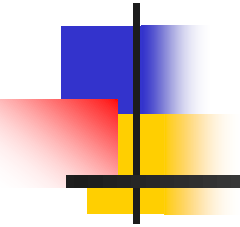


Transformasi Citra





Pendahuluan

- Mengapa perlu transformasi ?
 - Setiap orang pada suatu saat pernah menggunakan suatu teknik analisis dengan transformasi untuk menyederhanakan penyelesaian suatu masalah [Brigham,1974]
 - Contoh: penyelesaian fungsi $y = x/z$
 - Analisa konvensional : pembagian secara manual
 - Analisa transformasi : melakukan transformasi
 - $\log(y) = \log(x) - \log(z)$



Pendahuluan

- Transformasi juga diperlukan bila kita ingin mengetahui suatu informasi tertentu yang tidak tersedia sebelumnya
- Contoh :
 - jika ingin mengetahui informasi frekuensi kita memerlukan transformasi Fourier
 - Jika ingin mengetahui informasi tentang kombinasi skala dan frekuensi kita memerlukan transformasi wavelet



Transformasi Citra

- Transformasi citra, sesuai namanya, merupakan proses *perubahan bentuk* citra untuk mendapatkan suatu informasi tertentu
- Transformasi bisa dibagi menjadi 2 :
 - Transformasi piksel/transformasi geometris:
 - Transformasi ruang/domain/space



Transformasi Piksel

- Transformasi piksel masih bermain di ruang/domain yang sama (domain spasial), hanya posisi piksel yang kadang diubah
- Contoh: rotasi, translasi, scaling, invers, shear, dll.
- Transformasi jenis ini relatif mudah diimplementasikan dan banyak aplikasi yang dapat melakukannya (Paint, ACDSee, dll)



Transformasi Ruang/domain

- Transformasi ruang merupakan proses perubahan citra dari suatu ruang/domain ke ruang/domain lainnya, contoh: dari ruang spasial ke ruang frekuensi
- Masih ingat istilah 'ruang' ? Ingat-ingat kembali pelajaran Aljabar Linier tentang Basis dan Ruang 😊
 - Contoh : Ruang vektor. Salah satu basis yang merentang ruang vektor 2 dimensi adalah $[1 \ 0]$ dan $[0 \ 1]$. Artinya, semua vektor yang mungkin ada di ruang vektor 2 dimensi selalu dapat direpresentasikan sebagai kombinasi linier dari basis tersebut.



Transformasi Ruang

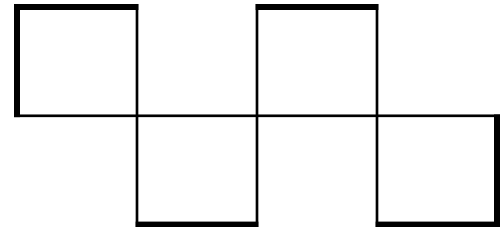
- Ada beberapa transformasi ruang yang akan kita pelajari, yaitu :
 - Transformasi Fourier (basis: cos-sin)
 - Transformasi Hadamard/Walsh (basis: kolom dan baris yang ortogonal)
 - Transformasi DCT (basis: cos)
 - Transformasi Wavelet (basis: scaling function dan mother wavelet)

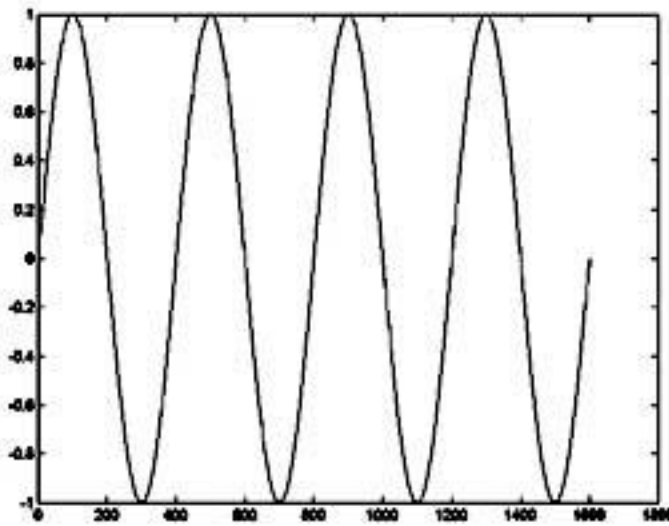


Transformasi Fourier (FT)

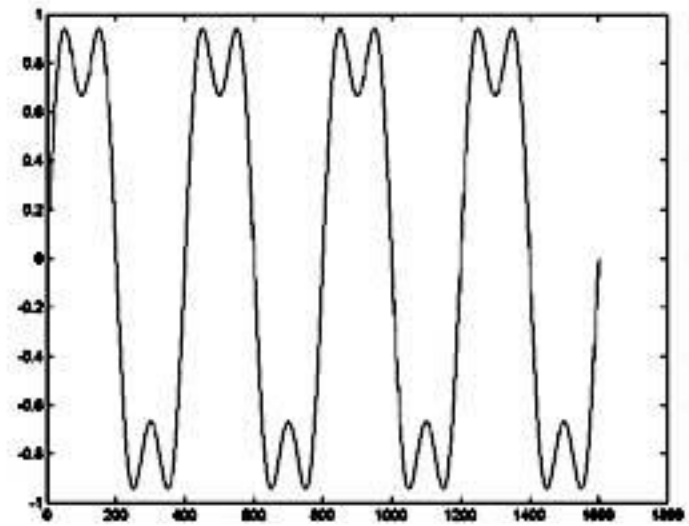
- Pada tahun 1822, Joseph Fourier, ahli matematika dari Prancis menemukan bahwa: *setiap fungsi periodik (sinyal) dapat dibentuk dari penjumlahan gelombang-gelombang sinus/cosinus.*
- Contoh : Sinyal kotal merupakan penjumlahan dari fungsi-fungsi sinus berikut (lihat gambar pada halaman berikut)

$$f(x) = \sin(x) + \sin(3x)/3 + \sin(5x)/5 + \sin(7x)/7 + \sin(9x)/9 \dots$$

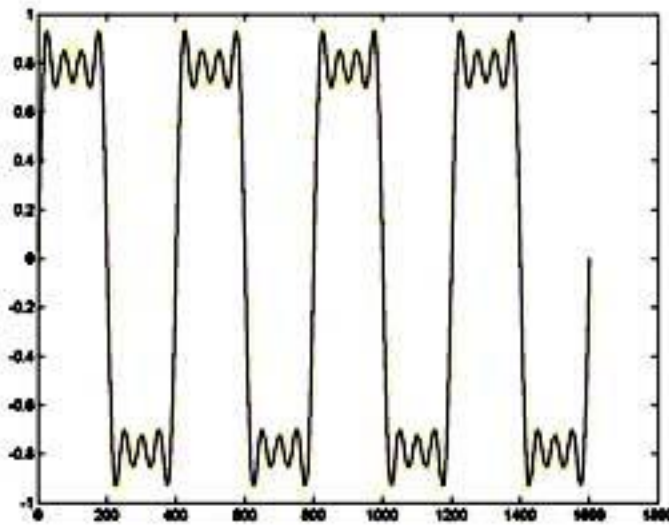




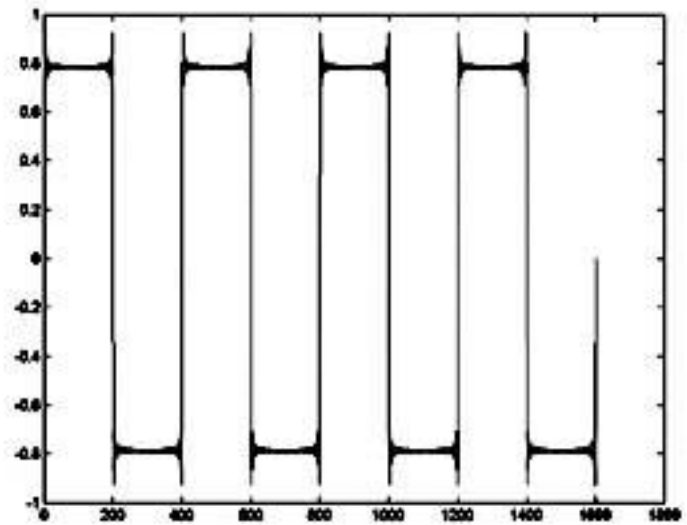
(a)



(b)



(c)



(d)

Gambar a) $n = 1$, b) $n = 3$, c) $n = 7$, d) $n = 99$



FT - Motivasi

- Jika semua sinyal periodik dapat dinyatakan dalam penjumlahan fungsi-fungsi sinus-cosinus, pertanyaan berikutnya yang muncul adalah:
 - Jika saya memiliki sebuah sinyal sembarang, bagaimana saya tahu fungsi-fungsi \cos – \sin apa yang membentuknya ?
- Atau dengan kata lain
 - Berapakah frekuensi yang dominan di sinyal tersebut ?
- Pertanyaan di atas dapat dijawab dengan menghitung nilai $F(u)$ dari sinyal tersebut. Dari nilai $F(u)$ kemudian dapat diperoleh kembali sinyal awal dengan menghitung $f(x)$, menggunakan rumus:



Rumus FT – 1 dimensi

- Rumus FT kontinu 1 dimensi

$$F(u) = \int_{-\infty}^{\infty} f(x) \exp[-2j\pi ux] dx$$

$$f(x) = \int_{-\infty}^{\infty} F(u) \exp[2j\pi ux] du$$

Euler's formula : $\exp[-2j\pi ux] = \cos 2\pi ux - j \sin 2\pi ux$

- Rumus FT diskret 1 dimensi

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp[-2j\pi ux / N]$$

$$f(x) = \frac{1}{N} \sum_{u=0}^{N-1} F(u) \exp[2j\pi ux / N]$$



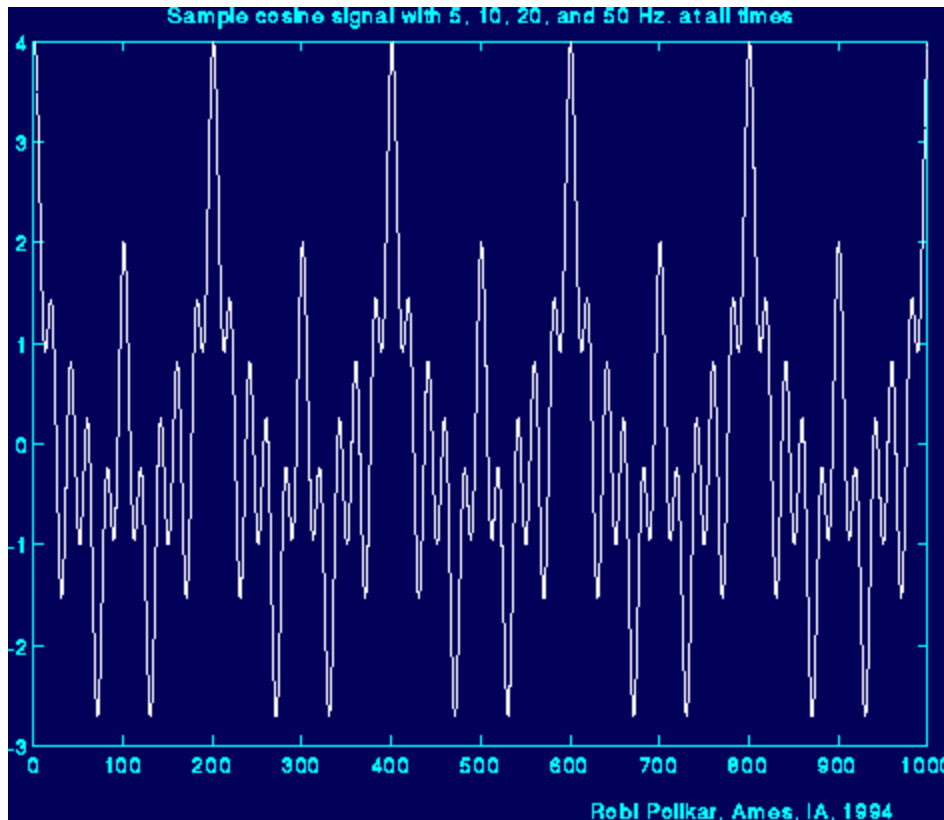
Contoh FT 1 dimensi

Misalkan kita memiliki sinyal $x(t)$ dengan rumus sbb:

$$x(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 10 \cdot t) + \\ \cos(2\pi \cdot 20 \cdot t) + \cos(2\pi \cdot 50 \cdot t)$$

Sinyal ini memiliki empat komponen frekuensi yaitu
5,10,20,50

Contoh sinyal 1 Dimensi $x(t)$

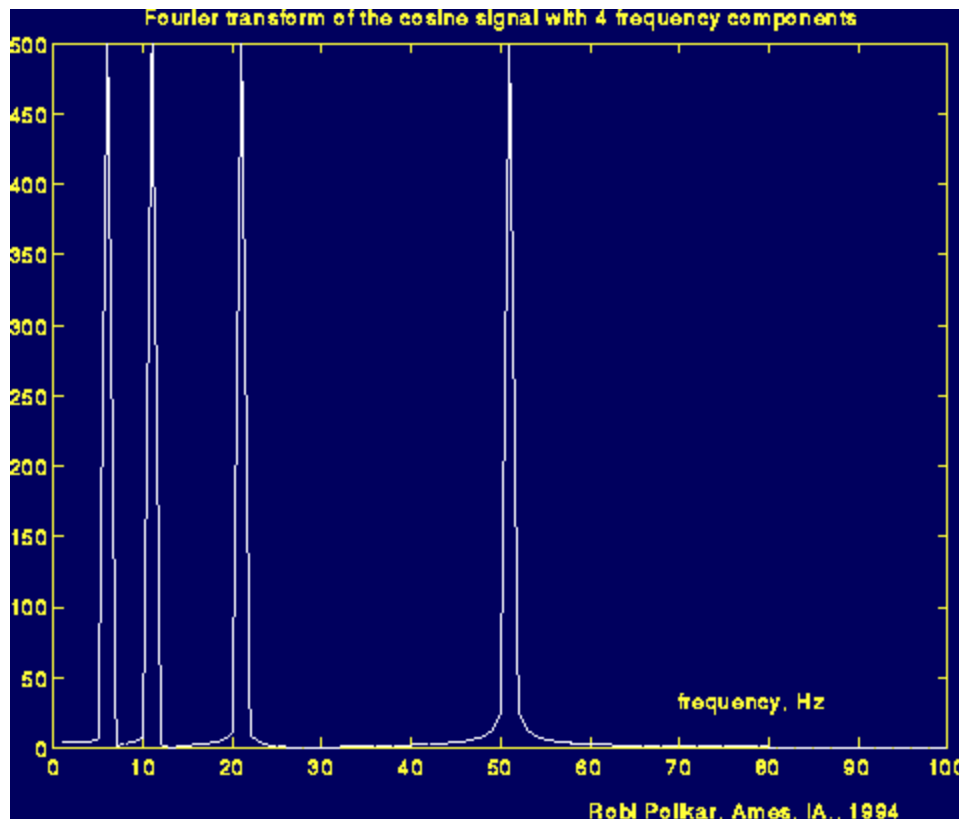


Gambar sinyal satu dimensi dengan rumus

$$x(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 10 \cdot t) + \cos(2\pi \cdot 20 \cdot t) + \cos(2\pi \cdot 50 \cdot t)$$

(Sumber: Polikar)

FT dari sinyal tersebut



FT dari sinyal tersebut. Terlihat bahwa FT dapat menangkap frekuensi-frekuensi yang dominan dalam sinyal tersebut, yaitu 5, 10, 20, 50

(nilai maksimum $F(u)$ berada pada angka 5, 10, 20, 50)



Contoh Penghitungan FT 1 dimensi

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp[-2j\pi ux / N] = \frac{1}{N} \sum_{x=0}^{N-1} f(x)(\cos(2\pi ux / N) - j \sin(2\pi ux / N))$$

contoh: $f(0) = 2, f(1) = 3, f(2) = 4, f(3) = 4$

$$\begin{aligned} F(0) &= \frac{1}{N} \sum_{x=0}^{N-1} f(x)(\cos(2\pi 0x / N) - j \sin(2\pi 0x / N)) \\ &= \frac{1}{4} [f(0) + f(1) + f(2) + f(3)] = 3.25 \end{aligned}$$

$$\begin{aligned} F(1) &= \frac{1}{4} \sum_{x=0}^3 f(x)(\cos(2\pi x / 4) - j \sin(2\pi x / 4)) \\ &= \frac{1}{4} [2(1-0) + 3(0-j) + 4(-1-0) + 4(0+j)] \\ &= \frac{1}{4} (2-3j-4+4j) = \frac{1}{4} (-2+j) = -0.5+0.25j \end{aligned}$$

$$F(2) = -\frac{1}{4} [1] = -0.25 \quad F(3) = -\frac{1}{4} [2+j] = -0.5-0.25j$$



Contoh Penghitungan FT

- Hasil penghitungan FT biasanya mengandung bilangan real dan imajiner
 - Fourier Spectrum didapatkan dari magnitude kedua bilangan tersebut shg $|F(u)| = [R^2(u) + I^2(u)]^{1/2}$
 - Untuk contoh di halaman sebelumnya, Fourier Spectrumnya adalah sebagai berikut:
-
- $|F(0)| = 3.25$ $|F(1)| = [(-0.5)^2 + (0.25)^2]^{1/2} = 0.5590$
 - $|F(2)| = 0.25$ $|F(3)| = [(0.5)^2 + (0.25)^2]^{1/2} = 0.5590$



Rumus FT – 2 dimensi

- Rumus FT 2 dimensi

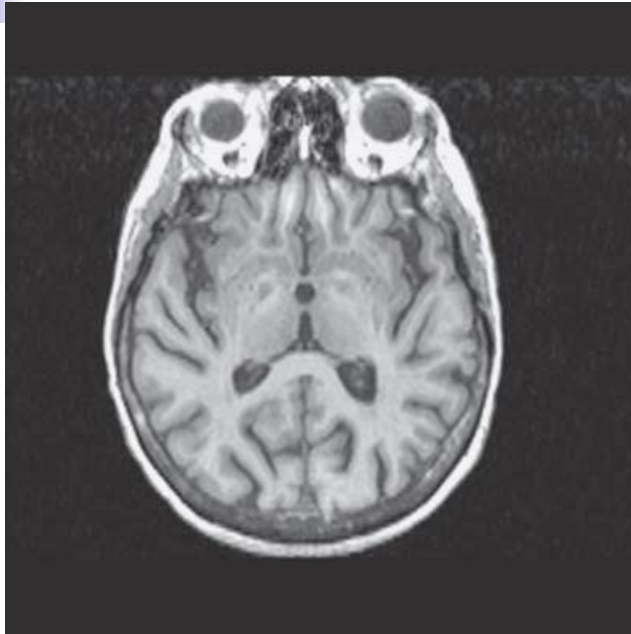
$$FT : F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp[-2j\pi(ux/M + vy/N)]$$

$$InversFT : f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp[2j\pi(ux/M + vy/N)]$$

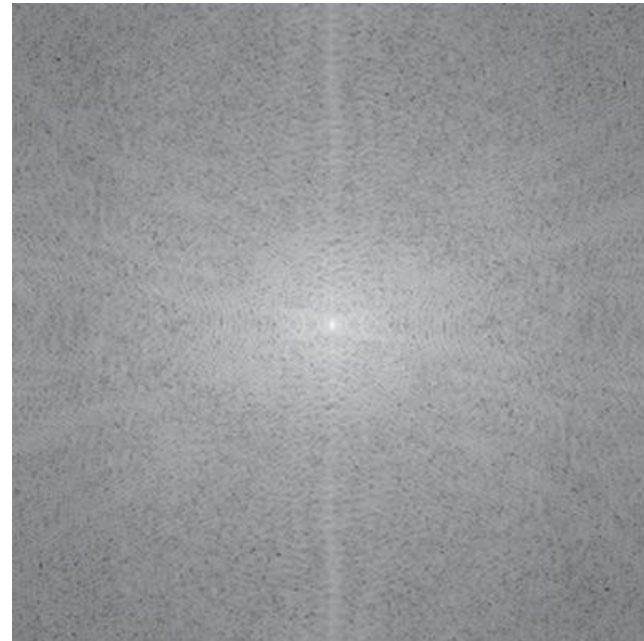
M = tinggi citra (jumlah baris)

N = lebar citra (jumlah kolom)

Contoh FT 2 Dimensi

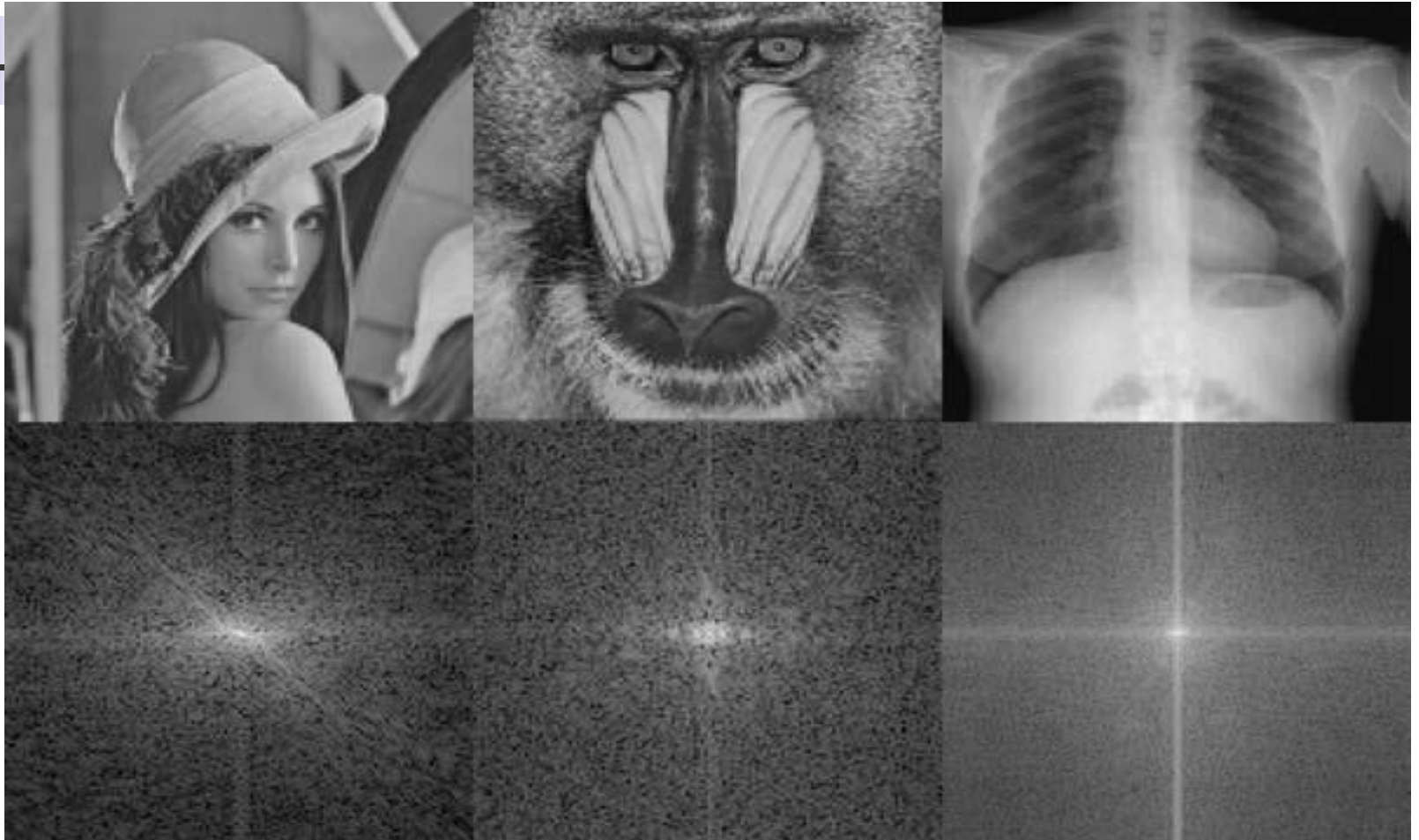


(i) MRI image;



(ii) Fourier transform of (i)

Contoh FT 2 Dimensi





Fast Fourier Transform (FFT)

- Merupakan algoritma penghitungan yang mengurangi kompleksitas FT biasa dari N^2 menjadi $N \log_2 N$ saja
- Pada implementasinya, FFT merupakan cara yang umum digunakan untuk menghitung FT diskret
- InversFT juga dapat dihitung dengan kompleksitas $N \log_2 N$ (IFFT)
 - Di Matlab : `fft(x)` atau `fft2(X)` untuk FT dan `ifft(x)` atau `ifft2(X)` untuk invers FT



Transformasi Walsh

- Jika FT berdasarkan pada basis fungsi trigonometri (sin-cos), maka Tr. Walsh berdasarkan pada fungsi basis yang nilainya $+1$ dan -1
- Kompleksitas algoritma Tr. Walsh juga dapat diefisienkan menjadi $N \log_2 N$



Rumus Tr. Walsh

■ Rumus Tr. Walsh 1 dimensi

Fungsi basis/kernel
 $g(x,u)$

$$W(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u)}$$

Dengan $u = 0, 1, 2, \dots, N-1$ dan $x = 0, 1, 2, \dots, N-1$, sedangkan nilai n mengikuti aturan :

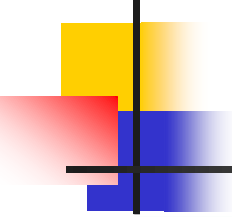
$$N = 2^n$$

$$f(x) = \frac{1}{N} \sum_{u=0}^{N-1} W(u) \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u)}$$

$b_i(x)$ adalah bit ke- i dari representasi biner x .

Contoh : $n = 3$, $x = 4$ (100) maka $b_0(x) = 0$, $b_1(x) = 0$, $b_2(x) = 1$

Nilai kernel Tr. Walsh 1-D untuk $N=8$



$u \backslash x$	0	1	2	3	4	5	6	7
0	1	1	1	1	1	1	1	1
1	1	1	1	1	-1	-1	-1	-1
2	1							
3	1							
4	1							
5	1							
6	1							
7								

Hasil Tr. Walsh dari citra


$$f(x) = (10 \ 10 \ 10 \ 10 \ 20 \ 20 \ 20 \ 20)$$

- $W(u=0) = (10 + 10 + 10 + 10 + 20 + 20 + 20 + 20) / 8 = 120 / 8 = 15$
- $W(u=1) = -5$
-
-

$$W(u) = (15 \ -5 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$$



Rumus Tr. Walsh

- Rumus Tr. Walsh 2 dimensi:

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

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

Transformasi Walsh

Jika digambarkan secara visual, maka untuk $N = 4$, bentuk basisnya dapat dilihat seperti gambar disamping.

Karena rumus forward dan invers-nya sama, maka basis ini dapat dipakai baik untuk forward maupun invers transform

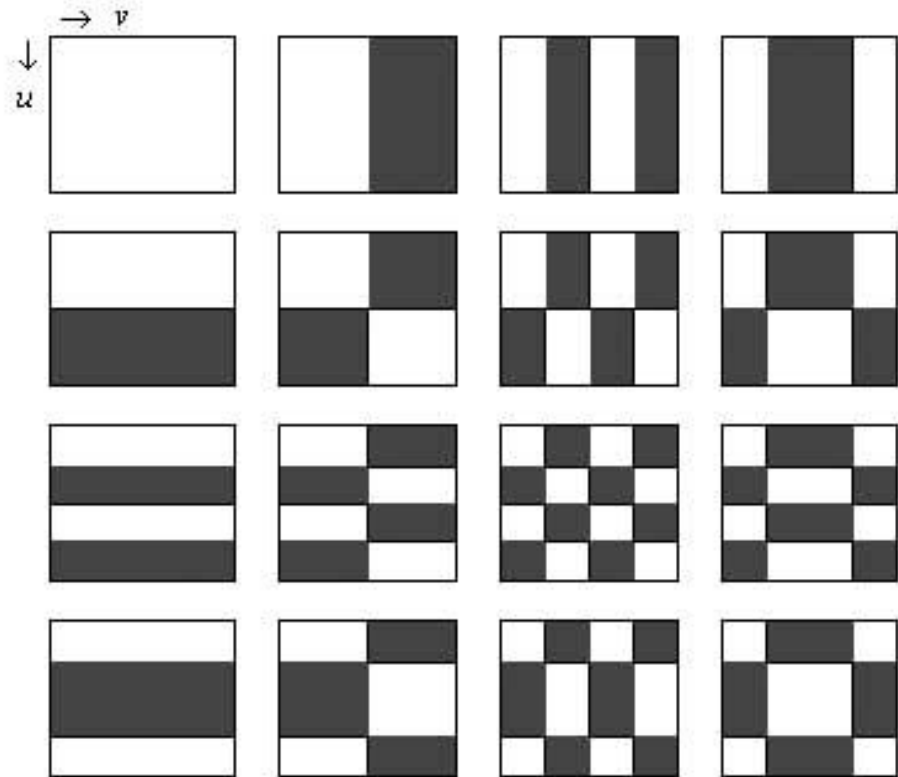


Figure 3.25 Walsh basis function for $N = 4$. Each block consist of 4×4 elements, corresponding to x and y varying from 0 to 3. The origin of each block is at its top left. White and black denote +1 and -1 respectively. (Gonzalez, 1993)



Hasil Tr. Walsh pada citra

2	2	10	10
2	2	10	10
20	20	4	4
20	20	4	4

$$W(0,0)=(2+2+10+10+2+2+10+10+20+20+4+4+20+20+4+4)/4 = 144/4=36$$

$$W(0,1)=(2+2-10-10+2+2-10-10+20+20-4-4+20+20-4-4)/4 = 32/4=8$$

36	8	0	0
-12	-24	0	0
0	0	0	0
0	0	0	0



Transformasi Hadamard

- Rumus Tr. Hadamard untuk 2 dimensi:

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

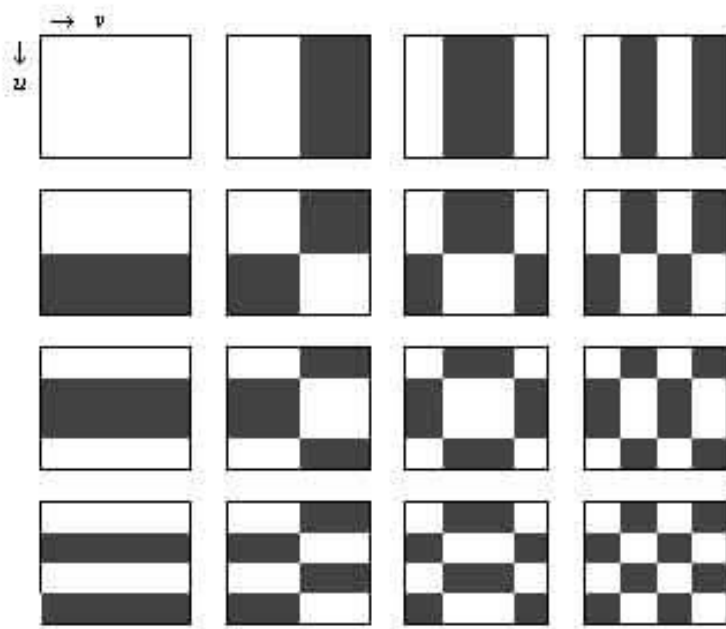
$$f(x, y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} H(u, v) (-1)^{\sum_{i=0}^{n-1} [b_i(x)b_i(u) + b_i(y)b_i(v)]}$$

$b_k(z)$ adalah bit ke- k dari representasi biner z .

Contoh : $n = 3, z = 6$ (110) maka $b_0(z) = 0, b_1(z) = 1, b_2(z) = 1$

Tr. Hadamard

- Jika digambarkan secara visual, untuk $N=4$, nilai $(-1)^{(\dots)}$ dapat dilihat sbb:





Transformasi Hadamard

- Jika basis FT adalah fungsi cos-sin, maka basis dari transformasi Hadamard adalah kolom dan baris yang ortogonal
- Ilustrasi : input citra 4x4

100	100	50	50
100	100	50	50
50	50	100	100
50	50	100	100



Contoh Tr. Hadamard

- Untuk memperoleh transformasinya, kalikan basis dengan citra input (putih untuk +, hitam untuk -). Satu posisi pada $H(u,v)$ hanya menggunakan satu blok.
- $H(0,0) =$
 $(100+100+50+50+100+100+50+50+50+50+100+100+50+50+100+100)/4 = 1200/4 = 300$
- $H(0,1) = (100+100-50-50+100+100-50-50+50+50-100-100+50+50-100-100)/4 = 0$
- $H(0,2) = (100-100-50+50+100-100-50+50+50-50-100+100+50-50-100+100)/4 = 0$
- $H(0,3) = (100-100+50-50+100-100+50-50+50-50+100-100+50-50+100-100)/4 = 0$



Contoh Tr. Hadamard

- $H(1,0) = (\dots)/4 = 0$
- $H(1,1) = (\dots)/4 = 400/4 = 100$
- $H(1,2) = (\dots)/4 = 0$
- $H(1,3) = (\dots)/4 = 0$

300	0	0	0
0	100	0	0
0	0	0	0
0	0	0	0

- $H(2,0) = (\dots)/4 = 0$
- $H(2,1) = (\dots)/4 = 0$
- $H(2,2) = (\dots)/4 = 0$
- $H(2,3) = (\dots)/4 = 0$

- $H(3,0) = (\dots)/4 = 0$
- $H(3,1) = (\dots)/4 = 0$
- $H(3,2) = (\dots)/4 = 0$
- $H(3,3) = (\dots)/4 = 0$

Perhatikan bahwa nilainya besar hanya pada koordinat (0,0) dan (1,1). Nilainya pada H(1,1) besar karena polanya sama dengan citra input.

Perhatikan juga bahwa jika kita hanya perlu menyimpan nilai yang bukan nol, maka representasi citra yang kita miliki juga menjadi sangat kecil (dapat dikompresi).



Contoh Tr. Hadamard

- Dari citra hasil transformasi, diperoleh gambar asal (dengan melihat kembali pada basis, satu posisi $f(x,y)$ menggunakan semua blok pada posisi tertentu (x,y)).

$$f(0,0) =$$

$$(300+0+0+0+0+0+100+0+0+0+0+0+0+0+0+0)/4 = 400/4 = 100$$

$$f(0,1) = (300\ldots+100\ldots)/4 = 400/4 = 100$$

$$f(0,2) = (300\ldots-100\ldots)/4 = 200/4 = 50$$

$$f(0,3) = (300\ldots-100\ldots)/4 = 200/4 = 50$$

$$f(1,0) = (300\ldots+100\ldots)/4 = 400/4 = 100$$

$$f(1,1) = (300\ldots+100\ldots)/4 = 400/4 = 100$$

$$f(1,2) = (300\ldots-100\ldots)/4 = 200/4 = 50$$

$$f(1,3) = (300\ldots-100\ldots)/4 = 200/4 = 50$$



Contoh Tr. Hadamard

$$f(2,0) = (300.....-100.....)/4 = 200/4 = 50$$

$$f(2,1) = (300.....-100.....)/4 = 200/4 = 50$$

$$f(2,2) = (300.....+100.....)/4 = 400/4 = 100$$

$$f(2,3) = (300.....+100.....)/4 = 400/4 = 100$$

$$f(3,0) = (300.....-100.....)/4 = 200/4 = 50$$

$$f(3,1) = (300.....-100.....)/4 = 200/4 = 50$$

$$f(3,2) = (300.....+100.....)/4 = 400/4 = 100$$

$$f(3,3) = (300.....+100.....)/4 = 400/4 = 100$$

- Citra rekonstruksi yang dihasilkan persis dengan citra awal

Transformasi Kosinus Diskret (DCT)

- Rumus Discrete Cosine Transform (DCT) untuk 2 dimensi :

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2x+1)v\pi}{2N}\right]$$

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u)\alpha(v) C(u, v) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2x+1)v\pi}{2N}\right]$$

$$u, v = 0, 1, 2, \dots, N-1$$

$$\alpha(u) = \sqrt{\frac{1}{N}} \quad \text{for } u = 0, \quad \alpha(u) = \sqrt{\frac{2}{N}} \quad \text{for } u = 1, 2, 3, \dots, N-1$$

DCT – contoh basis untuk $N=4$

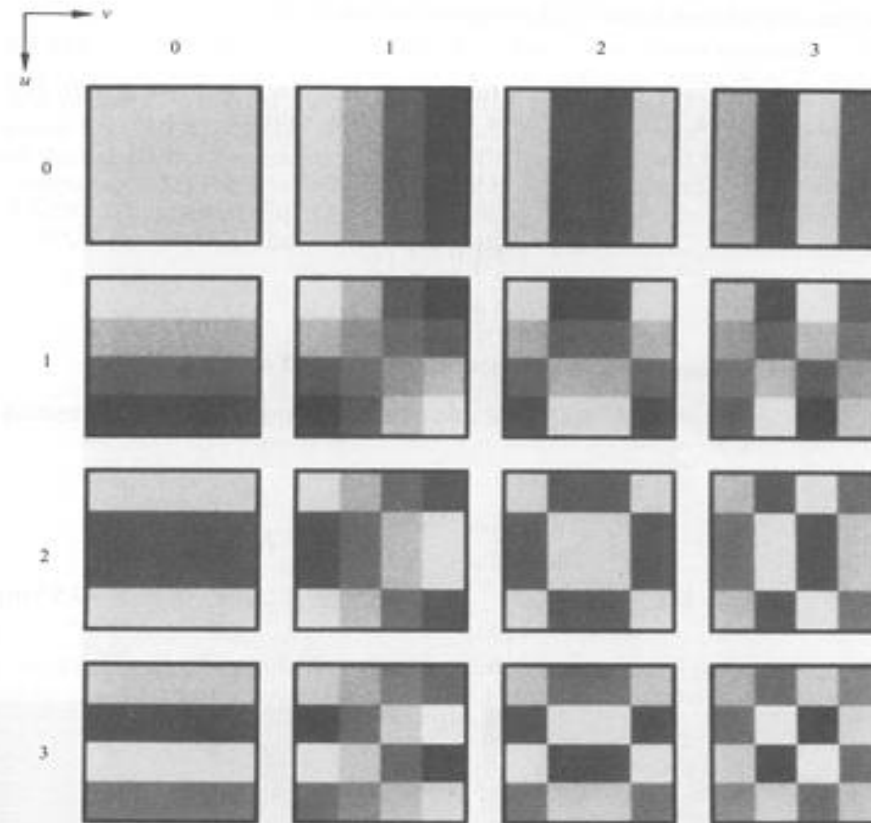



Figure 3.28 Discrete cosine transform basis functions for $N = 4$. Each block consists of 4×4 elements, corresponding to x and y varying from 0 to 3. The origin of each block is at its top left. The highest value is shown in white. Other values are shown in grays, with darker meaning smaller.

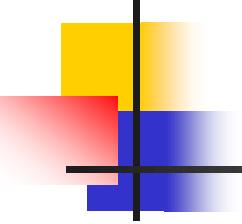


Transformasi Wavelet



Pada tahun 1987, wavelet telah menjadi pendekatan baru yang sangat baik dalam pemrosesan dan analisis sinyal.

- Wavelet adalah fungsi matematika yang memotong data menjadi kumpulan-kumpulan frekuensi yang berbeda, sehingga masing masing komponen tersebut dapat dipelajari dengan menggunakan skala resolusi yang berbeda.

- 
-
- Transformasi wavelet menggunakan dua komponen penting dalam melakukan transformasi yakni fungsi skala (*scaling function*) dan fungsi wavelet (*wavelet function*).
 - Fungsi skala (*scaling function*) disebut juga sebagai Lowpass Filter, sedangkan fungsi wavelet (*wavelet function*) disebut juga sebagai Highpass Filter.
 - Kedua fungsi ini digunakan pada saat transformasi wavelet dan inverse transformasi wavelet.

1. Fungsi wavelet

disebut juga *highpass filter* yang mengambil citra dengan gradiasi intensitas yang tinggi dan perbedaan intensitas yang rendah akan dikurangi atau dibuang.

2. Fungsi skala

disebut juga *lowpass filter* yang mengambil citra dengan gradiasi intensitas yang halus dan perbedaan intensitas yang tinggi akan dikurangi atau dibuang.

Kedua komponen diatas dapat disebut sebagai *mother wavelet* yang harus memenuhi kondisi :

$$\int_{-\infty}^{\infty} \psi(x) dx = 0$$

yang menjamin terjaminnya sifat ortogonalitas vektor.

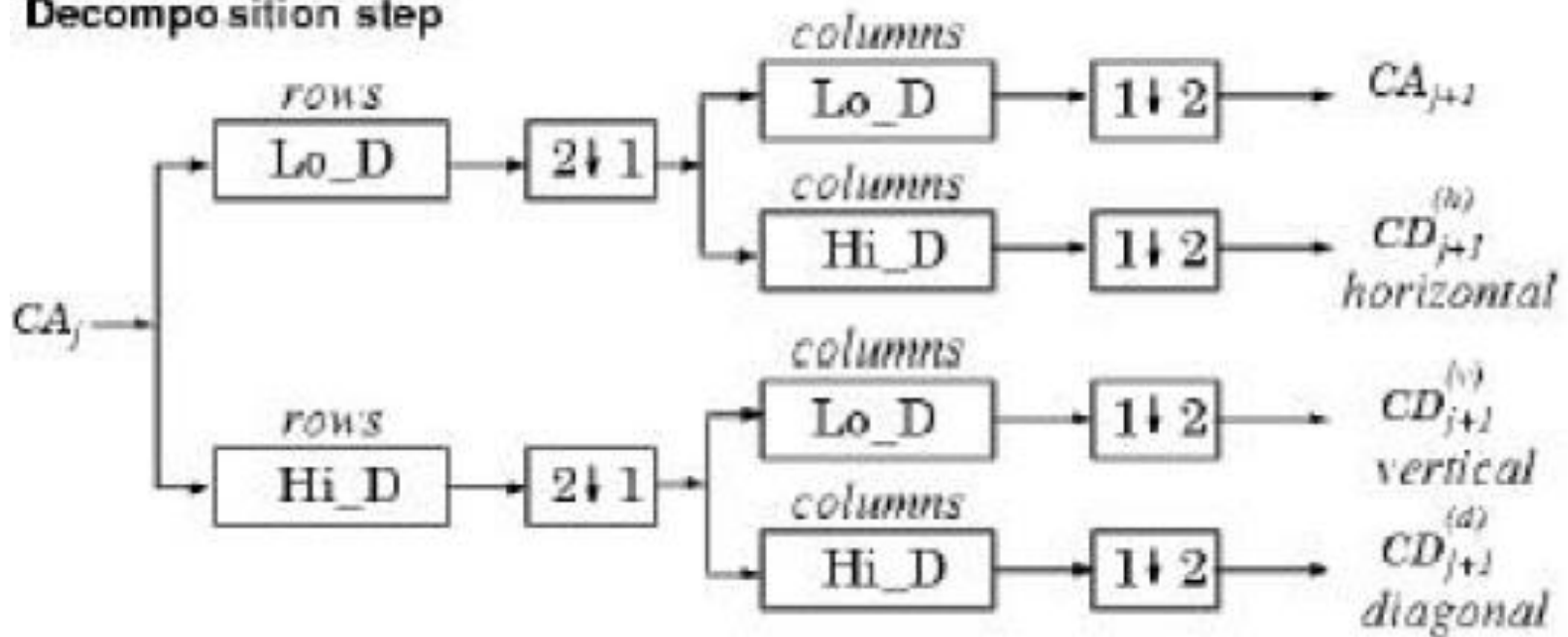
Transformasi Wavelet Diskrit




- Transformasi wavelet diskrit secara umum merupakan dekomposisi citra pada frekuensi *subband citra tersebut dimana komponennya dihasilkan dengan cara penurunan level* dekomposisi.
- Implementasi transformasi wavelet diskrit dapat dilakukan dengan cara melewatkan sinyal frekuensi tinggi atau *highpass filter* dan frekuensi rendah atau *lowpass filter*.

Transformasi wavelet diskrit dua dimensi dengan level dekomposisi satu

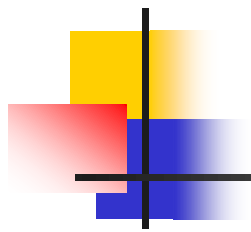
Decomposition step



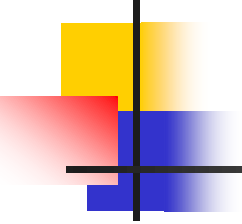


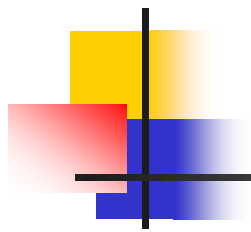
Seperti yang terlihat pada Gambar diatas , jika suatu citra dilakukan proses transformasi wavelet diskrit dua dimensi dengan level dekomposisi satu, maka akan menghasilkan empat buah *subband*, yaitu :

1. Koefisien Aproksimasi (CA_{j+1}) atau disebut juga *subband LL*
2. Koefisien Detil Horisontal ($CD(h)_{j+1}$) atau disebut juga *subband HL*
3. Koefisien Detil Vertikal ($CD(v)_{j+1}$) atau disebut juga *subband LH*
4. Koefisien Detil Diagonal ($CD(d)_{j+1}$) atau disebut juga *subband HH*

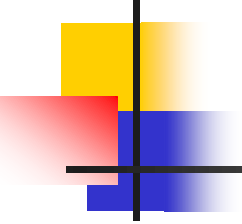


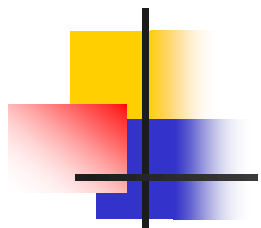
LL	HL
LH	HH

- 
-
- dengan Level Dekomposisi 1 *Subband hasil dari dekomposisi dapat didekomposisi lagi*
 - karena level dekomposisi wavelet bernilai dari 1 sampai n atau disebut juga transformasi
 - wavelet multilevel. Jika dilakukan dekomposisi lagi, maka *subband LL yang akan*
 - didekomposisi karena *subband LL berisi sebagian besar dari informasi citra. Jika*
 - dilakukan dekomposisi dengan level dekomposisi dua maka *subband LL akan*
 - menghasilkan empat buah *subband baru, yaitu subband LL2 (Koefisien Aproksimasi 2),*
 - HL2 (Koefisien Detil Horisontal 2), LH2 (Koefisien Detil Vertikal 2), dan HH2
 - (Koefisien Detil Diagonal 2). Dan begitu juga seterusnya jika dilakukan dekomposisi lagi.



LL2	HL2	HL1
LH2	HH2	
LH1		HH1

- 
-
- Bila citra asli f dengan $M \times N$ pixel didekomposisi menjadi empat subband sesuai
 - frekuensinya yakni LL, LH, HL, dan HH dengan menggunakan transformasi wavelet
 - dengan filter *Haar* (Daubechies orde 1), secara matematis dihasilkan dengan
 - menggunakan persamaan sebagai

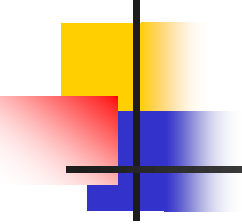


$$ll(x, y) = \frac{1}{4} \sum_{i=0}^1 f(2x + i, 2y + j)$$

$$lh(x, y) = \frac{1}{4} \sum_{i=0}^1 f(2x + i, 2y) - \frac{1}{4} \sum_{i=0}^1 f(2x + i, 2y + 1)$$

$$hl(x, y) = \frac{1}{4} \sum_{i=0}^1 f(2x, 2y + i) - \frac{1}{4} \sum_{i=0}^1 f(2x + 1, 2y + i)$$

$$hh(x, y) = \frac{1}{4} \{f(2x, 2y) + f(2x + 1, 2y + 1) - f(2x + 1, 2y) - f(2x, 2y + i)\}$$

- 
-
- dimana :
 - - $0 \leq x < M/2$, $0 \leq y < N/2$ dan $f(x,y)$ merupakan nilai piksel pada koordinat (x,y) pada
 - citra f .
 - - $ll(x,y)$, $lh(x,y)$, $hl(x,y)$, dan $hh(x,y)$ secara berturut-turut dalam komponen pada koordinat
 - (x,y) dari LL, LH, HL, dan HH.
 - - LL merupakan setengah dari resolusi citra asli, LH merupakan *subband detail vertikal*,
 - HL merupakan *subband detail horizontal*, dan HH merupakan *subband detail diagonal*.

Hasil Trans. Wavelet 1 level

