

# Transformasi Citra

IF4073 Interpretasi dan Pengolahan Citra

Oleh: Rinaldi Munir

Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung  
2019

# Ranah Frekuensi

- Citra dapat dioperasikan dalam ranah spasial maupun dalam ranah frekuensi.
- Dalam ranah spasial artinya memanipulasi langsung nilai-nilai pixel.
- Karena citra adalah sinyal dari (gelombang) cahaya, maka citra memiliki atribut frekuensi.
- Mengoperasikan citra dalam ranah frekuensi artinya memanipulasi nilai-nilai frekuensi yang merepresentasikan sinyal.
- Beberapa metode penapisan citra untuk menghilangkan daerah hanya berhasil dilakukan dalam ranah frekuensi ketimbang dalam ranah spasial.



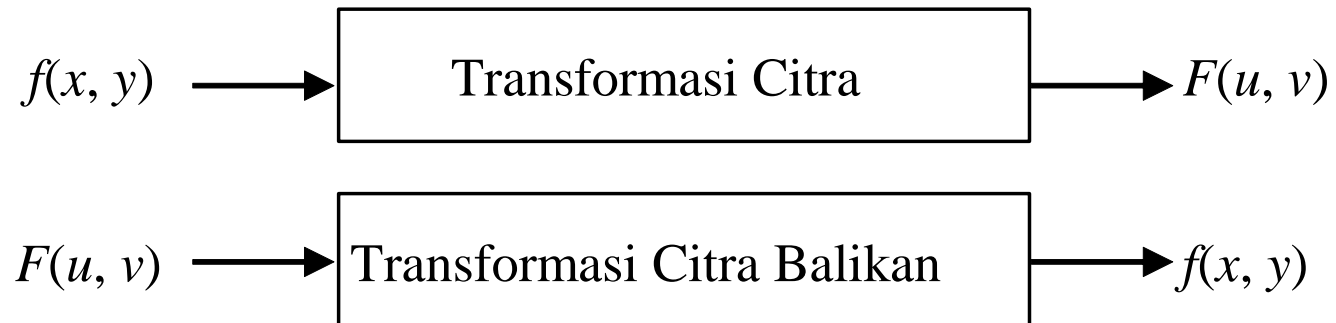
Citra dengan derau periodik. Derau tidak dapat dihilangkan dengan penapis dalam ranah spasial



Citra dengan derau periodik lainnya

- Selain itu, konvolusi dalam ranah spasial memakan waktu komputasi yang besar.
- Jika citra berukuran  $N \times N$  dan *kernel* berukuran  $m \times m$ , maka jumlah perkalian di dalam operasi konvolusi adalah dalam orde  $N^2m^2$ .
- Sebagai contoh jika citra berukuran  $512 \times 512$  dan kernel berukuran  $16 \times 16$ , maka ada sekitar 32 juta perkalian yang dibutuhkan.
- Ini jelas tidak cocok untuk proses yang *real time* tanpa perangkat keras yang *dedicated*.

- Untuk mengoperasikan citra dalam ranah frekuensi, maka citra dalam ranah spasial,  $f(x,y)$ , harus ditransformasikan menjadi citra dalam ranah frekuensi,  $F(u,v)$ .
- Setelah selesai dioperasikan dalam ranah frekuensi, maka citra dalam ranah frekuensi,  $F(u,v)$ , dapat dikembalikan ke dalam ranah spasial,  $f(x,y)$ .
- Kedua transformasi ini dinamakan transformasi citra (*image transform*) dan transformasi citra balikan (*inverse image transform*)



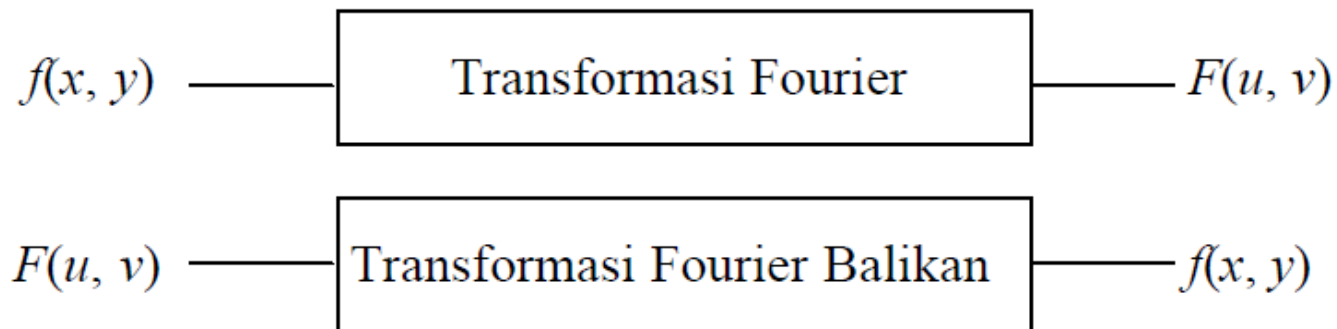
- Operasi konvolusi dalam ranah frekuensi cukup dilakukan dengan melakukan perkalian antara citra  $F(u,v)$  dengan kernel  $H(u,v)$ .

$$h(x, y) = f(x, y) * g(x, y) \leftrightarrow H(u, v) = F(u, v) G(u, v)$$

- Dua transformasi citra yang banyak digunakan adalah:
  1. Transformasi Fourier (*Fourier Transform*)
  2. Transformasi Cosine (*Discrete Cosine Transform*)

# Transformasi Fourier

- Transformasi Fourier adalah kakas (*tool*) untuk mengubah fungsi dari ranah waktu/spasial ke ranah frekuensi.
- Ranah waktu misalnya gelombang bunyi/suara (1-D). Ranah spasial misalnya citra (2-D).
- Untuk pengubahan sebaliknya, dari ranah frekuensi ke ranah waktu/spasial, digunakan Transformasi Fourier Balikan.



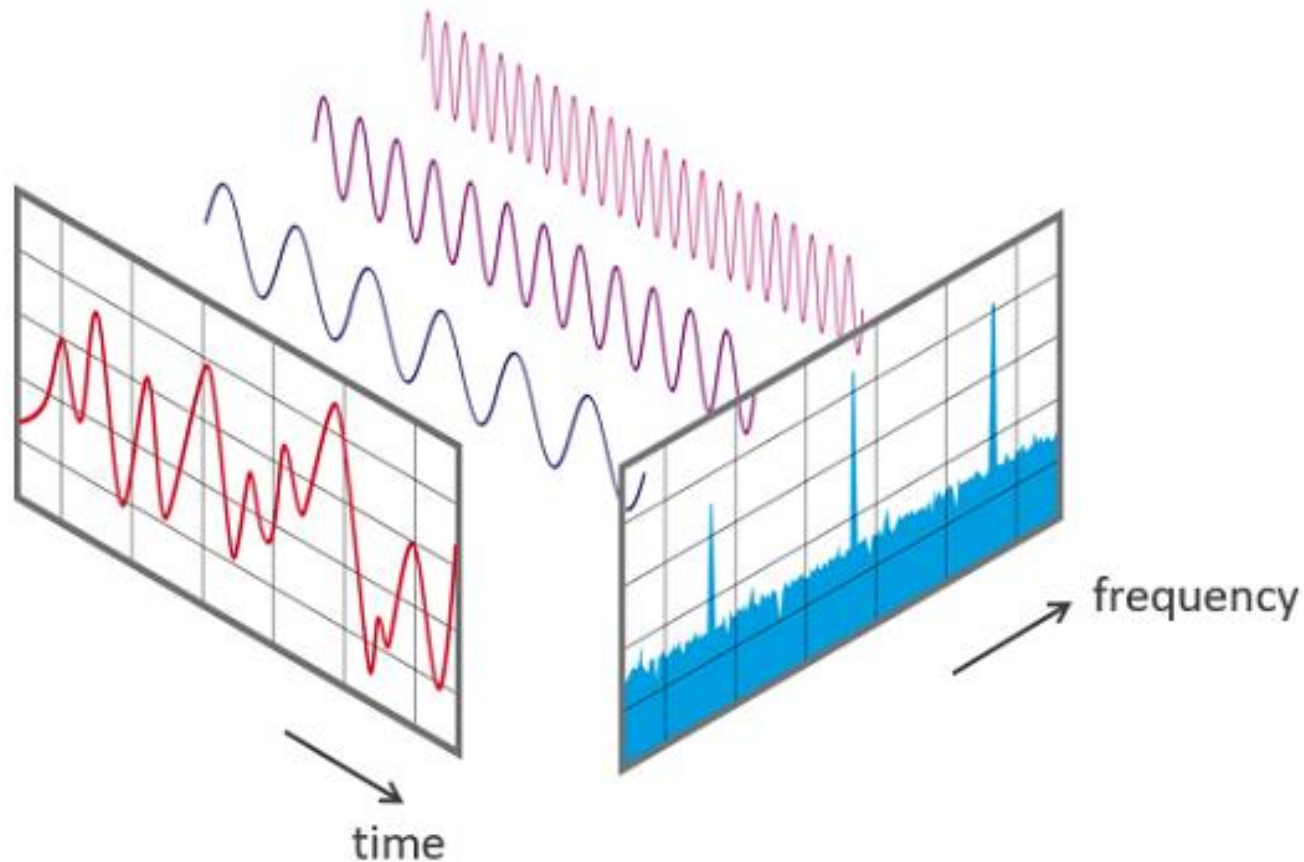


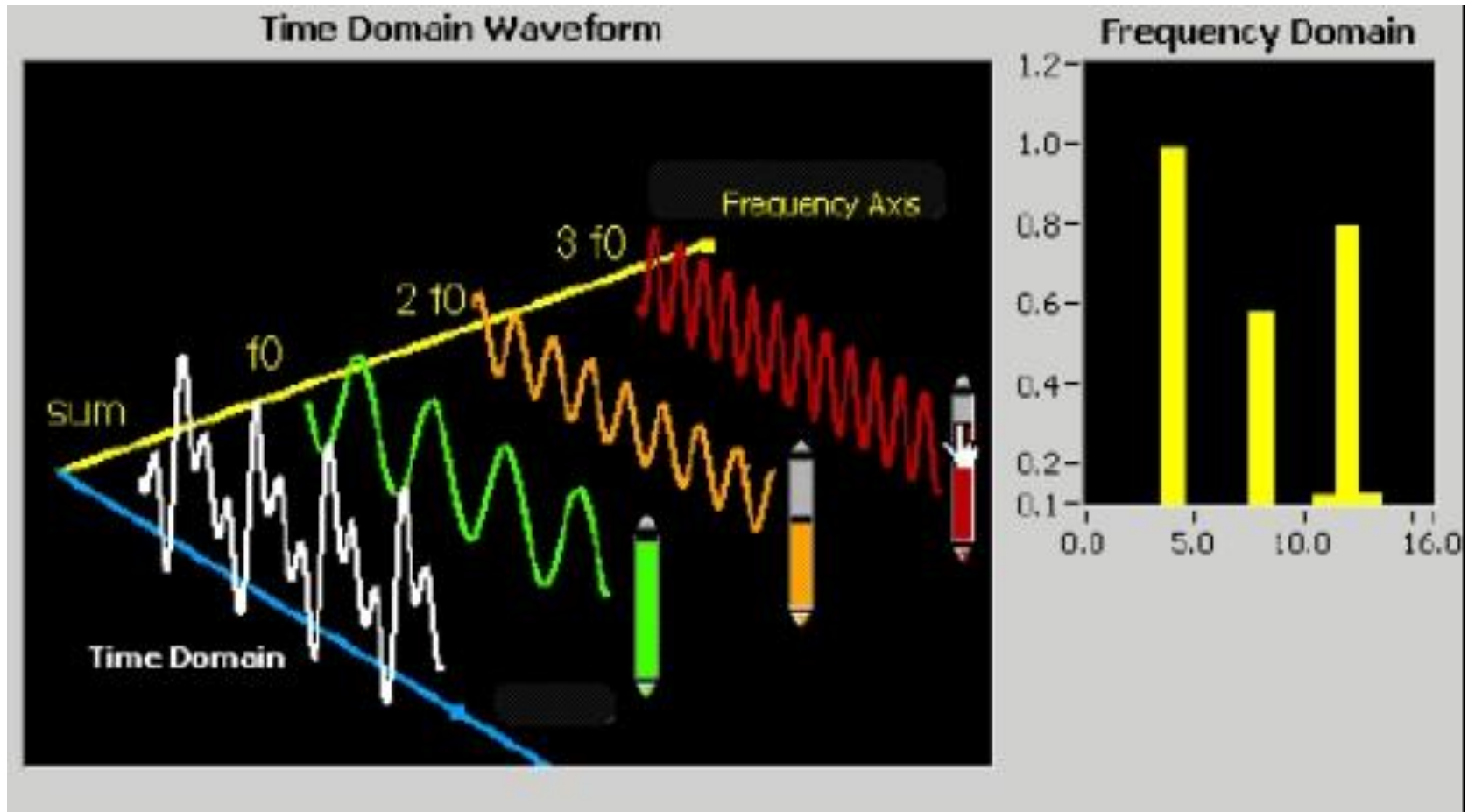
# Transformasi Fourier



Jean Baptiste Joseph Fourier (1768 - 1830)

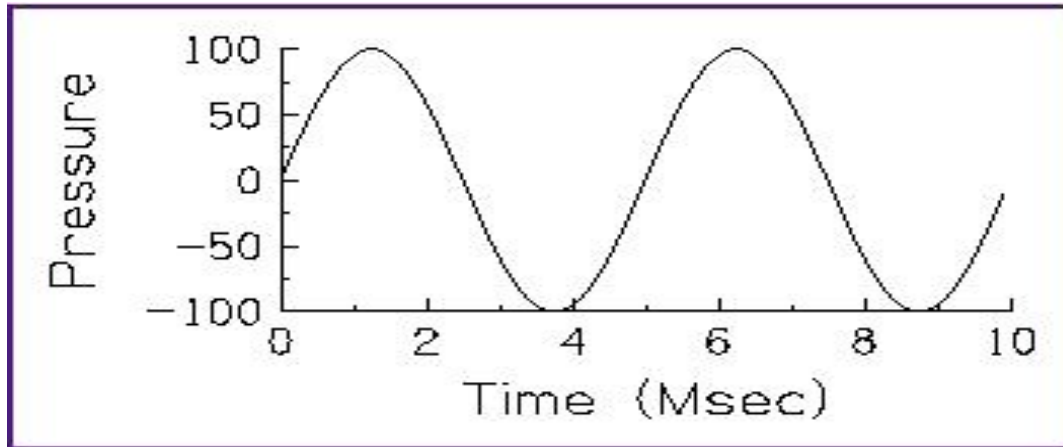
- Intisari dari Transformasi Fourier adalah menguraikan sinyal atau gelombang menjadi sejumlah sinusoida dari berbagai frekuensi, yang jumlahnya ekuivalen dengan gelombang asal.
- Contoh: sebuah gelombang dalam ranah waktu dibagi menjadi tiga buah gelombang sinusoida dengan masing-masing frekuensi berbeda.



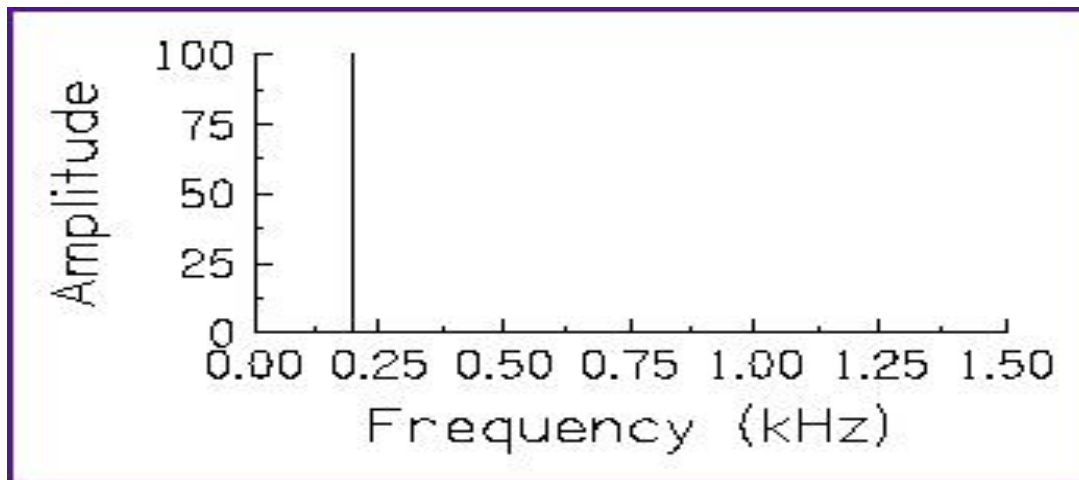


Sumber: Njegos Nincic, *Fourier Transform and Applications*

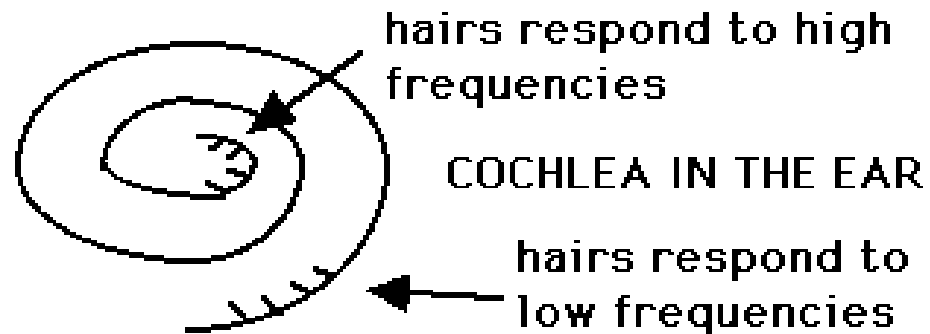
- Ranah Waktu (*Time Domain*) vs Ranah Frekuensi (*Frequency domain*)
  - Contoh: tekanan udara (sebagai fungsi bunyi) berubah terhadap waktu



Amplitudo = 100  
Frekuensi = banyaknya gelombang dalam satu detik = 200 Hz

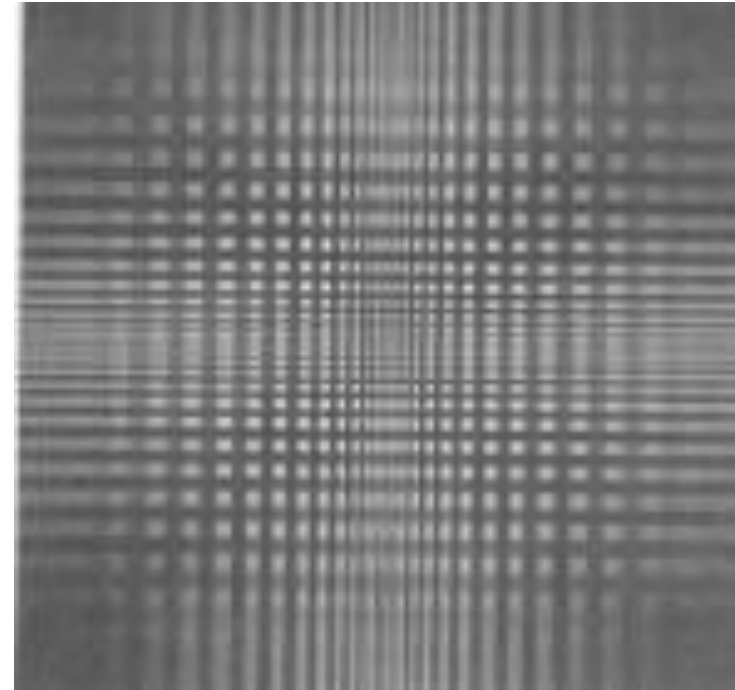


- Contoh:
  - Telinga manusia tidak mendengar bunyi seperti gelombang berosilasi, tetapi nada dengan frekuensi tertentu



- Seringkali lebih mudah menganalisis sinyal dalam ranah frekuensi daripada ranah waktu/spasial

- Di dalam pengolahan citra, umumnya citra disajikan dalam ranah spasial,  $I(x,y)$ .
- Citra juga dapat disajikan dalam *ranah frekuensi*  $F(u,v)$ , yaitu ruang di mana setiap nilai pada posisi citra  $F$  menyatakan besaran bahwa nilai intensitas citra berubah pada jarak tertentu.
- Contoh: jika frekuensi pada nilai *pixel* 20 adalah 0.1, artinya 1 periode setiap 10 pixel. Intensitas pixel berubah dari gelap ke terang dan kembali ke gelap setelah 10 pixel.



Gambar di atas adalah cira dalam ranah frekuensi:

- Frekuensi tinggi:  
Sekitar pusat
- Frekuensi rendah:  
Sekitar titik sudut

# Transformasi Fourier Kontinu

- Transformasi Fourier kontinu untuk satu peubah ( $x$  atau  $t$ ):

$$\mathfrak{F}\{f(x)\} = F(u) = \int_{-\infty}^{\infty} f(x)e^{-i2\pi ux} dx$$

- Transformasi Fourier Balikan untuk satu peubah:

$$\mathfrak{F}^{-1}\{F(u)\} = f(x) = \int_{-\infty}^{\infty} F(u)e^{i2\pi ux} du$$

yang dalam hal ini,

$$i = \text{imaginer} = \sqrt{-1}$$

$u$  adalah peubah frekuensi

$$2\pi x = \omega$$

- Untuk  $f(x)$  real,  $F(u)$  adalah fungsi kompleks dan dapat dituliskan sebagai:

$$F(u) = R(u) + iI(u) = |F(u)|e^{i\phi(u)}$$

- Amplitudo atau  $|F(u)|$  disebut **spektrum Fourier** dari  $f(x)$  dan didefinisikan sebagai:

$$|F(u)| = \sqrt{R^2(u) + I^2(u)}$$

- Sudut fase spektrum,

$$\Theta(u) = \tan^{-1}\left[\frac{I(u)}{R(u)}\right]$$

menyatakan pergeseran fase atau sudut fase dari setiap frekuensi  $u$ .



- Dengan mengingat kesamaan Euler,

$$e^{\pm ix} = \cos(x) \pm i \sin(x)$$

maka pasangan transformasi Euler dapat juga ditulis sebagai

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi ux} dx = \int_{-\infty}^{\infty} f(x) \{ \cos(2\pi ux) - i \sin(2\pi ux) \} dx$$

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{i2\pi ux} du = \int_{-\infty}^{\infty} F(u) \{ \cos(2\pi ux) + i \sin(2\pi ux) \} du$$

# Latihan

1. Misalkan  $f(t) = \begin{cases} 1, & |t| \leq T \\ 0, & t \text{ lainnya} \end{cases}$

Hitunglah hasil transformasi Fourier dan transformasi Fourier balikkannya.  
Hitung juga spektrum Fourier nya.

2. Misalkan  $f(t) = \begin{cases} t, & 0 \leq t \leq 1 \\ -\frac{(t-3)}{2}, & 1 < t \leq 3 \\ 0, & t \text{ lainnya} \end{cases}$

Hitunglah hasil transformasi Fourier dan transformasi Fourier balikkannya.  
Hitung juga spektrum Fourier nya.

- Transformasi Fourier kontinu untuk dua peubah:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i2\pi(ux+vy)} dx dy$$

Transformasi Fourier Balikannya adalah

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{i2\pi(ux+vy)} du dv$$

- Spektrum Fourier-nya adalah:

$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)}$$

- Sudut fase adalah

$$\Theta(u, v) = \tan^{-1} \left[ \frac{I(u, v)}{R(u, v)} \right]$$

- Sifat-sifat transformasi Fourier:

Sifat	Ranah Waktu	Ranah Frekuensi
1. Kelanjutan	$af(t) + bg(t)$	$aF(u) + bG(u)$
2. Penskalaan	$f(at)$	$\frac{1}{ a } F(u/a)$
3. Pergeseran	$f(t - a)$	$F(u - a)$
4. Modulasi	$e^{i2\pi at} f(t)$	$F(u) e^{-i2\pi ua}$
5. Konyugasi	$f^*(t)$	$F^*(-u)$
6. Konvolusi	$h(t) = f(t) * g(t)$	$H(u) = F(u)G(u)$
7. Perkalian	$h(t) = f(t)g(t)$	$H(u) = F(u) * G(u)$
8. Diferensiasi	$\frac{d^n f(t)}{dt^n}$	$(i2\pi u)^n F(u)$
9. Simetri	$F(t)$	$f(-u)$
10. Hasil kali dalam	$\int_{-\infty}^{\infty} f(t)g^*(t)dt$	$\int_{-\infty}^{\infty} F(u)G^*(u)du$

# Transformasi Fourier Diskrit

- Transformasi Fourier diskrit untuk satu peubah:

$$F_u = \frac{1}{N} \sum_{x=0}^{N-1} f_x e^{-i2\pi ux/N} \quad , \quad u = 0, 1, 2, \dots, N-1$$

- Transformasi Fourier Balikan untuk satu peubah:

$$f_x = \sum_{u=0}^{N-1} F_u e^{i2\pi ux/N} \quad , \quad x = 0, 1, 2, \dots, N-1$$

yang dalam hal ini, sinyal kontinu  $f(x)$  diterok menjadi  $N$  nilai diskrit sejarak  $\Delta x$ , yaitu himpunan nilai  $\{f(x_0), f(x_0 + \Delta x), f(x_0 + 2\Delta x), \dots, f(x_0 + (N-1)\Delta x)\}$ .

Jadi,

$$f_x = f(x_0 + x \Delta x), \quad x = 0, 1, 2, \dots, N-1$$

- Interpretasi dari TFD adalah sebagai berikut: TFD mengkonversi data diskrit menjadi sejumlah sinusoida diskrit yang frekuensinya dinomori dengan  $u = 0, 1, 2, \dots, N - 1$ , dan ampiltudonya diberikan oleh  $F(u)$ .
- Faktor  $1/N$  pada persamaan  $F(u)$  adalah faktor skala yang dapat disertakan dalam persamaan  $F(u)$  atau dalam persamaan  $f(x)$ , tetapi tidak kedua-duanya.
- Contoh: Diketahui fungsi sinyal  $f(x)$  dengan hasil penerokan ke dalam nilai-nilai diskrit sebagai berikut ( $N = 4$ ):
  - $x_0 = 0.5, f_0 = 2$
  - $x_1 = 0.75, f_1 = 3$
  - $x_2 = 1.0, f_2 = 4$
  - $x_3 = 1.25, f_3 = 4$

Transformasi Fourier Diskrit adalah sebagai berikut

$$F_0 = \frac{1}{4} \sum_{x=0}^3 f_x e^{-i \cdot 0.2\pi x/4} = \frac{1}{4} \sum_{x=0}^3 f_x e^0 = \frac{1}{4} \sum_{x=0}^3 f_x = \frac{1}{4} (f_0 + f_1 + f_2 + f_3) = 3.25$$

$$\begin{aligned} F_1 &= \frac{1}{4} \sum_{x=0}^3 f_x e^{-i \cdot 1.2\pi x/4} = \frac{1}{4} (f_0 e^0 + f_1 e^{-i\pi/2} + f_2 e^{-i\pi} + f_3 e^{-i3\pi/2}) \\ &= \frac{1}{4} (2 + 3[\cos(\pi/2) - i \sin(\pi/2)] + 4[\cos(\pi) - i \sin(\pi)] + 4[\cos(3\pi/2) - i \sin(3\pi/2)]) \\ &= \frac{1}{4} (2 + 3[0 - i] + 4[-1 - 0] + 4[0 + i]) = \frac{1}{4} (-2 - i) \end{aligned}$$

$$F_2 = \frac{1}{4} \sum_{x=0}^3 f_x e^{-i \cdot 2.2\pi x/4} = \frac{1}{4} (2e^0 + 3e^{-i\pi} + 4e^{-i2\pi} + 4e^{-i3\pi})$$

$$= \frac{1}{4} (1 + 0i) = -\frac{1}{4}$$

$$F_3 = \frac{1}{4} (2 + i)$$

Spektrum Fouriernya:

$$|F_0| = 3.25$$

$$|F_1| = \sqrt{(1/2)^2 + (1/4)^2} = \sqrt{1/4 + 1/16} = \sqrt{5/16} = \frac{1}{4}\sqrt{5}$$

$$|F_2| = \frac{1}{4}$$

$$|F_3| = \frac{1}{4}\sqrt{5}$$



```

void TFD(int N)
/* Melakukan Transformasi Fourier Diskrit untuk N buah data masukan. Hasil transformasi disimpan di
dalam array R dan I. Array R menyimpan bagian riil, dan array I menyimpan bagian bagian imajiner. Kedua
array ini dideklarasikan sebagai peubah global. Data masukan disimpan di dalam array f[0] s/d f[N-1]
*/
{
    int j, k;
    double tetha;

    for (j=0; j<N; j++)
    {
        R[j]=0.0;
        I[j]=0.0;
    }
    for (k=0; k<=N; k++)
        for (j=0; j<=N-1; j++)
            {
                tetha= k*2*3.14*j/(double)N;
                R[k]=R[k]+(f[j]*cos(tetha))/(double)N;
                I[k]=I[k]-(f[j]*sin(tetha))/(double)N;
            }
}

```

```

void TFD_balikan(int N)
/* Melakukan Transformasi Fourier Diskrit Balik untuk N buah data masukan. Masukan disimpan di dalam
array R dan I. Array R menyimpan bagian riil, dan array I menyimpan bagian bagian imajiner. Kedua array
ini dideklarasikan sebagai peubah global. Data keluaran disimpan di dalam array fReal[0] s/d fReal[N-1]
dan array fImag[1] s/d fImag[N-1]. */
{
    int j, k;
    double tetha, epsilon = 1E-12;

    for (j=0; j<N; j++)
    { fReal[j]=0;
      fImag[j]=0;
    }

    for (k=0; k<N; k++)
    { for (j=0; j<N; j++)
      {
          tetha=k*2*3.14*j/(double)N;
          fReal[k]=fReal[k]+(R[j]*cos(tetha)-I[j]*sin(tetha));
          fImag[k]=fImag[k]+(I[j]*cos(tetha)+ R[j]sin(tetha));
      }
      if (fImag[k] < epsilon) fImag[k]=0;
    }
}

```

- Transformasi Fourier Diskrit untuk dua peubah:

$$F_{u,v} = \frac{1}{NM} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f_{x,y} e^{-i2\pi(ux/N+vy/M)} \quad , u \text{ dan } v = 0, 1, 2, \dots, N-1$$

$$f_{x,y} = \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} F_{u,v} e^{i2\pi(ux/N+vy/M)} \quad , x \text{ dan } y = 0, 1, 2, \dots, N-1$$

atau

$$F_{u,v} = \frac{1}{N} \sum_{x=0}^{N-1} f_{x,y} e^{-i2\pi ux/N} \frac{1}{M} \sum_{y=0}^{M-1} f_{x,y} e^{-i2\pi vy/M} \quad , u \text{ dan } x = 0, 1, \dots, N-1$$

$$f_{x,y} = \sum_{u=0}^{N-1} F_{u,v} e^{i2\pi ux/N} \sum_{v=0}^{M-1} F_{u,v} e^{i2\pi vy/M} \quad , v \text{ dan } y = 0, 1, \dots, M-1$$

Spektrum Fourier-nya adalah:

$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)}$$

Sudut fasenya adalah:

$$\Theta(u, v) = \tan^{-1} \left[ \frac{I(u, v)}{R(u, v)} \right]$$

# Latihan

- Lakukan transformasi Fourier untuk citra berikut:

148 164 174

144 159 167

128 147 157

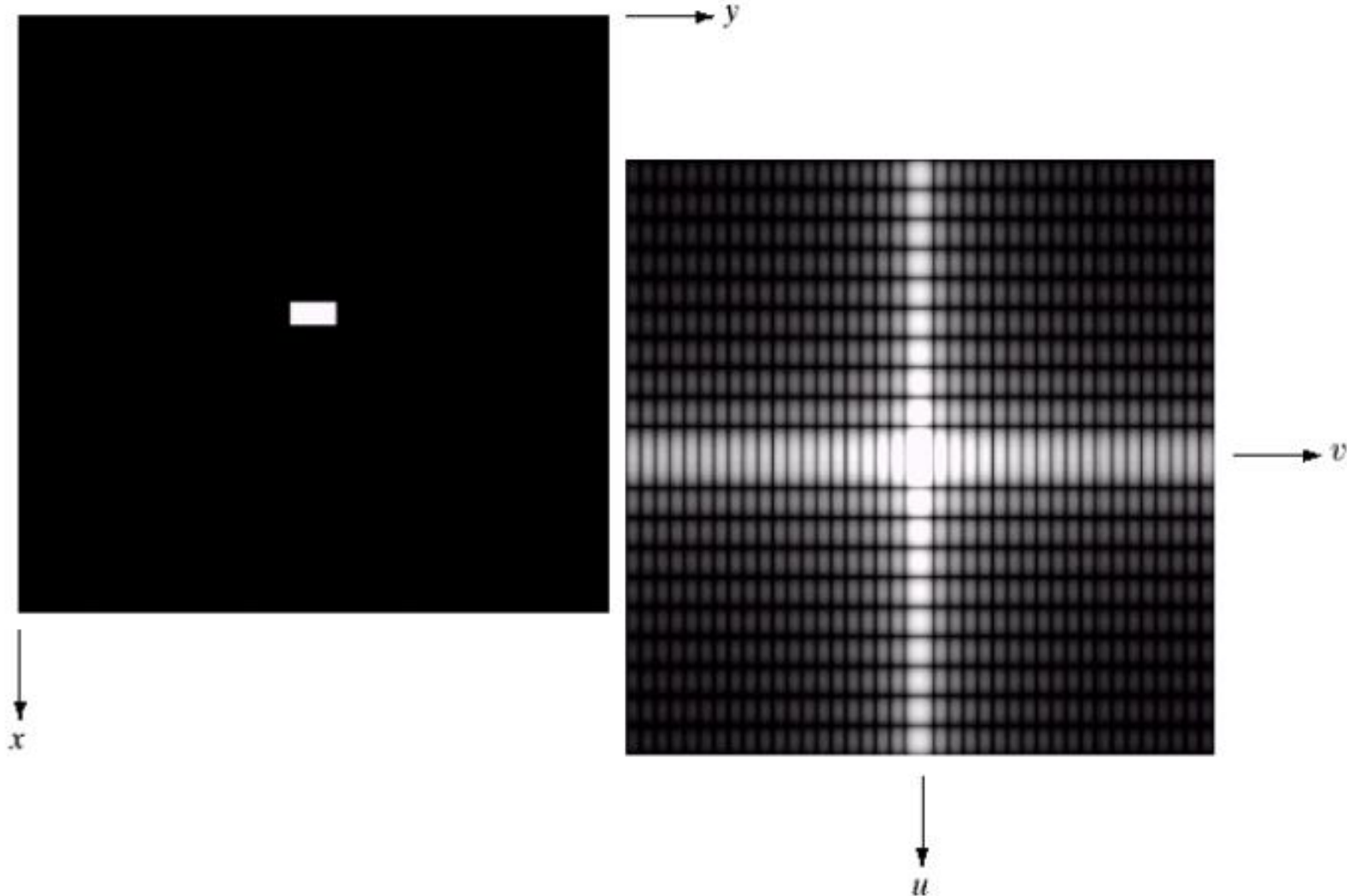
## Menampilkan citra spektrum Fourier

a b

**FIGURE 4.3**

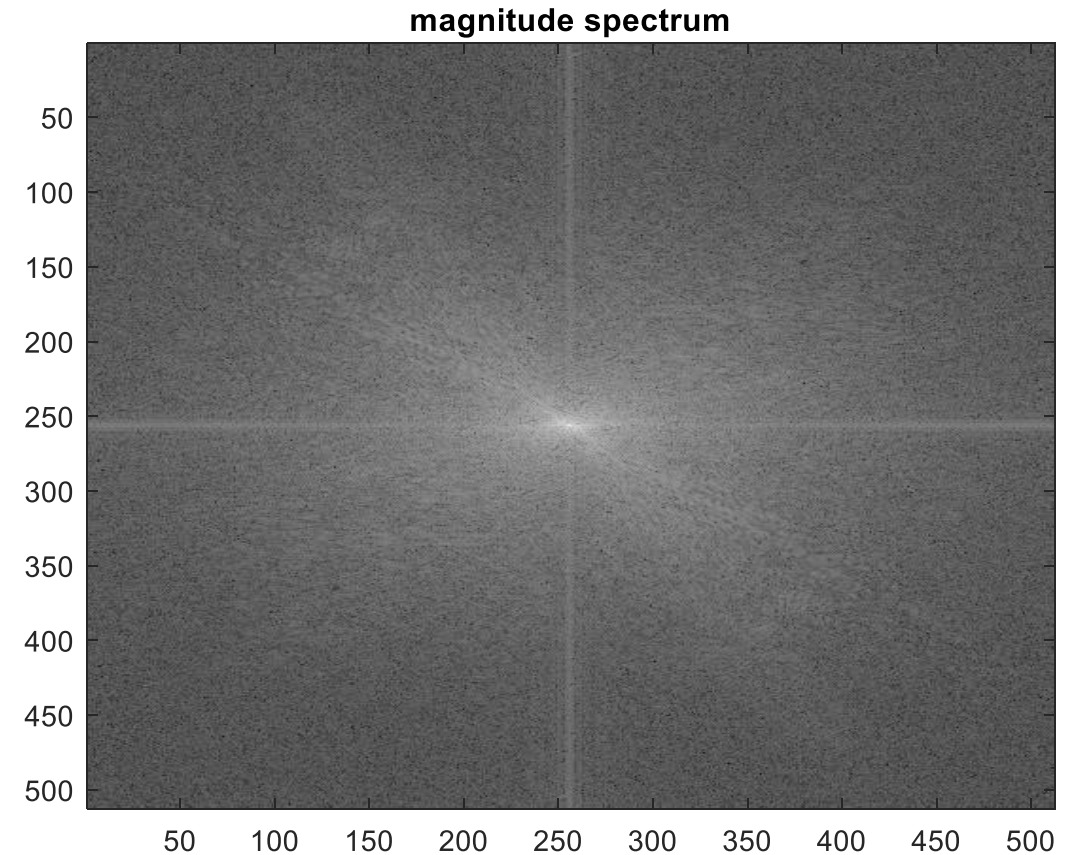
(a) Image of a  $20 \times 40$  white rectangle on a black background of size  $512 \times 512$  pixels.

(b) Centered Fourier spectrum shown after application of the log transformation given in Eq. (3.2-2). Compare with Fig. 4.2.



## Menampilkan citra spektrum Fourier

```
img = imread('lena-gray.bmp');  
imshow(img);  
F = fft2(img); % Perform Fourier Transform  
F2 = fftshift(F); % Center FFT  
F2 = abs(F2); % Get the magnitude  
F2 = log(F2+1); % Use log, for perceptual scaling, and +1 since log(0) is undefined  
figure, imagesc(100*F2); colormap(gray);  
title('magnitude spectrum');
```



# Simulasi FFT2, IFFT2, dan FFTShift pada sebuah matriks

```
>> A = magic(3)
```

```
A =  
 8  1  6  
 3  5  7  
 4  9  2
```

```
>> B = fft2(A)
```

```
B =  
45.0000 + 0.0000i  0.0000 + 0.0000i  0.0000 + 0.0000i  
0.0000 + 0.0000i 13.5000 + 7.7942i  0.0000 - 5.1962i  
0.0000 - 0.0000i  0.0000 + 5.1962i 13.5000 - 7.7942i
```

```
>> B = abs(B)
```

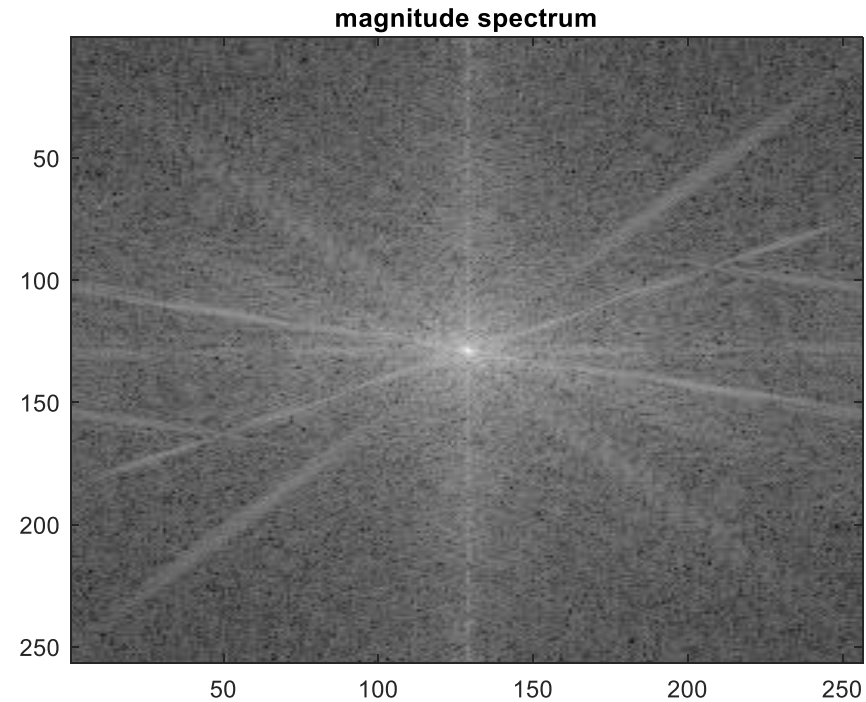
```
B =  
45.0000  0  0  
0.0000 15.5885  5.1962  
0.0000  5.1962 15.5885
```

```
>> B = fftshift(B)
```

```
B =  
15.5885  0.0000  5.1962  
  0 45.0000  0  
5.1962  0.0000 15.5885
```



- Transformasi citra dari ranah spasial ke ranah frekuensi dan sebaliknya



```
I = imread('camera.bmp');  
figure; imshow(I)
```

```
F = fft2(I);  
F2 = fftshift(F); % Center FFT  
F2 = abs(F2); % Get the magnitude  
F2 = log(F2+1); % Use log  
figure, imagesc(100*F2);  
colormap(gray);  
title('magnitude spectrum');
```

```
J = ifft2(F);  
J = uint8(J);  
figure, imshow(J)
```

- Kasus khusus:  $f(x,y)$  berukuran  $N \times N$ .

- Transformasi Fourier Diskrit:

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi \left( \frac{ux+vy}{N} \right)}, \quad u, v = 0, 1, 2, \dots, N-1$$

$$f(x, y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi \left( \frac{ux+vy}{N} \right)}, \quad x, y = 0, 1, 2, \dots, N-1$$

- Algoritma TFD 1-D dan balikkannya dapat diterapkan untuk fungsi diskrit dwimatra (2-D). Mula-mula transformasi dilakukan dalam arah  $x$  (dengan nilai  $y$  tetap). Kemudian, hasilnya ditransformasikan lagi dalam arah  $y$ .

- Cara lain: 
$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi\left(\frac{ux+vy}{N}\right)}$$

Pisahkan: 
$$e^{-j2\pi\left(\frac{ux+vy}{N}\right)} = e^{-j2\pi\left(\frac{ux}{N}\right)} e^{-j2\pi\left(\frac{vy}{N}\right)}$$

Tulis ulang  $F(u, v)$  sebagai berikut: 
$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{-j2\pi\left(\frac{ux}{N}\right)} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi\left(\frac{vy}{N}\right)}$$

Misalkan: 
$$\sum_{y=0}^{N-1} f(x, y) e^{-j2\pi\left(\frac{vy}{N}\right)} = F(x, v)$$
 , maka: 
$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{-j2\pi\left(\frac{ux}{N}\right)} F(x, v)$$

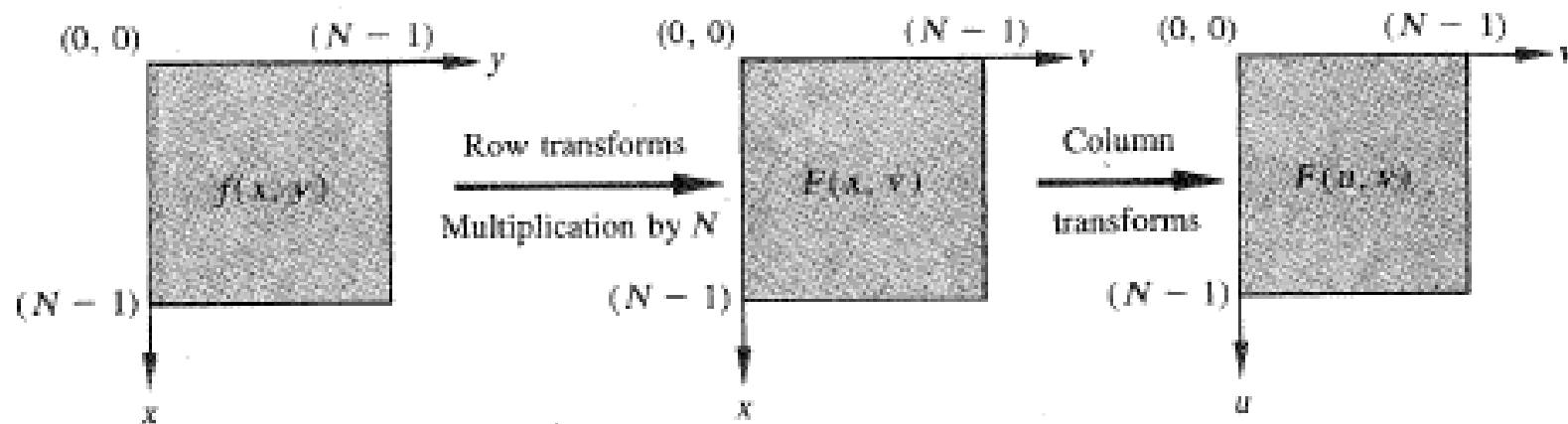
- Bagaimana cara menghitung  $F(x,v)$ ?

$$\sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(\frac{vy}{N})} = F(x, v) = N \left( \frac{1}{N} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(\frac{vy}{N})} \right)$$

**N x TFD dari baris-baris  $f(x,y)$**

- Bagaimana cara menghitung  $F(u,v)$ ?

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{-j2\pi(\frac{ux}{N})} F(x, v) \quad \text{TFD dari kolom-kolom } F(x,v)$$



- Algoritma TFD tidak bagus untuk  $N$  yang besar karena komputasinya memakan waktu yang lama. Kompleksitas waktu algoritmanya adalah  $O(N^2)$ .
- Algoritma yang dikenal cepat untuk menghitung transformasi Fourier diskrit adalah FFT (*Fast Fourier Transform*).
- Algoritma FFT mempunyai kompleksitas waktu  $O(N^2 \log N)$ . Jadi, untuk  $N = 50$ , FFT kira-kira 10 kali lebih cepat daripada TFD, untuk  $N = 1000$  sekitar 100 kali lebih cepat.

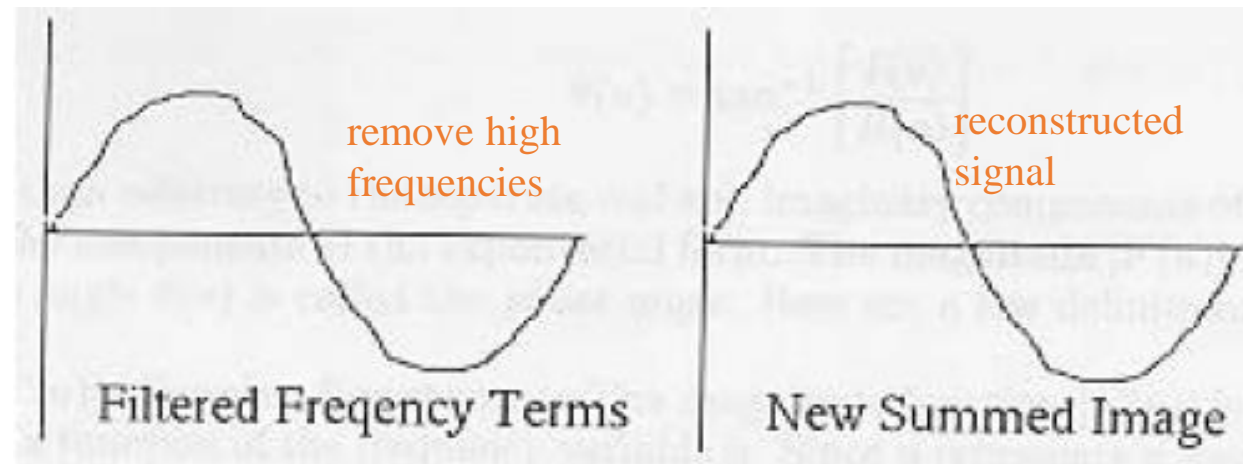
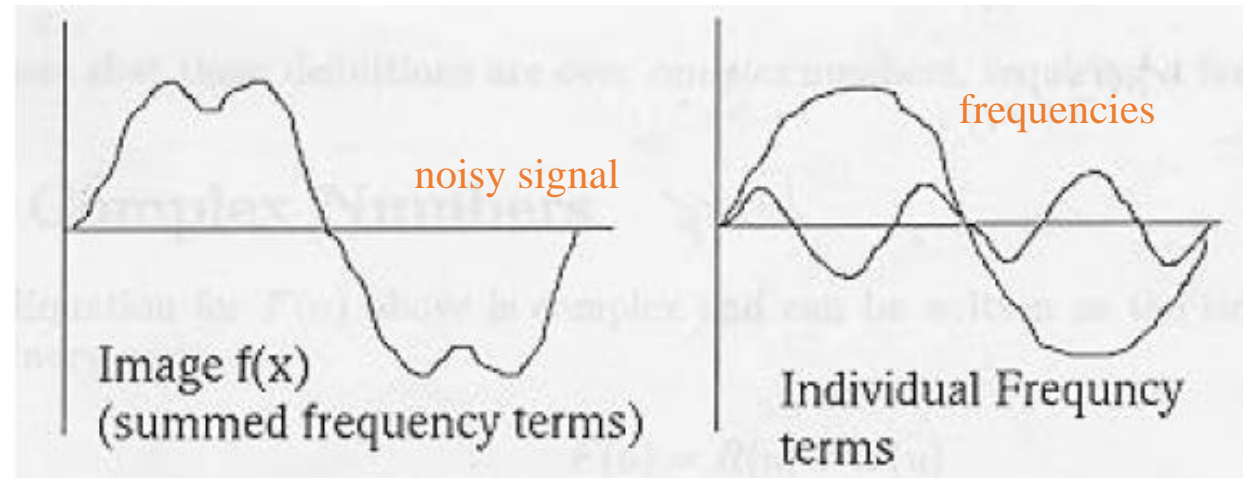
# Kenapa Transformasi Fourier bermanfaat?

- **Lebih mudah** menghilangkan frekuensi yang tidak diinginkan dalam ranah frekuensi.
- **Lebih cepat** melakukan operasi tertentu dalam ranah frekuensi daripada dalam ranah spasial (khususnya jika menggunakan **Fast Fourier Transform** atau **FFT**).

# Penapisan Frekuensi: Langkah utama

1. Lakukan Transformasi Fourier terhadap citra:  $F(u,v) = \mathcal{F}(f(x,y))$
2. Hilangkan frekuensi yang tidak diinginkan:  $G(u,v) = D(F(u,v))$
3. Transformasi balik ke ranah spasial:  $f(x,y) = \mathcal{F}^{-1}(F(u,v))$

# Contoh: menghilangkan frekuensi yang tidak diinginkan

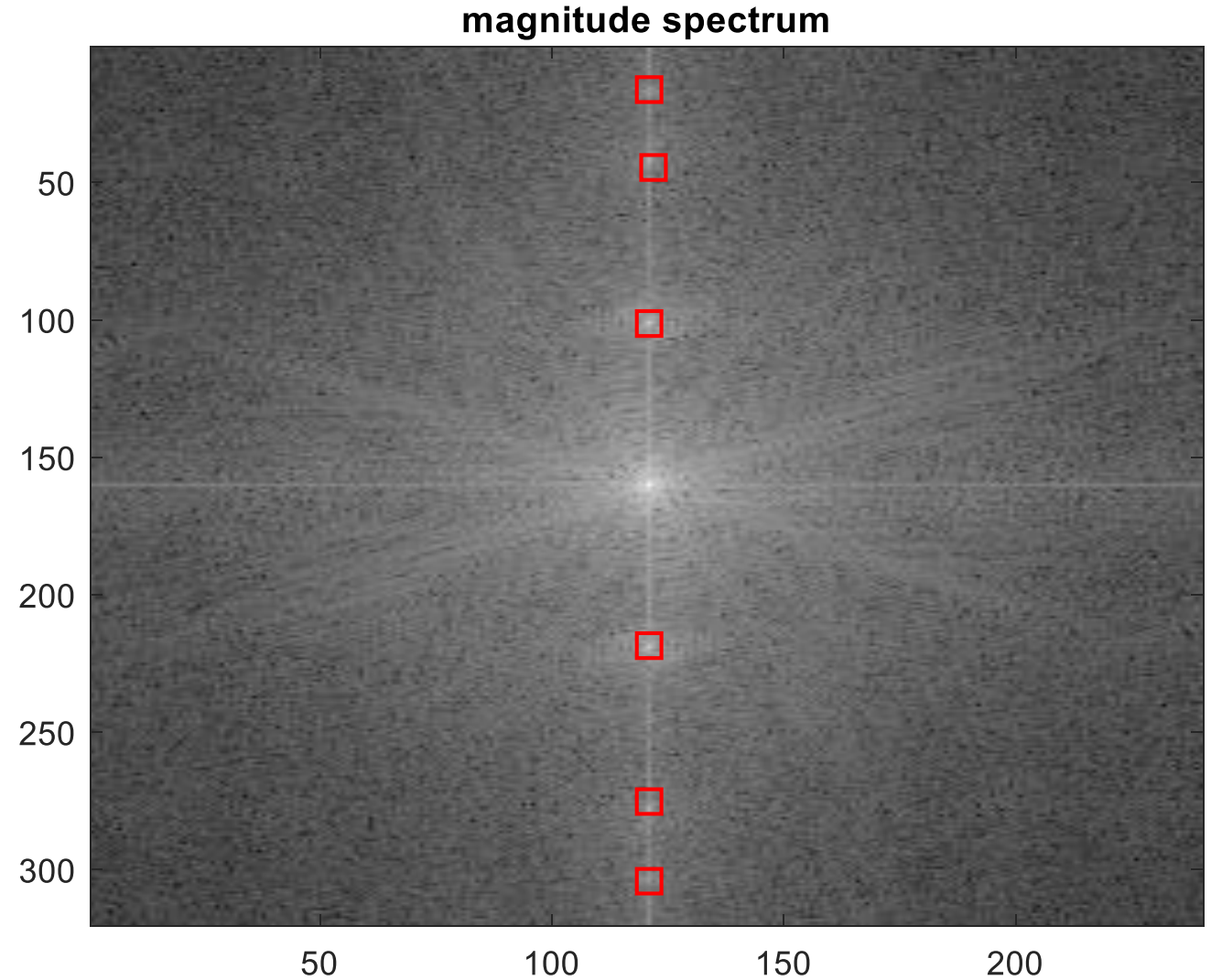


Untuk menghilangkan frekuensi tertentu, set koefisien  $F(u,v)$  yang berkoresponden menjadi 0

$$f(x) = \int_{-\infty}^{\infty} F(u)e^{j2\pi ux} du$$



Contoh:



Frekuensi yang mengalami derau ditentukan secara manual, kira-kira pada kolom 115-125, dan pada baris 96-106, 216-226, 274-284, 298-308, 12-22, 37-47.

## Hasil penapisan frekuensi derau:



```
I = imread('india.bmp'); % Baca citra
imshow(I);

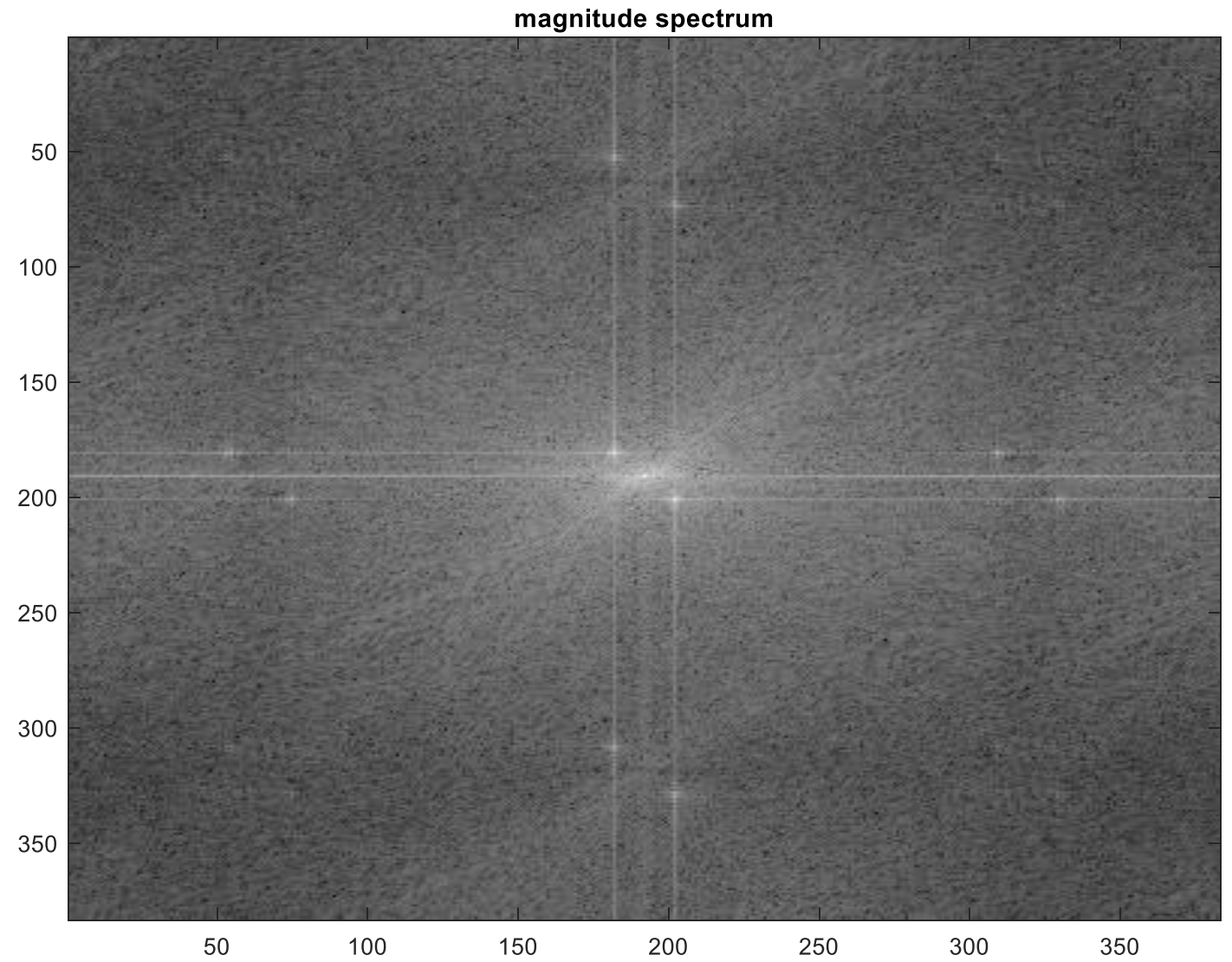
% Terapkan Fourier Transform
F = fft2(double(I));
F1 = fftshift(F); % Pusatkan FFT

% Tampilkan magnitute spektrum Fourier
F2 = F1;
F2 = abs(F2); % Get the magnitute
F2 = log(F2+1); % Use log
figure, imagesc(100*F2); colormap(gray);
title('magnitute spectrum');
```

```
% Buang frekuensi yang mengganggu, set jadi 0
for j = 115:125
    for i = 96:106
        F1(i,j) = 0;
    end
    for i = 216:226
        F1(i,j) = 0;
    end
    for i = 274:284
        F1(i,j) = 0;
    end
    for i = 298:308
        F1(i,j) = 0;
    end
    for i = 12:22
        F1(i,j) = 0;
    end
    for i = 37:47
        F1(i,j) = 0;
    end
end
end

%Kembalikan ke ranah spasial
J = real(ifft2(ifftshift(F1)));
figure, imshow(J,[]);
```

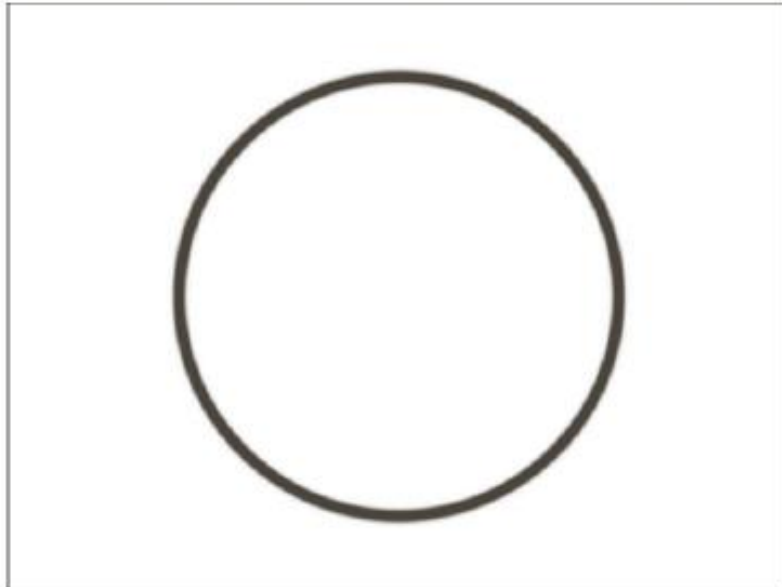
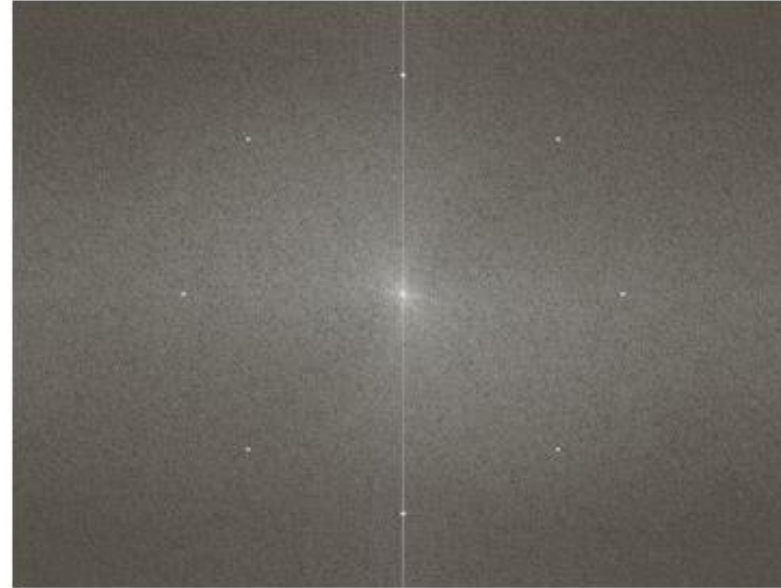
# Latihan:



Input image



Spectrum (frequency domain)



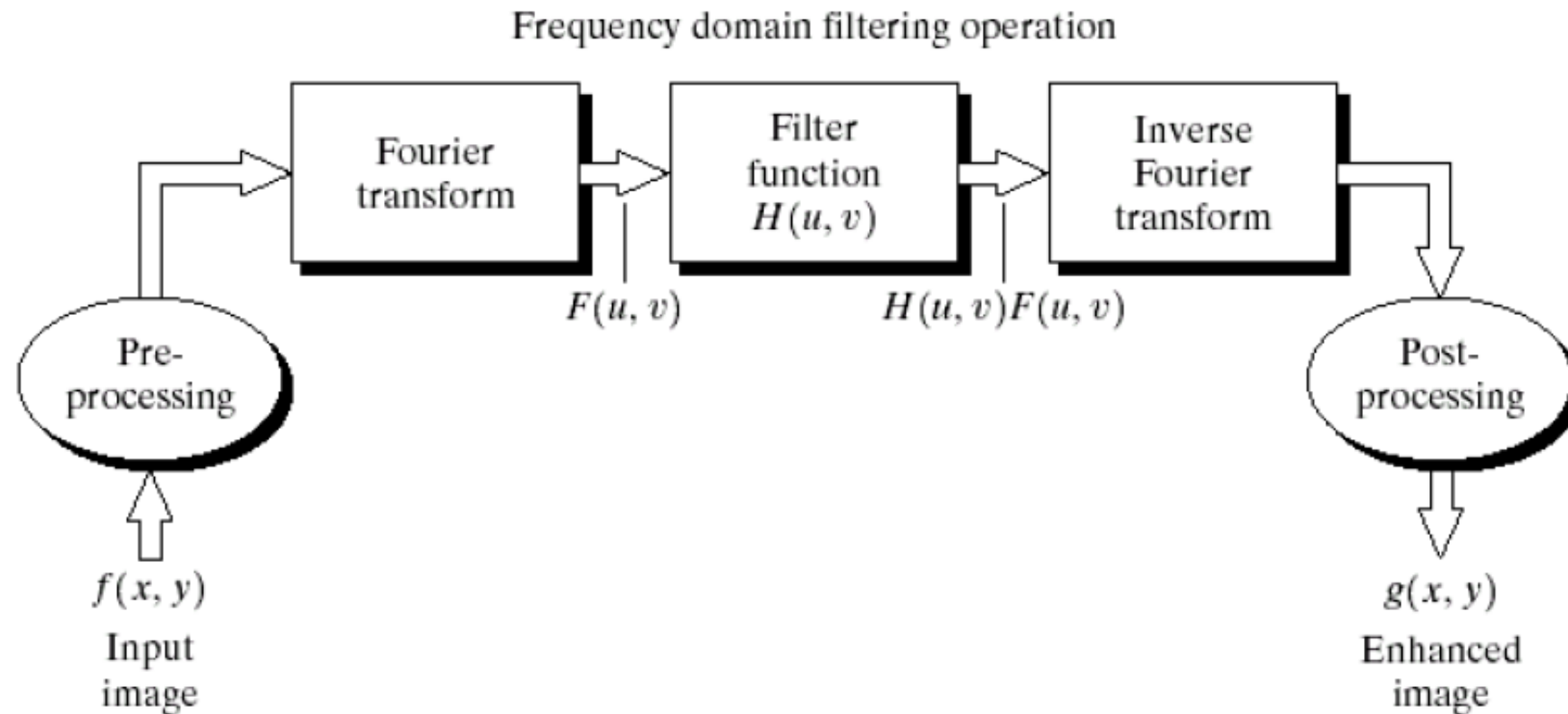
Band-reject filter



Output image



# Penapisan dalam Ranah Frekuensi



**FIGURE 4.5** Basic steps for filtering in the frequency domain.

```
>> A = magic(5)
```

```
A =
```

```
17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9
```

```
>> C = fft2(A).*fft2(B)
```

```
C =
```

```
8125 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

```
>> B = ones(5)
```

```
B =
```

```
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
```

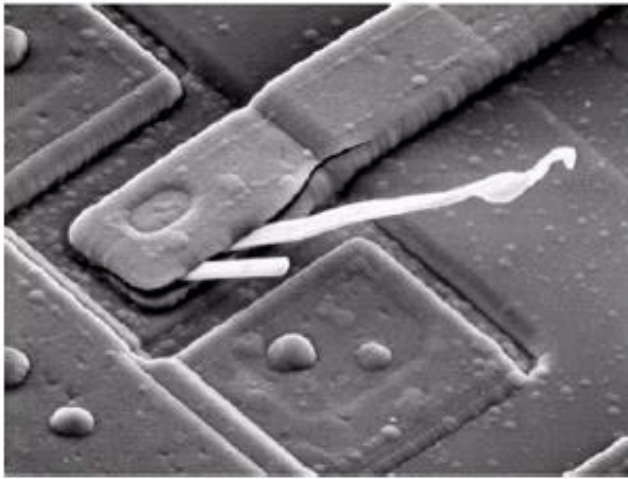
```
>> D = ifft2(C)
```

```
D =
```

```
325 325 325 325 325
325 325 325 325 325
325 325 325 325 325
325 325 325 325 325
325 325 325 325 325
```

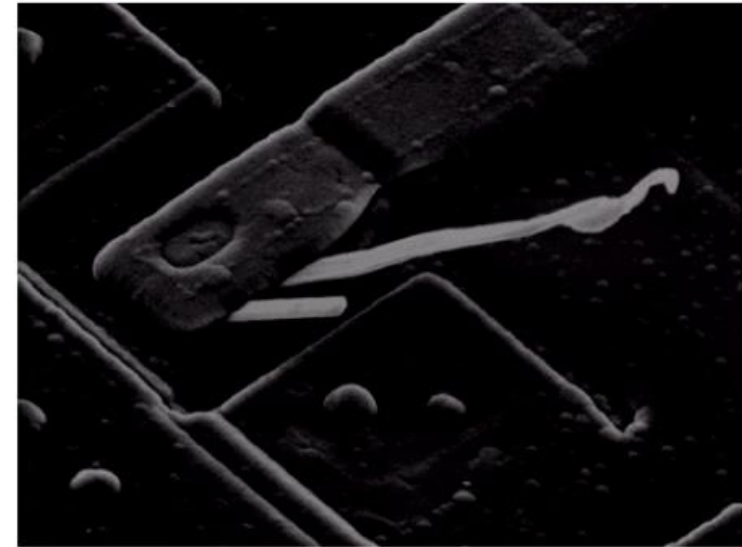
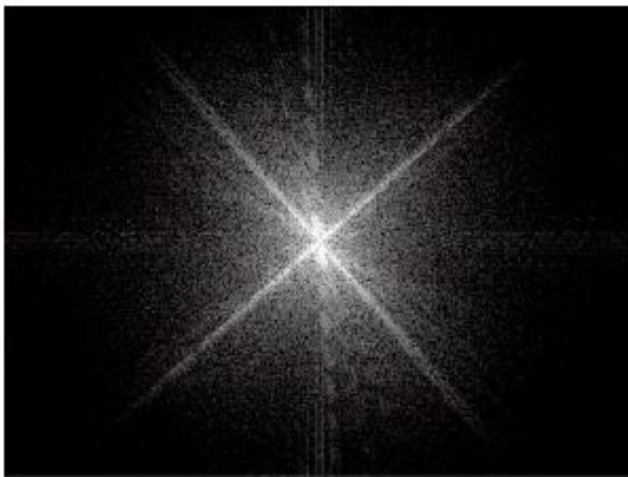
## Notch Filter: Membuang nilai rata-rata di dalam citra

$$H(u, v) = \begin{cases} 0 & \text{if } (u, v) = (M/2, N/2) \\ 1 & \text{otherwise} \end{cases}$$



a  
b

**FIGURE 4.4**  
(a) SEM image of a damaged integrated circuit.  
(b) Fourier spectrum of (a). (Original image courtesy of Dr. J. M. Hudak, Brockhouse Institute for Materials Research, McMaster University, Hamilton, Ontario, Canada.)



**FIGURE 4.6**  
Result of filtering the image in Fig. 4.4(a) with a notch filter that set to 0 the  $F(0, 0)$  term in the Fourier transform.



# Discrete Cosine Transform (DCT)

- DCT mentransformasikan sinyal dalam ranah waktu/spasial ke dalam ranah frekuensi.
- Tidak memiliki bagian imajiner.
- Pasangan transformasi DCT dan Inverse DCT

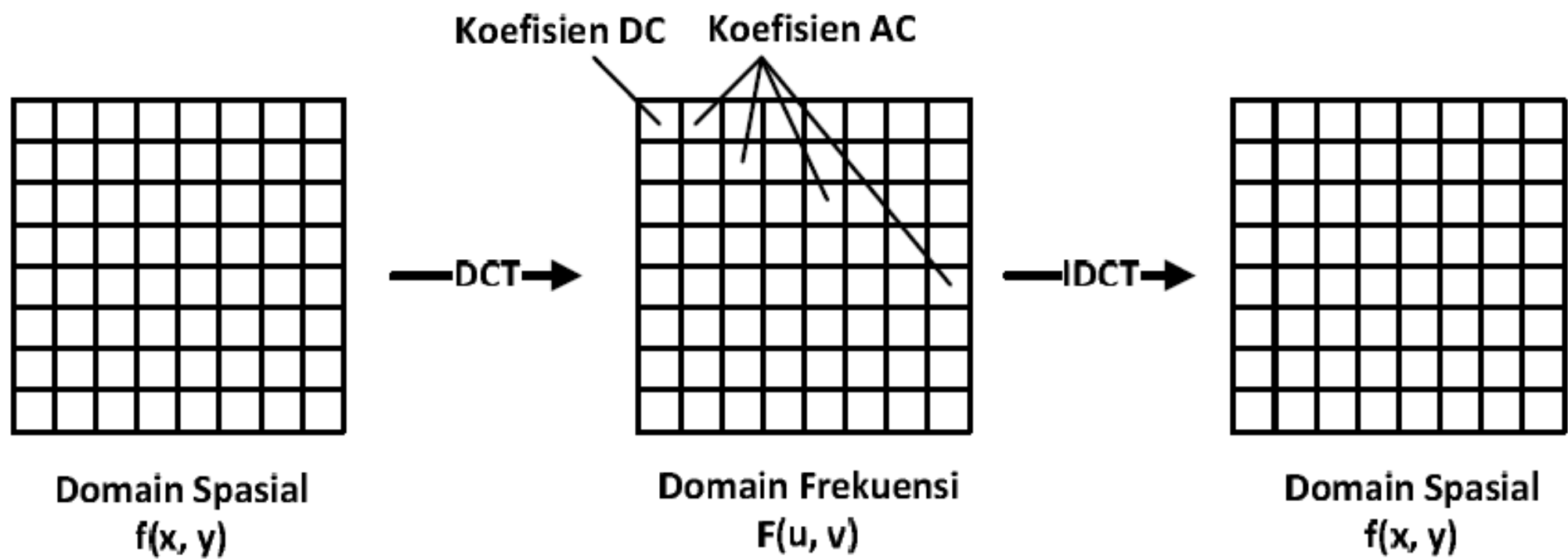
$$C(u, v) = \alpha_u \alpha_v \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x, y) \cos \frac{\pi(2x+1)u}{2M} \cos \frac{\pi(2y+1)v}{2N} \quad (1)$$

$C(u, v)$  disebut koefisien-koefisien DCT

$$I(x, y) = \alpha_u \alpha_v \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u, v) \cos \frac{\pi(2x+1)u}{2M} \cos \frac{\pi(2y+1)v}{2N} \quad (4)$$

$$\alpha_v = \begin{cases} \frac{1}{\sqrt{N}} & , v = 0 \\ \sqrt{\frac{2}{N}} & , 1 \leq v \leq N - 1 \end{cases}$$

$$\alpha_u = \begin{cases} \frac{1}{\sqrt{M}} & , u = 0 \\ \sqrt{\frac{2}{M}} & , 1 \leq u \leq M - 1 \end{cases}$$

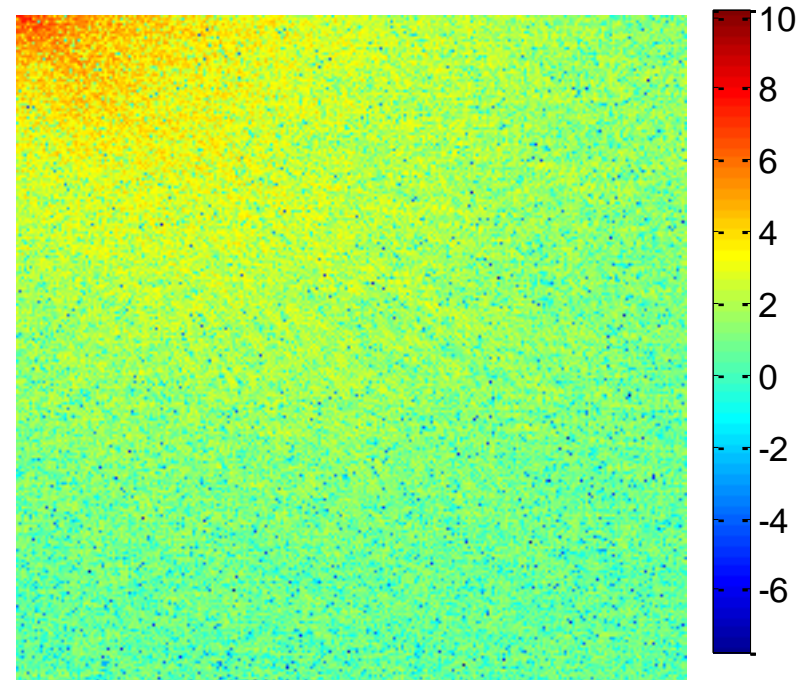




```
I = imread('lena-gray.bmp');  
J = dct2(I);  
imshow(log(abs(J)), []),  
colormap(gca, jet(64)), colorbar
```



Citra dalam ranah spasial



Citra dalam ranah frekuensi

# Tugas

- Implementasikan Transformasi Fourier dan balikkannya untuk suatu citra. Tampilkan citra spektrum Fourier nya.

