Introduction to Image Transform

- Fourier Transform
- Discrete Fourier Transform
- Fast Fourier Transform (FFT) O($N^2 \log(N)$)
- Discrete Cosine (Sine) Transform (*JPEG*)
- Discrete Wavelet Transform (*JP2*)
  - ♢ Haar transform
  - ♢ Daubechies' Four transform
- Singular Value Decomposition (SVD)
  - ♢ Spectrum Decomposition
  - • Hotelling (Karhunen-L*o*eve) Transform
  - ♢ Principal Component Analysis (PCA)

♣ Other Transforms (Gabor, 9/7 and 5/3 Wavelets)

Introduction to Fourier Transform

Let $f(x)$ be a continuous function of a real variable x. The Fourier transform of $f(x)$ is defined by

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-2\pi j u x} dx$$

Given $F(u)$, $f(x)$ can be obtained by using the inverse Fourier transform

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{2\pi j u x} du$$

In practical applications, $f(x)$ is a real function, however, its Fourier transform $F(u)$ is in general *complex* and can he written as

$$F(u) = R(u) + jI(u) = |F(u)| e^{-j\phi(u)}$$

where $R(u)$ and $I(u)$ are the real and imaginary component, respectively. $|F(u)|$ is called the Fourier spectrum of $f(x)$, and $\phi(u)$ is called the phase angle of $f(x)$. $|F(u)|^2$ is called the *Fourier power spectrum* or *spectral density) of* $f(x)$.

Example: *Let* $f(x) = A$ *for* $0 \leq x \leq X$, *and* $f(x) = 0$ *otherwise. Then*

$$F(u) = \frac{A \sin(\pi u X)}{\pi u} e^{-j\pi u X}$$

$$|F(u)| = \frac{A \sin(\pi u X)}{\pi u} = AX \left| \frac{\sin(\pi u X)}{\pi u X} \right|$$

2D Fourier Transform

The extension of Fourier transform from 1D to 2D is obvious. The 2D Fourier transform of $f(x, y)$ *can be written as*

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi(ux+vy)j} dxdy$$

The 2D inverse Fourier transform of $F(u, v)$ *is then*

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{2\pi(ux+vy)j} dudv$$

For a real function $f(x, y)$*, The Fourier transform* $F(u, v)$ *of* $f(x, y)$ *is in general* complex *which can be written as*

$$F(u, v) = R(u, v) + jI(u, v) = |F(u, v)|e^{-j\phi(u,v)}$$

where $R(u, v)$ *and* $I(u, v)$ *are the real and imaginary component, respectively.* $|F(u, v)|$ *is called the Fourier spectrum of* $f(x, y)$ *and* $\phi(u, v)$ *is called the phase angle of* $f(x, y)$*,* $|F(u, v)|^2$ *is called the* Fourier power spectrum *or* spectral density *of* $f(x, y)$*.*

Example: *Let* $f(x, y) = A$ *if* $0 \leq x \leq X, \quad 0 \leq y \leq Y$*, and* $f(x, y) = 0$ *otherwise. Then*

$$F(u, v) = \frac{A}{\pi uv} \sin(\pi uX) e^{-j\pi uX} \sin(\pi vY) e^{-j\pi vY}$$

$$|F(u, v)| = AXY \left|\frac{\sin(\pi uX)}{\pi uX}\right| \left|\frac{\sin(\pi vY)}{\pi vY}\right|$$

Discrete Fourier Transform (DFT)

For many applications, Discrete Fourier transform (DFT) may be more useful than the continuous FT. The DFT of $f(x)$, $0 \le x \le N-1$ *is defined as*

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-2\pi u x j/N} \quad for \quad u = 0, 1, \cdots, N-1.$$

The inverse DFT of $F(u)$, $0 \le u \le N-1$ *is defined as*

$$f(x) = \sum_{u=0}^{N-1} F(u) e^{2\pi u x j/N} \quad for \quad x = 0, 1, \cdots, N-1.$$

For 2D DFT, we have

$$F(u,v) = \frac{1}{NM} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-2\pi[\frac{ux}{M}+\frac{vy}{N}]j} \quad for \quad 0 \le u \le M-1,\ 0 \le v \le N-1$$

The inverse DFT of $F(u,v)$, $0 \le u \le M-1,\ 0 \le v \le N-1$, *is defined as*

$$f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{2\pi[\frac{ux}{M}+\frac{vy}{N}]j} \quad for \quad 0 \le x \le M-1,\ 0 \le v \le N-1$$

Example: *Let* $f(0)$*=0.5,* $f(1)$*=0.75,* $f(2)$*=1.00,* $f(3)$*=1.25. Then*

$$F(0) = 0.875,\ F(1) = -0.125 + 0.125j,\ F(2) = -0.125,\ F(3) = -0.125 + 0.125j.$$

Introduction to Fast Fourier Transform

Assume that $N = 2^n$ *and* $M = \frac{N}{2}$. *The DFT is defined as*

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x)\omega_N^{ux}, \quad where \;\; \omega_N = e^{-2\pi j/N}, \;\; \forall\, 0 \le u \le N$$

Then

$$F(u) = \frac{1}{2M} \sum_{x=0}^{2M-1} f(x)\omega_{2M}^{ux} = \frac{1}{2}\left[\frac{1}{M}\sum_{x=0}^{M-1} f(2x)\omega_{2M}^{u(2x)} + \frac{1}{M}\sum_{x=0}^{M-1} f(2x+1)\omega_{2M}^{u(2x+1)}\right]$$

Note that

$$\omega_{2M}^{2ux} = \omega_M^{ux}, \;\; \omega_M^{u+M} = \omega_M^u, \;\; and \;\; \omega_{2M}^{u+M} = -\omega_{2M}^u$$

Define

$$F_{even}(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(2x)\omega_M^{ux}, \quad F_{odd}(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(2x+1)\omega_M^{ux}$$

Thus

$$F(u) = \frac{1}{2}\left[F_{even}(u) + F_{odd}(u) * \omega_{2M}^u\right]$$

$$F(u+M) = \frac{1}{2}\left[F_{even}(u) - F_{odd}(u) * \omega_{2M}^u\right]$$

Computational Complexity of Fast Fourier Transform

To implement the above FFT, let

$$m(n) = \ \#\ of\ complex\ multiplications$$

$$a(n) = \ \#\ \ of\ complex\ additions/subtractions$$

Then

$$m(n) = 2m(n-1) + 2^{n-1}, \quad where \ \ m(0) = 0, \ \ m(1) = 1.$$

$$a(n) = 2a(n-1) + 2^{n}, \quad where \ \ a(0) = 0, \ \ a(1) = 2.$$

Then

$$\begin{aligned} m(n) &= 2m(n-1) + 2^{n-1} \\ &= 2[2m(n-2) + 2^{n-2}] + 2^{n-1} \\ &= \dots \\ &= 2^{n-1} \times m(1) + (n-1) \times 2^{n-1} \\ &= 2^{n-1} \times n \\ &= \tfrac{1}{2} N \log_2 N \end{aligned}$$

Similarly

$$a(n) = 2^n \log_2 N = N \log_2 N$$

Implement Inverse FFT by Forward FFT Algorithm

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-2\pi uxj/N} \quad for \quad 0 \leq u \leq N-1$$

$$f(x) = \sum_{u=0}^{N-1} F(u) e^{2\pi uxj/N} \quad for \quad 0 \leq x \leq N-1$$

$$f^*(x) = \sum_{u=0}^{N-1} F^*(u) e^{-2\pi uxj/N} \quad for \quad 0 \leq x \leq N-1$$

Then

$$\frac{1}{N} f^*(x) = \frac{1}{N} \sum_{u=0}^{N-1} F^*(u) e^{-2\pi uxj/N}$$

Similarly

$$\frac{1}{N^2} f^*(x,y) = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F^*(u,v) e^{-2\pi(ux+vy)j/N}$$

Discrete Cosine Transform (DCT)

The 1D DCT of $f(x)$, $0 \leq x \leq n-1$ *can be defined as*

$$C(0) = \frac{1}{\sqrt{n}} \sum_{x=0}^{n-1} f(x)$$

$$C(u) = \frac{\sqrt{2}}{\sqrt{n}} \sum_{x=0}^{n-1} f(x) \cos\left[\frac{(2x+1)u\pi}{2n}\right], \quad u = 1, 2, \cdots, n-1$$

The inverse DCT of $C(u)$, $0 \leq u \leq n-1$, *is then defined as*

$$f(x) = \frac{C(0)}{\sqrt{n}} + \frac{\sqrt{2}}{\sqrt{n}} \sum_{u=1}^{n-1} C(u) \cos\left[\frac{(2x+1)u\pi}{2n}\right], \quad x = 0, 1, \cdots, n-1$$

The n-point DCT Implementation could be done by applying 2n-point FFT.

$$C(0) = \frac{1}{\sqrt{n}} \sum_{x=0}^{n-1} f(x)$$

$$C(u) = \frac{\sqrt{2}}{\sqrt{n}} \times Real\left[e^{-\pi uj/2n} \cdot \sum_{x=0}^{2n-1} f(x) e^{-\pi uxj/n}\right]$$

where $u = 1, 2, \cdots, n-1$, $f(x) = 0 \ \ \forall\ x = n, n+1, \cdots, 2n-1$

2D Discrete Cosine Transform

The corresponding 2D DCT pair is defined as

$$C(u,v) = \alpha_u \alpha_v \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} f(x,y) \cos[\frac{(2x+1)u\pi}{2n}] \cos[\frac{(2y+1)v\pi}{2n}] \quad for \quad 0 \le u, v \le n-1$$

The inverse DCT of $C(u,v)$ *is defined as*

$$f(x,y) = \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} \alpha_u \alpha_v C(u,v) \cos[\frac{(2x+1)u\pi}{2n}] \cos[\frac{(2y+1)v\pi}{2n}] \quad for \quad 0 \le x, y \le n-1$$

where

$$\alpha_u = \begin{cases} \frac{1}{\sqrt{n}} & if \ u = 0, \\ \frac{\sqrt{2}}{\sqrt{n}} & if \ 1 \le u \le n-1 \end{cases}$$

In practice, $n = 8$ *is used, we have*

$$C(u,v) = \frac{\gamma_u \gamma_v}{4} \sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \cos[\frac{(2x+1)u\pi}{16}] \cos[\frac{(2y+1)v\pi}{16}]$$

$$f(x,y) = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} \gamma_u \gamma_v C(u,v) \cos[\frac{(2x+1)u\pi}{16}] \cos[\frac{(2y+1)v\pi}{16}]$$

where $\gamma_0 = \frac{1}{\sqrt{1}}$, *and* $\gamma_v = 1$ *if* $v = 1, 2, \cdots, n-1$

DCT in Matrix Forms

$Y = CXC^t$, where X is an 8×8 image block and $C = Q_8$ and Q_4 are orthogonal matrices defined below

$$Q_8 = \frac{1}{2}\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ cos\frac{\pi}{16} & cos\frac{3\pi}{16} & cos\frac{5\pi}{16} & cos\frac{7\pi}{16} & cos\frac{9\pi}{16} & cos\frac{11\pi}{16} & cos\frac{13\pi}{16} & cos\frac{15\pi}{16} \\ cos\frac{2\pi}{16} & cos\frac{6\pi}{16} & cos\frac{10\pi}{16} & cos\frac{14\pi}{16} & cos\frac{18\pi}{16} & cos\frac{22\pi}{16} & cos\frac{26\pi}{16} & cos\frac{30\pi}{16} \\ cos\frac{3\pi}{16} & cos\frac{9\pi}{16} & cos\frac{15\pi}{16} & cos\frac{21\pi}{16} & cos\frac{27\pi}{16} & cos\frac{33\pi}{16} & cos\frac{39\pi}{16} & cos\frac{45\pi}{16} \\ cos\frac{4\pi}{16} & cos\frac{12\pi}{16} & cos\frac{20\pi}{16} & cos\frac{28\pi}{16} & cos\frac{36\pi}{16} & cos\frac{44\pi}{16} & cos\frac{52\pi}{16} & cos\frac{60\pi}{16} \\ cos\frac{5\pi}{16} & cos\frac{15\pi}{16} & cos\frac{25\pi}{16} & cos\frac{35\pi}{16} & cos\frac{45\pi}{16} & cos\frac{55\pi}{16} & cos\frac{65\pi}{16} & cos\frac{75\pi}{16} \\ cos\frac{6\pi}{16} & cos\frac{18\pi}{16} & cos\frac{30\pi}{16} & cos\frac{42\pi}{16} & cos\frac{54\pi}{16} & cos\frac{66\pi}{16} & cos\frac{78\pi}{16} & cos\frac{90\pi}{16} \\ cos\frac{7\pi}{16} & cos\frac{21\pi}{16} & cos\frac{35\pi}{16} & cos\frac{49\pi}{16} & cos\frac{63\pi}{16} & cos\frac{77\pi}{16} & cos\frac{91\pi}{16} & cos\frac{105\pi}{16} \end{bmatrix}$$

$$Q_4 = \frac{1}{\sqrt{2}}\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ cos\frac{\pi}{8} & cos\frac{3\pi}{8} & cos\frac{5\pi}{8} & cos\frac{7\pi}{8} \\ cos\frac{2\pi}{8} & cos\frac{6\pi}{8} & cos\frac{10\pi}{8} & cos\frac{14\pi}{8} \\ cos\frac{3\pi}{8} & cos\frac{9\pi}{8} & cos\frac{15\pi}{8} & cos\frac{21\pi}{8} \end{bmatrix} \approx \frac{1}{26}\begin{bmatrix} 13 & 13 & 13 & 13 \\ 17 & 7 & -7 & -17 \\ 13 & -13 & -13 & 13 \\ 7 & -17 & 17 & -7 \end{bmatrix}$$

Quantization Table and An 8×8 Image Block

16	*11*	*10*	*16*	*24*	*40*	*51*	*61*
12	*12*	*14*	*19*	*26*	*58*	*60*	*55*
14	*13*	*16*	*24*	*40*	*57*	*69*	*56*
14	*17*	*22*	*29*	*51*	*87*	*80*	*62*
18	*22*	*37*	*56*	*68*	*109*	*103*	*77*
24	*35*	*55*	*64*	*81*	*104*	*113*	*92*
49	*64*	*78*	*87*	*103*	*121*	*120*	*101*
72	*92*	*95*	*98*	*112*	*100*	*103*	*99*

Table 1: **Quantization Table for DCT Coefficients**

139	*144*	*149*	*153*	*155*	*155*	*155*	*155*
144	*151*	*153*	*156*	*159*	*156*	*156*	*156*
150	*155*	*160*	*163*	*158*	*156*	*156*	*156*
159	*161*	*162*	*160*	*160*	*159*	*159*	*159*
159	*160*	*161*	*162*	*162*	*155*	*155*	*155*
161	*161*	*161*	*161*	*160*	*157*	*157*	*157*
162	*162*	*161*	*163*	*162*	*157*	*157*	*157*
162	*162*	*161*	*161*	*163*	*158*	*158*	*158*

Table 2: **An 8× Image Block X**

An 8×8 Image Block and Its DCT Coefficients

139	*144*	*149*	*153*	*155*	*155*	*155*	*155*
144	*151*	*153*	*156*	*159*	*156*	*156*	*156*
150	*155*	*160*	*163*	*158*	*156*	*156*	*156*
159	*161*	*162*	*160*	*160*	*159*	*159*	*159*
159	*160*	*161*	*162*	*162*	*155*	*155*	*155*
161	*161*	*161*	*161*	*160*	*157*	*157*	*157*
162	*162*	*161*	*163*	*162*	*157*	*157*	*157*
162	*162*	*161*	*161*	*163*	*158*	*158*	*158*

Table 3: **An 8× Image Block X**

235.6	*-1.0*	*-12.1*	*-5.2*	*2.1*	*-1.7*	*-2.7*	*1.3*
-22.6	*-17.5*	*-6.2*	*-3.2*	*-2.9*	*-0.1*	*0.4*	*-1.2*
-10.9	*-9.3*	*-1.6*	*1.5*	*0.2*	*-0.9*	*-0.6*	*-0.1*
-7.1	*-1.9*	*0.2*	*1.5*	*0.9*	*-0.1*	*-0.0*	*0.3*
-0.6	*-0.8*	*1.5*	*1.6*	*-0.1*	*-0.7*	*0.6*	*1.3*
1.8	*-0.2*	*1.6*	*-0.3*	*-0.8*	*1.5*	*1.0*	*-1.0*
-1.3	*-0.4*	*-0.3*	*-1.5*	*-0.5*	*1.7*	*1.1*	*-0.8*
-2.6	*1.6*	*-3.8*	*-1.8*	*1.9*	*1.2*	*-0.6*	*-0.4*

Table 4: **DCT Coefficients of Image Block X**

Representation of Quantized DCT Coefficients

235.6	*-1.0*	*-12.1*	*-5.2*	*2.1*	*-1.7*	*-2.7*	*1.3*
-22.6	*-17.5*	*-6.2*	*-3.2*	*-2.9*	*-0.1*	*0.4*	*-1.2*
-10.9	*-9.3*	*-1.6*	*1.5*	*0.2*	*-0.9*	*-0.6*	*-0.1*
-7.1	*-1.9*	*0.2*	*1.5*	*0.9*	*-0.1*	*-0.0*	*0.3*
-0.6	*-0.8*	*1.5*	*1.6*	*-0.1*	*-0.7*	*0.6*	*1.3*
1.8	*-0.2*	*1.6*	*-0.3*	*-0.8*	*1.5*	*1.0*	*-1.0*
-1.3	*-0.4*	*-0.3*	*-1.5*	*-0.5*	*1.7*	*1.1*	*-0.8*
-2.6	*1.6*	*-3.8*	*-1.8*	*1.9*	*1.2*	*-0.6*	*-0.4*

Table 5: **DCT Coefficients of Image Block X**

15	*0*	*-1*	*0*	*0*	*0*	*0*	*0*
-2	*-1*	*0*	*0*	*0*	*0*	*0*	*0*
-1	*-1*	*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	*0*	*0*	*0*	*0*	*0*	*0*	*0*
0	*0*	*0*	*0*	*0*	*0*	*0*	*0*

Table 6: **Quantized DCT Coefficients for Image Block X**

Suppose that the quantized DC *coefficient for the last block is* 12.

*(***S,2,3***), (1,2,-2), (0,1,-1), (0,1,-1), (0,1,-1), (2,1,-1), (0,1,-1) EOB*

DAUB4 Wavelet Transformation Matrix

$$c_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}, \quad c_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \quad c_2 = \frac{3-\sqrt{3}}{4\sqrt{2}}, \quad c_3 = \frac{1-\sqrt{3}}{4\sqrt{2}}$$

$$W = \begin{bmatrix}
c_0 & c_1 & c_2 & c_3 & & & & & & & & \\
c_3 & -c_2 & c_1 & -c_0 & & & & & & & & \\
 & & c_0 & c_1 & c_2 & c_3 & & & & & & \\
 & & c_3 & -c_2 & c_1 & -c_0 & & & & & & \\
 & & & & & & \ddots & & & & & \\
 & & & & & & & \ddots & & & & \\
 & & & & & & & & c_0 & c_1 & c_2 & c_3 \\
 & & & & & & & & c_3 & -c_2 & c_1 & -c_0 \\
c_2 & c_3 & & & & & & & & & c_0 & c_1 \\
c_1 & -c_0 & & & & & & & & & c_3 & -c_2
\end{bmatrix}$$

$$W = DAUB4, \quad {c_0}^2 \ + \ {c_1}^2 \ + \ {c_2}^2 \ + \ {c_3}^2 \ = \ 1.$$

$$Y \ \leftarrow \ P \ *_4 \ W \ *_3 \ X \ *_1 \ W^t \ *_2 \ Q$$

A 3-scale Daubechies' four transform of an 8×8 image

64	90	83	88	127	174	174	179
67	93	89	82	125	175	180	178
65	79	79	69	127	169	180	181
59	74	91	62	117	170	182	181
57	82	98	57	102	162	180	180
59	75	104	58	90	163	187	183
59	79	105	67	74	159	189	182
61	76	100	67	65	131	188	186

935	-316	61	-84	2	-11	10	-73
21	-3	0	-75	24	-9	15	-71
-4	5	11	3	46	-39	25	-76
29	7	18	0	36	-52	37	-74
-2	-4	0	2	-6	7	-1	3
4	-2	-4	-5	2	1	-2	-4
2	2	10	-1	1	-10	0	-1
-1	7	18	-6	-14	12	-11	0

Haar Wavelet Transformation Matrix

$$c_0 = 1/\sqrt{2},\ c_1 = 1/\sqrt{2}.$$

$$H = \begin{bmatrix}
c_0 & c_1 & & & & & & & & \\
c_0 & -c_1 & & & & & & & & \\
 & & c_0 & c_1 & & & & & & \\
 & & c_0 & -c_1 & & & & & & \\
 & & & & \ddots & & & & & \\
 & & & & & \ddots & & & & \\
 & & & & & & c_0 & c_1 & & \\
 & & & & & & c_0 & -c_1 & & \\
 & & & & & & & & c_0 & c_1 \\
 & & & & & & & & c_0 & -c_1
\end{bmatrix}$$

$$Y \leftarrow P *_4 H *_3 X *_1 H^t *_2 Q$$

Example of 3-scale Haar transform of an 8×8 image

64	90	83	88	127	174	174	179
67	93	89	82	125	175	180	178
65	79	79	69	127	169	180	181
59	74	91	62	117	170	182	181
57	82	98	57	102	162	180	180
59	75	104	58	90	163	187	183
59	79	105	67	74	159	189	182
61	76	100	67	65	131	188	186

926	-325	-12	-62	-26	0	-48	-1
26	-19	-26	-131	-14	19	-47	0
19	1	0	7	-20	43	-66	2
-4	18	4	24	-17	35	-75	4
-2	0	0	-2	0	-5	0	-2
4	-2	4	0	0	-8	4	0
2	-2	5	-4	-3	-2	5	-1
0	2	17	-1	-2	2	-9	1

Singular Value Decomposition (SVD) + PCP

Spectrum Decomposition Theorem: $A = W\Lambda W^t$

Every symmetric matrix can be diagonalized.

SVD Theorem: $A = U\Sigma V^t = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^t$

Each matrix $A \in R^{m\times n}$ can be decomposed as $A = U\Sigma V^t$, where both $U \in R^{m\times m}$ and $V \in R^{n\times n}$ are orthogonal. Moreover, $\Sigma \in R^{m\times n} = diag[\sigma_1, \sigma_2, \ldots, \sigma_k, 0, \ldots, 0]$ is essentially diagonal with the singular values satisfying $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k > 0$, and $\sigma_j = 0 \ \forall \ j \geq (k+1)$.

Example:

$$A = \begin{bmatrix} -2 & -2 & 6 & 6 \\ -2 & 2 & -6 & -6 \\ 6 & -6 & 5 & -1 \\ 6 & -6 & -1 & 5 \end{bmatrix}$$

Then $\Sigma(A) = diag[16, 8, 6, 0]$, $\lambda(A) = \{16, -8, 6, 0\}$.

$$U = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & -\frac{1}{\sqrt{2}} \\ -\frac{1}{2} & -\frac{1}{2} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{2} & -\frac{1}{2} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix}, \quad V = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix}$$

$$W = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{2} & \frac{1}{2} & 0 & -\frac{1}{\sqrt{2}} \\ -\frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{2} & \frac{1}{2} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix}$$

Gabor Filter(s) and Transform(s)

A 2-dimensional (symmetrical) gabor filter can be defined as

$$f(x,y) = \exp\left(-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right) \times \cos[2\pi\mu_0(x\cos\theta + y\sin\theta)] \tag{1}$$

where μ_0 is the frequency of a sinusoidal plane and σ_x and σ_y are corresponding to the variance of the Gaussian distribution (shape). Simple computations lead to the Fourier transform which possesses a similar form as the gabor filter has.

$$\begin{aligned} F(u,v) &= \pi\sigma_x\sigma_y \left\{\exp\left(-\tfrac{1}{2}\left[\tfrac{(u-\mu_0\cos\theta)^2}{\sigma_u^2} + \tfrac{(v-\mu_0\sin\theta)^2}{\sigma_v^2}\right]\right)\right. \\ &+ \left.\exp\left(-\tfrac{1}{2}\left[\tfrac{(u+\mu_0\cos\theta)^2}{\sigma_u^2} + \tfrac{(v+\mu_0\sin\theta)^2}{\sigma_v^2}\right]\right)\right. \end{aligned}$$

where $\sigma_u = \frac{1}{2\pi\sigma_x}$, $\sigma_v = \frac{1}{2\pi\sigma_y}$.

$$I[x,y] * F_j[x,y] = InvFT\{FT(I[x,y]) \times FT(F_j[x,y])\}$$