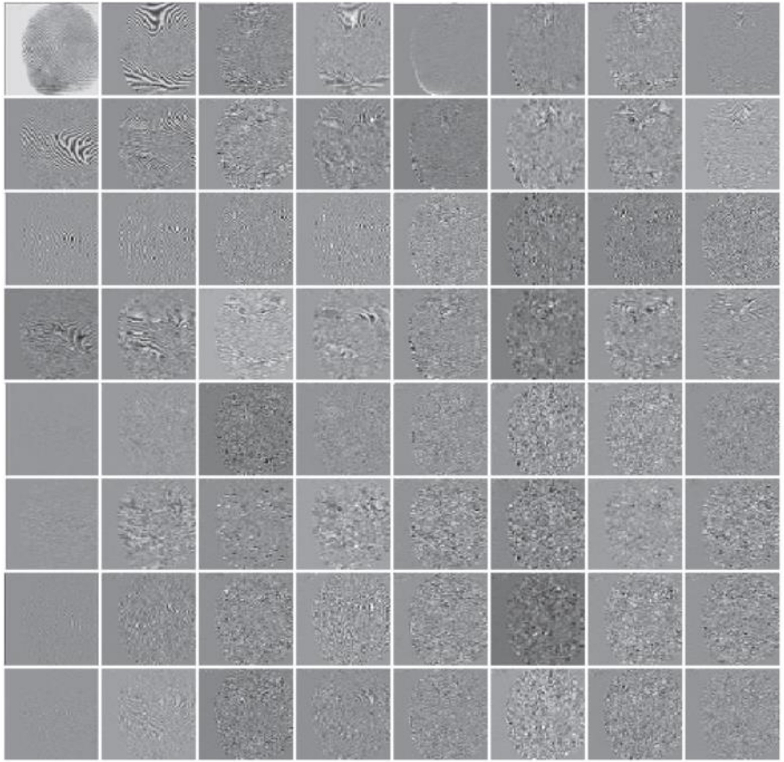


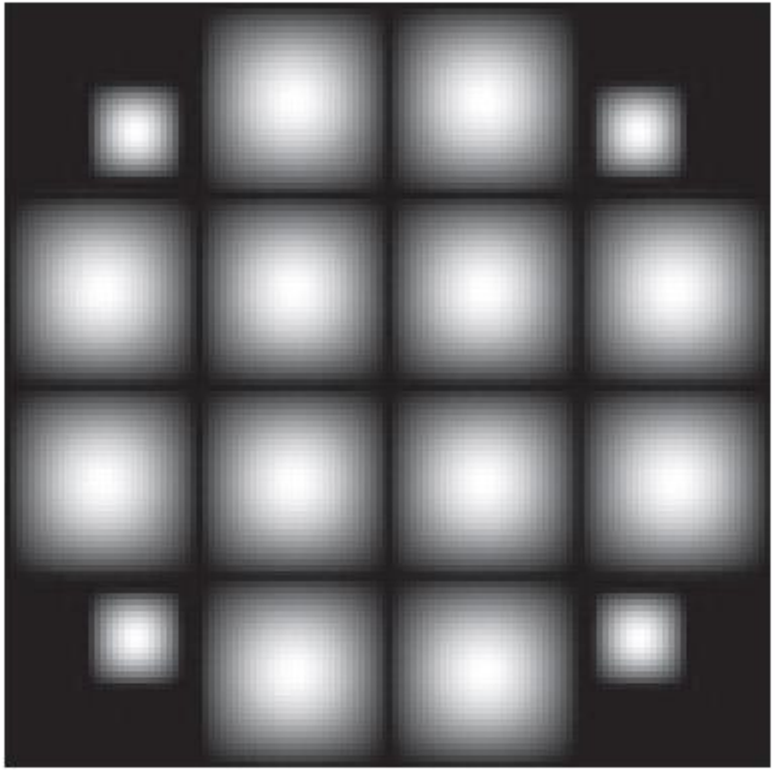
$$(V_{j-2} + W_{j-2}^{H} + W_{j-2}^{V} + W_{j-2}^{O}) + W_{j-1}^{H} + W_{j-1}^{V} + W_{j-1}^{O}$$

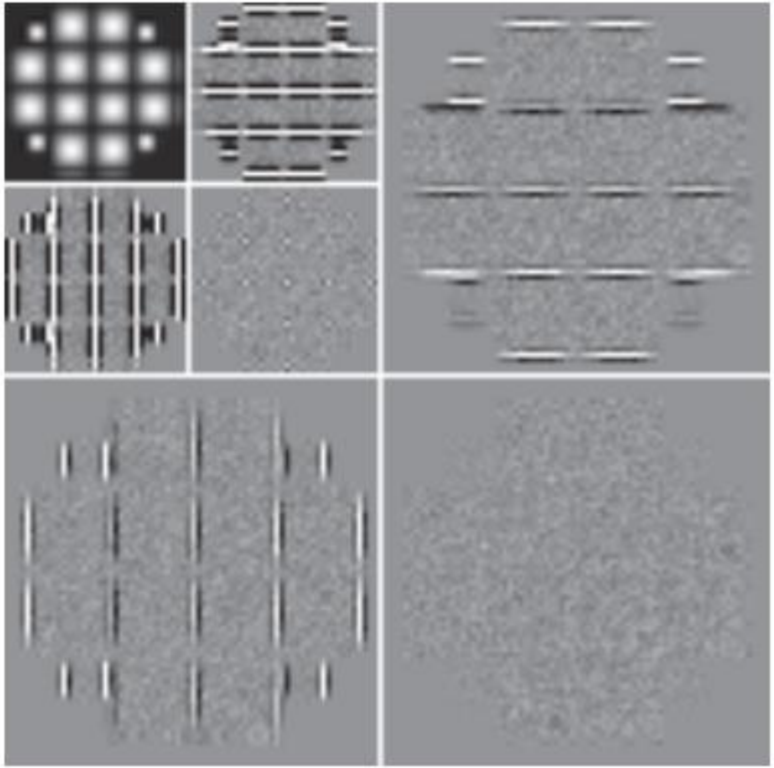


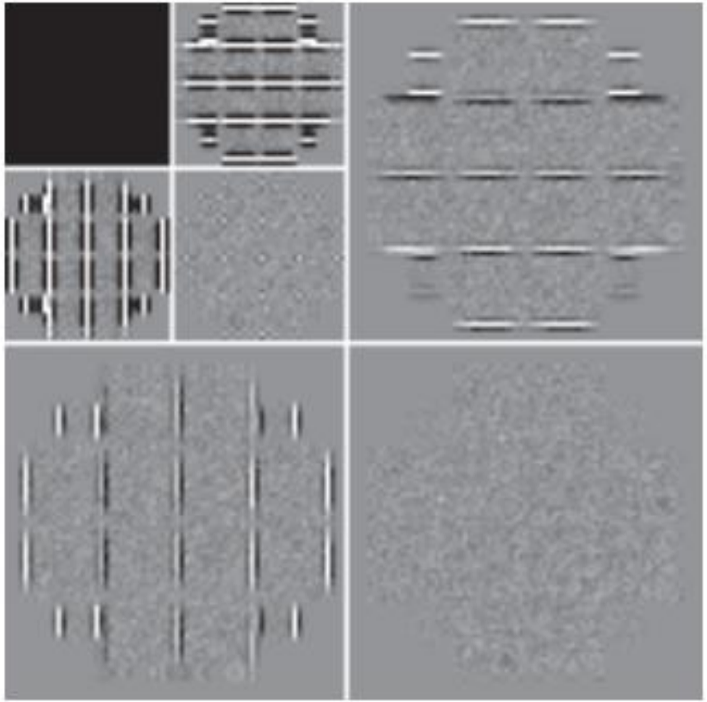
2D Wavelet Transform

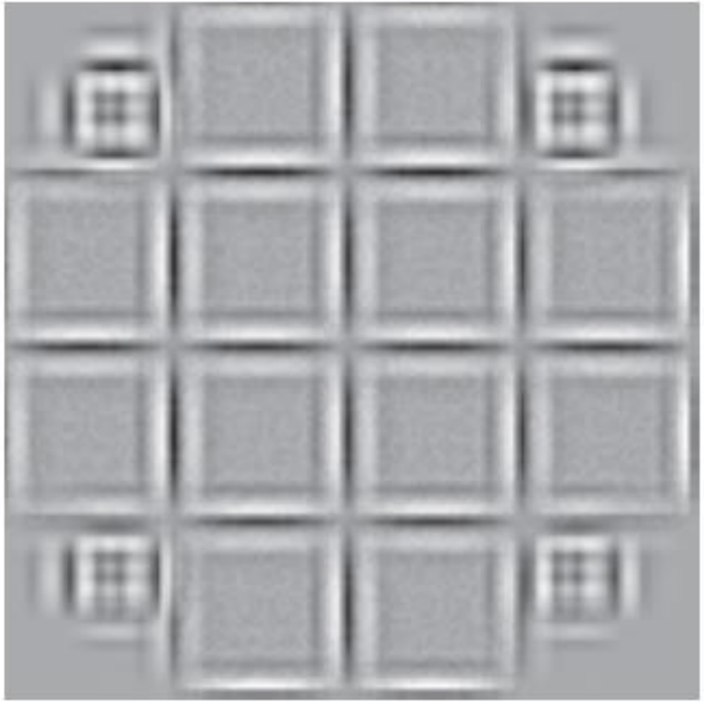


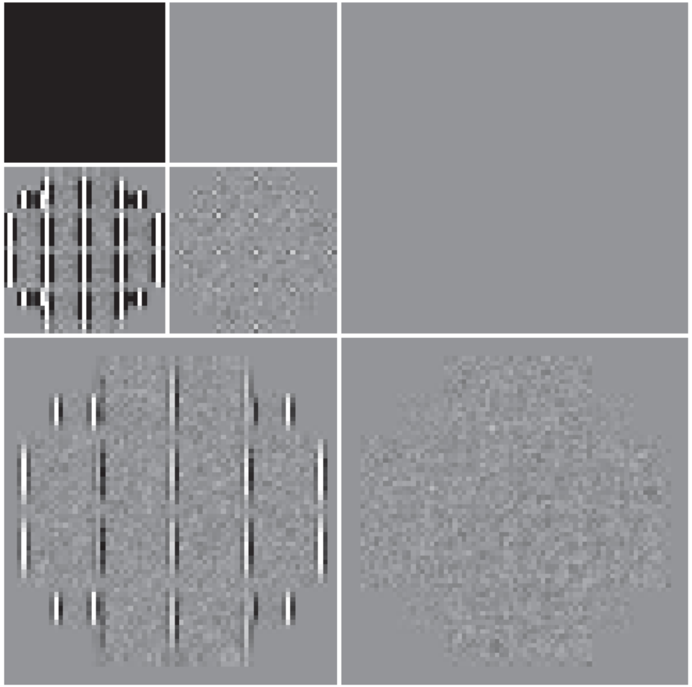


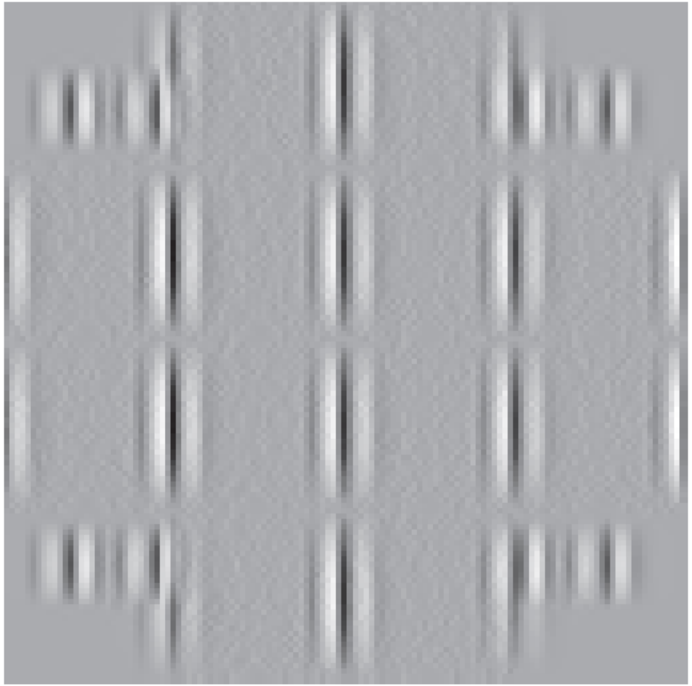










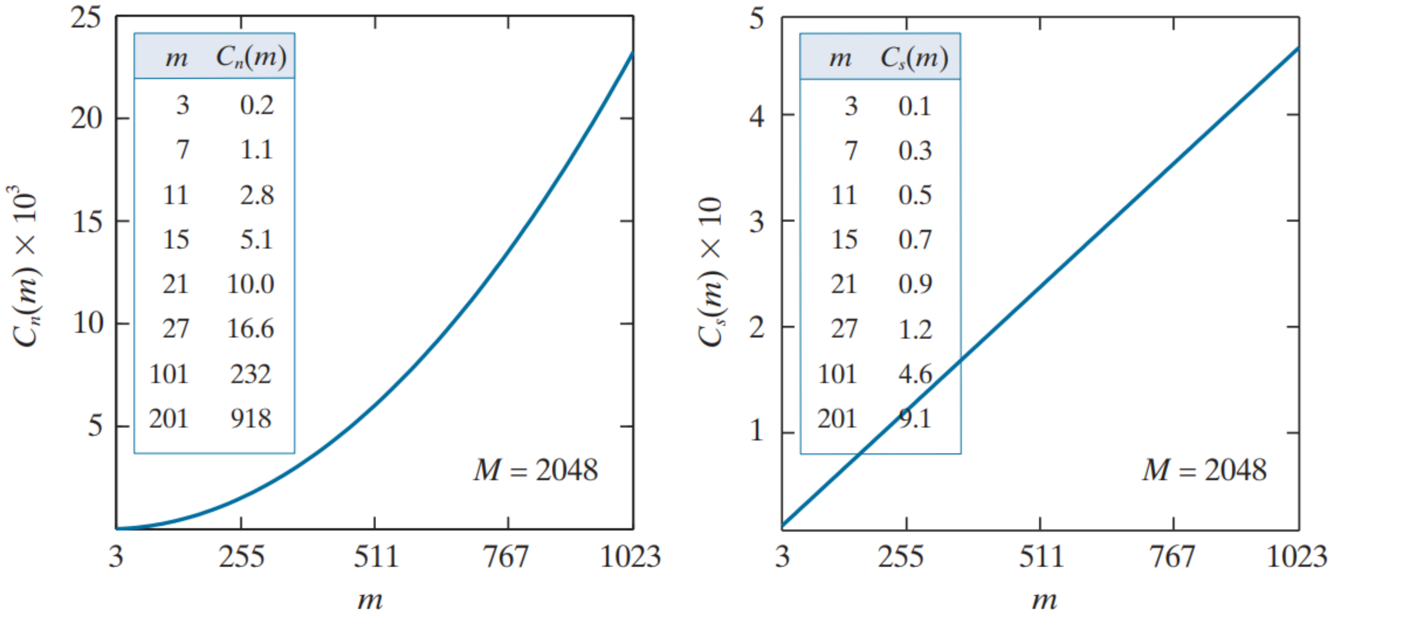


To get an idea of the relative computational advantages of filtering in the frequency versus the spatial domain, consider square images and kernels, of sizes M × M and m × m , respectively. The computational advantage (as a function of kernel size) of filtering one such image with the FFT as opposed to using a nonseparable kernel is defined as

$$C_n(m) = \frac{M^2 m^2}{2M^2 \log_2 M^2} = \frac{m^2}{4 \log_2 M}$$

If the kernel is separable, the advantage becomes

$$C_s(m) = \frac{2M^2 m}{2M^2 \log_2 M^2} = \frac{m}{2 \log_2 M}$$



a b

FIGURE 4.2
(a) Computational advantage of the FFT over non-separable spatial kernels.
(b) Advantage over separable kernels.
The numbers for $C(m)$ in the inset tables are not to be multiplied by the factors of 10 shown for the curves.

END