

LAMPIRAN A
KODE PROGRAM

```

%-----%
% %Program ini untuk menyembunyikan error dengan metode maximally smooth
%
%-----%

clear;
close all;
clc;

%-----%
% Gambar masukan yang sehat
%-----%
masukan_sehat=imread('lena','gif');
x_sehat=masukan_sehat;
x_sehat=double(x_sehat);

%-----%
% Hitung ukuran gambar
%-----%
baris=size(x_sehat,1);
kolom=size(x_sehat,2);

%-----%
% Hitung jumlah total blok image
%-----%
total_blok_image=round((baris*kolom)/(8*8));
baris_blok=baris/8;
kolom_blok=kolom/8;

%-----%
% Hitung DCT 2 dimensi untuk gambar sehat --> koefisien-DCT-per-blok
% Bagi blok input 8 * 8 --> hasilnya nanti sudah dalam domain DCT
%-----%
i=1;
for j=1:baris_blok
    for k=1:kolom_blok
        dct_x_sehat_temp(1:8,1:8)=dct2(x_sehat(8*(j-1)+1:8*j,8*(k-1)+1:8*k));
        if i<= total_blok_image
            blok_dct_sehat{i}=dct_x_sehat_temp;
            i=i+1;
        end;
        clear dct_x_sehat_temp;
    end;
end;

% Hapus variabel bantu
clear i j k

```

```

%-----%
% Proses mengembalikan koefisien-DCT-per-blok menjadi koefisien-DCT-satu-
gambar
%-----%
i=1;
dct_balik_sehat=[];
for j=1:baris_blok
    for k=1:kolom_blok
        if i<=total_blok_image
            dct_balik_sehat(8*(j-1)+1:8*j,8*(k-1)+1:8*k)=blok_dct_sehat{i};
            i=i+1;
        end;
    end;
end;

%Hapus variabel bantu
clear i j k

%-----%
% Proses membuat error koefisien-DCT-satu-gambar
% (1) Kasus dc hilang saja --> koefisien DCT posisi (1,1) tiap blok sama dengan
% nol
%-----%
i=1;
dct_dc_hilang=dct_balik_sehat;
for j=1:1:baris_blok
    for k=1:1:kolom_blok
        if i<=total_blok_image
            dct_dc_hilang((j-1).*8+1,(k-1).*8+1)=0;
            i=i+1;
        end;
    end;
end;

%Hapus variabel bantu
clear i j k

%-----%
% Mulai proses merekonstruksi koefisien dc yang hilang
%-----%
i=1;
rekonstruksi_dct_dc_hilang=[];
for j=1:1:baris_blok
    for k=1:1:kolom_blok
        if i<=total_blok_image

```

```

% Mengembalikan dahulu ke domain spasial yang benar saja
rekonstruksi_dct_dc_hilang((j-1).*8+1,:)=0;
rekonstruksi_dct_dc_hilang(j.*8,:)=0;
rekonstruksi_dct_dc_hilang(:,(k-1).*8+1)=0;
rekonstruksi_dct_dc_hilang(:,k.*8)=0;
rekonstruksi_dct_dc_hilang((j-1).*8+2:(j-1).*8+7,...
(k-1).*8+2:(k-1).*8+7)=x_sehat((j-1).*8+2:(j-1).*8+7,...
(k-1).*8+2:(k-1).*8+7);
% rekonstruksi_dct_dc_hilang((j-1).*8+2:(j-1).*8+7,...
% (k-1).*8+2:(k-1).*8+7)=idct2(dct_dc_hilang((j-1).*8+2:(j-1).*8+7,...
% (k-1).*8+2:(k-1).*8+7));
i=i+1;
end;
end;
end;

% gbr_dct_dc_rekon_full=idct2(dct_dc_hilang);
figure;
imshow(uint8(rekonstruksi_dct_dc_hilang));

%-----%
% Proses membalikkan koefisien-DCT-satu-gambar menjadi satu-gambar -->
% untuk gambar koefisien dc hilang
%-----%
i=1;
gbr_dct_dc_hilang=[];
for j=1:baris_blok
    for k=1:kolom_blok
        if i<=total_blok_image
            gbr_dct_dc_hilang(8*(j-1)+1:8*j,8*(k-1)+1:8*k)=...
                idct2(dct_dc_hilang(8*(j-1)+1:8*j,8*(k-1)+1:8*k));
            i=i+1;
        end;
    end;
end;

%-----%
% Proses membalikkan koefisien-DCT-satu-gambar menjadi satu-gambar -->
% untuk gambar sehat
%-----%
i=1;
gbr_balik_sehat=[];
for j=1:baris_blok
    for k=1:kolom_blok

```

```

    if i<=total_blok_image
        gbr_balik_sehat(8*(j-1)+1:8*j,8*(k-1)+1:8*k)=...
            idct2(dct_balik_sehat(8*(j-1)+1:8*j,8*(k-1)+1:8*k));
        i=i+1;
    end;
end;
end;

% Hapus variabel bantu
clear i j k

%-----%
% Rusakkan DCT 2 dimensi untuk gambar sehat
%-----%

figure;
imshow(masukan_sehat),title('Gambar sehat');
% subplot(1,2,2),imshow(uint8(gbr_dct_dc_hilang)),title('Gambar dc hilang');

figure;
% subplot(1,2,1),imshow(masukan_sehat),title('Gambar sehat');
imshow(uint8(gbr_dct_dc_hilang)),title('Gambar dc hilang');
-%-----%
% % Kondisi koefisien dc dan 5 frek rendah hilang / rusak
%-----%
for j=1:round(brs/8)
    for k=1:round(kol/8)
        J_modif=J_baru;
        J_modif(8*(j-1)+1,8*(k-1)+1)=0;
        J_modif(8*(j-1)+2,8*(k-1)+1)=0;
        J_modif(8*(j-1)+1,8*(k-1)+2)=0;
        J_modif(8*(j-1)+2,8*(k-1)+3)=0;
        J_modif(8*(j-1)+2,8*(k-1)+2)=0;
        J_modif(8*(j-1)+3,8*(k-1)+1)=0;
    end
end;
J_modif=idct2(J_modif);

clear j k
%-----%
% % 3. Kondisi semua koefisien rendah hilang / rusak
%-----%
J_modif_all=zeros(brs,kol);
J_modif_all=idct2(J_modif_all);

clear;

```

```

close all;
clc;

masukan_sehat=imread('lena.gif');

% Cari ukuran image
baris=size(masukan_sehat,1);
kolom=size(masukan_sehat,2);
ruang=size(masukan_sehat,3);

% Baca input yang sudah rusak
x_masuk=imread('lena1.gif');

% Hitung jumlah total blok image
total_blok_image=(baris*kolom)/(8*8);
baris_blok=baris/8;
kolom_blok=kolom/8;

% Bagi blok input 8 * 8
% for i=1:total_blok_image
i=1;
for j=1:baris_blok
    for k=1:kolom_blok
        blok_temp(1:8,1:8)=x_input(8*(j-1)+1:8*j,8*(k-1)+1:8*k);
        if i<= total_blok_image
            blok_image{i}=blok_temp;
            i=i+1;
        end;
        clear blok_temp;
    end;
end;

% Hapus variabel bantu
clear i j k

% Proses mambalikkan ke gambar sesudah dilakukan error concealment
i=1;
x_out_tiruan=[];
for j=1:baris_blok
    for k=1:kolom_blok
        if i<=total_blok_image
            x_out_tiruan(8*(j-1)+1:8*j,8*(k-1)+1:8*k)=blok_image{i};
            i=i+1;
        end;
    end;
end;
end;

```

```

%Hapus variabel bantu
clear i j k

% Ubah blok image yang dari 1 x 1024 menjadi blok image 32 x 32
row(1:baris_blok)=8;
column(1:kolom_blok)=8;
x_out_tiruan_cell=mat2cell(x_out_tiruan,[row],[column]);
x_out_tiruan_cell_1=cell2mat(x_out_tiruan_cell); % alat bantu untuk bisa
ditampilkan kembali

%=====
% Baris ke-59 sampai dengan baris ke-75 (bagian ini) bisa diabaikan %
%=====

Mencari/identifikasi blok yang rusak/error --> cari blok yang nilainya
% nol semua --> cell 1 x 1024
matriks_bantu=zeros(8,8);
matriks_rek=zeros(baris,kolom);
for i=1:total_blok_image
    if blok_image{i}==matriks_bantu
%        disp('Blok error');i
    else
%        disp('Blok sehat');i
    end;
end;

%Hapus variabel bantu
clear i
%=====
% Mencari/identifikasi blok yang rusak/error --> cari blok yang nilainya
% nol semua --> cell 32 x 32
%=====

matriks_bantu=zeros(8,8);

% -----%
% Start timer ---> awal program
% -----%
tic;

for j=1:baris_blok
    for k=1:kolom_blok
%
        if x_out_tiruan_cell{j,k}==matriks_bantu

```

```

% Beberapa nilai ini adalah inisialisasi bantu
vektor_Dk_total=zeros(1,8);
large_blok=zeros(24,24);

% Mencari blok tetangga yang tidak error
cell_pojok_kiri_atas=x_out_tiruan_cell{j-1,k-1};
x=cell_pojok_kiri_atas;

%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%

Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==5
            Dk_lima=Dk_lima+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==6
            Dk_enam=Dk_enam+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==7
            Dk_tujuh=Dk_tujuh+G_xy(m-1,n-1);
        end;
    end;
end;
% disp('Pojok kiri atas');
j;
k;

vektor_Dk_pojok_kiri_atas=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_li
ma,...
    Dk_enam,Dk_tujuh];

```



```

Dk_maks_pojok_kiri_atas=max(vektor_Dk_pojok_kiri_atas);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_pojok_kiri_atas;

indeks_Dk_maks_pojok_kiri_atas=find(vektor_Dk_pojok_kiri_atas==Dk_maks_
pojok_kiri_atas)-1;
%      pause;

cell_atas=x_out_tiruan_cell{j-1,k};
x=cell_atas;
%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%

Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;

for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==1
            Dk_satu=Dk_satu+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==2
            Dk_dua=Dk_dua+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==3
            Dk_tiga=Dk_tiga+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==4
            Dk_empat=Dk_empat+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==5
            Dk_lima=Dk_lima+G_xy(m-1,n-1);
        end;
    end;
end;

```

```

        if k_cek(m-1,n-1)==6
            Dk_enam=Dk_enam+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==7
            Dk_tujuh=Dk_tujuh+G_xy(m-1,n-1);
        end;
    end;
end;
% disp('Atas');
j;
k;
vektor_Dk_atas=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_lima,...
                Dk_enam,Dk_tujuh];
Dk_maks_atas=max(vektor_Dk_atas);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_atas;
indeks_Dk_maks_atas=find(vektor_Dk_atas==Dk_maks_atas)-1;
% pause;

cell_pojok_kanan_atas=x_out_tiruan_cell{j-1,k+1};
x=cell_pojok_kanan_atas;

%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%

Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==1
            Dk_satu=Dk_satu+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==2
            Dk_dua=Dk_dua+G_xy(m-1,n-1);

```

```

        end;

        if k_cek(m-1,n-1)==3
            Dk_tiga=Dk_tiga+G_xy(m-1,n-1);
        end;

    end;
end;

%     disp('Pojok kanan atas');
%     j;
%     k;

vektor_Dk_pojok_kanan_atas=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_lima,...
    Dk_enam,Dk_tujuh];
Dk_maks_pojok_kanan_atas=max(vektor_Dk_pojok_kanan_atas);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_pojok_kanan_atas;

indeks_Dk_maks_pojok_kanan_atas=find(vektor_Dk_pojok_kanan_atas==Dk_maks_pojok_kanan_atas)-1;
%     pause;

cell_kiri=x_out_tiruan_cell{j,k-1};
x=cell_kiri;
%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%

Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==0
            Dk_nol=Dk_nol+G_xy(m-1,n-1);
        end;
    end;
end;

```

```

    if k_cek(m-1,n-1)==1
        Dk_satu=Dk_satu+G_xy(m-1,n-1);
    end;

    if k_cek(m-1,n-1)==2
        Dk_dua=Dk_dua+G_xy(m-1,n-1);
    end;

    if k_cek(m-1,n-1)==3
        Dk_tiga=Dk_tiga+G_xy(m-1,n-1);
    end;

    if k_cek(m-1,n-1)==5
        Dk_lima=Dk_lima+G_xy(m-1,n-1);
    end;

    if k_cek(m-1,n-1)==6
        Dk_enam=Dk_enam+G_xy(m-1,n-1);
    end;

    if k_cek(m-1,n-1)==7
        Dk_tujuh=Dk_tujuh+G_xy(m-1,n-1);
    end;
end;
end;
%     disp('Kiri');
j;
k;
vektor_Dk_kiri=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_lima,...
                Dk_enam,Dk_tujuh];
Dk_maks_kiri=max(vektor_Dk_kiri);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_kiri;
indeks_Dk_maks_kiri=find(vektor_Dk_kiri==Dk_maks_kiri)-1;
%     pause;

cell_kanan=x_out_tiruan_cell{j,k+1};
x=cell_kanan;

%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%

Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7

```

```

        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==0
            Dk_nol=Dk_nol+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==1
            Dk_satu=Dk_satu+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==2
            Dk_dua=Dk_dua+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==3
            Dk_tiga=Dk_tiga+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==5
            Dk_lima=Dk_lima+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==6
            Dk_enam=Dk_enam+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==7
            Dk_tujuh=Dk_tujuh+G_xy(m-1,n-1);
        end;
    end;
end;
%     disp('Kanan');
j;
k;

vektor_Dk_kanan=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_lima,...
                Dk_enam,Dk_tujuh];
Dk_maks_kanan=max(vektor_Dk_kanan);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_kanan;
indeks_Dk_maks_kanan=find(vektor_Dk_kanan==Dk_maks_kanan)-1;

```

```

%      pause;

cell_bawah=x_out_tiruan_cell{j+1,k};
x=cell_bawah;
%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%
Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==1
            Dk_satu=Dk_satu+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==2
            Dk_dua=Dk_dua+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==3
            Dk_tiga=Dk_tiga+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==4
            Dk_empat=Dk_empat+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==5
            Dk_lima=Dk_lima+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==6
            Dk_enam=Dk_enam+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==7
            Dk_tujuh=Dk_tujuh+G_xy(m-1,n-1);
        end;
    end;
end;

```

```

        end;
    end;
end;
%     disp('Bawah');
    j;
    k;

vektor_Dk_bawah=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_lima,...
                 Dk_enam,Dk_tujuh];
Dk_maks_bawah=max(vektor_Dk_bawah);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_bawah;
indeks_Dk_maks_bawah=find(vektor_Dk_bawah==Dk_maks_bawah)-1;
%     pause;

cell_pojok_kiri_bawah=x_out_tiruan_cell{j+1,k-1};
x=cell_pojok_kiri_bawah;

%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%
Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==1
            Dk_satu=Dk_satu+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==2
            Dk_dua=Dk_dua+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==3
            Dk_tiga=Dk_tiga+G_xy(m-1,n-1);
        end;

    end;
end;

```

```

end;

%      disp('Pojok kiri bawah');
%      j;
%      k;

vektor_Dk_pojok_kiri_bawah=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,Dk_
lima,...
      Dk_enam,Dk_tujuh];
Dk_maks_pojok_kiri_bawah=max(vektor_Dk_pojok_kiri_bawah);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_pojok_kiri_bawah;

indeks_Dk_maks_pojok_kiri_bawah=find(vektor_Dk_pojok_kiri_bawah==Dk_m
aks_pojok_kiri_bawah)-1;
%      pause;

cell_pojok_kanan_bawah=x_out_tiruan_cell{j+1,k+1};
x=cell_pojok_kanan_bawah;
%-----%
Dk_nol=0;Dk_satu=0;Dk_dua=0;Dk_tiga=0;Dk_empat=0;
%-----%
Dk_lima=0;
Dk_enam=0;
Dk_tujuh=0;
for m=2:7
    for n=2:7
        S_x(m-1,n-1)=-x(m-1,n-1)-2*x(m,n-1)-x(m+1,n-1)+x(m-
1,n+1)+2*x(m,n+1)+x(m+1,n+1);
        S_y(m-1,n-1)=x(m-1,n-1)+2*x(m-1,n)+x(m-1,n+1)-x(m+1,n-1)-
2*x(m+1,n)-x(m+1,n+1);
        G_xy(m-1,n-1)=sqrt(S_x(m-1,n-1).^2+S_y(m-1,n-1).^2);
        sudut(m-1,n-1)=atan(S_y(m-1,n-1)/(S_x(m-1,n-1)+eps));
        sudut(m-1,n-1)=rad2deg(sudut(m-1,n-1));
        k_cek(m-1,n-1)=mod((round(sudut(m-1,n-1)/22.5)),8);

        if k_cek(m-1,n-1)==5
            Dk_lima=Dk_lima+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==6
            Dk_enam=Dk_enam+G_xy(m-1,n-1);
        end;

        if k_cek(m-1,n-1)==7
            Dk_tujuh=Dk_tujuh+G_xy(m-1,n-1);
        end;
    end;
end;

```



```

        end;
%       disp('Pojok kanan bawah');
        j;
        k;

vektor_Dk_pojok_kanan_bawah=[Dk_nol,Dk_satu,Dk_dua,Dk_tiga,Dk_empat,D
k_lima,...

        Dk_enam,Dk_tujuh];
Dk_maks_pojok_kanan_bawah=max(vektor_Dk_pojok_kanan_bawah);
vektor_Dk_total=vektor_Dk_total+vektor_Dk_pojok_kanan_bawah;

indeks_Dk_maks_pojok_kanan_bawah=find(vektor_Dk_pojok_kanan_bawah==
Dk_maks_pojok_kanan_bawah)-1;
%       pause;
vektor_Dk_maks=max(vektor_Dk_total);
indeks_vektor_Dk_maks=find(vektor_Dk_total==vektor_Dk_maks)-1;

if indeks_vektor_Dk_maks==0
    teta=0;
elseif indeks_vektor_Dk_maks==1
    teta=22.5;
elseif indeks_vektor_Dk_maks==2
    teta=45;
elseif indeks_vektor_Dk_maks==3
    teta=67.5;
elseif indeks_vektor_Dk_maks==4
    teta=90;
elseif indeks_vektor_Dk_maks==5
    teta=112.5;
elseif indeks_vektor_Dk_maks==6
    teta=135;
elseif indeks_vektor_Dk_maks==7
    teta=157.5;
end;
teta=deg2rad(teta);

%       disp('Ini lho ....')

large_blok(1:8,1:8)=cell_pojok_kiri_atas;
large_blok(1:8,9:16)=cell_atas;

```

```

large_blok(1:8,17:24)=cell_pojok_kanan_atas;
large_blok(9:16,1:8)=cell_kiri;
large_blok(9:16,9:16)=matriks_bantu;
large_blok(9:16,17:24)=cell_kanan;
large_blok(17:24,1:8)=cell_pojok_kiri_bawah;
large_blok(17:24,9:16)=cell_bawah;
large_blok(17:24,17:24)=cell_pojok_kanan_bawah;

% large_blok;

fft_large_blok=fft2(large_blok);

if vektor_Dk_total<T
% disp('Monotone (identik smooth)');
abs_fft_large_blok=abs(fft_large_blok);

for p=1:24 % p adalah variabel bantu untuk jumlah baris large_blok
    for q=1:24 % q adalah variabel bantu untuk jumlah kolom large_blok
        if abs_fft_large_blok(p,q)>R_th
            proyeksi_2_smooth(p,q)=0;
        else
            proyeksi_2_smooth(p,q)=fft_large_blok(p,q);
        end;
    end;
end;
proyeksi_2=proyeksi_2_smooth;
else
% disp('Edge');
delta_sudut=deg2rad(90);
r=fft_large_blok; % r adalah variabel bantu untuk fft_large_blok
for p=1:24 % p adalah variabel bantu untuk jumlah baris large_blok
    for q=1:24 % q adalah variabel bantu untuk jumlah kolom large_blok
        batas_ambang(p,q)=abs(real(r(p,q))-
imag(r(p,q)).*tan(teta+delta_sudut));
        if batas_ambang(p,q)>B_th
            proyeksi_2_edge(p,q)=0;
        else
            proyeksi_2_edge(p,q)=fft_large_blok(p,q);
        end;
    end;
end;
proyeksi_2=proyeksi_2_edge;
end;
fft_large_blok(9:16,9:16)=proyeksi_2(9:16,9:16);
matriks_rekonstruksi=real(ifft2(fft_large_blok(9:16,9:16)));

```

```

for g=1:8
    for h=1:8
        if matriks_rekonstruksi(g,h)<0
            proyeksi_1(g,h)=0;
        elseif matriks_rekonstruksi(g,h)>255
            proyeksi_1(g,h)=255;
        else
            proyeksi_1(g,h)=matriks_rekonstruksi(g,h);
        end;
    end;
end;
mat_ganti=(cell_pojok_kiri_atas+cell_atas+cell_pojok_kanan_atas+...
    cell_kiri+cell_kanan+cell_bawah+cell_pojok_kiri_bawah+...
    cell_pojok_kanan_bawah)/8;
x_out_tiruan_cell{j,k}=mat_ganti;
% x_out_tiruan_cell{j,k}=proyeksi_1;
% pause
clc
end;

matriks_sdh_sehat=cell2mat(x_out_tiruan_cell);
end;

end;

masuk_sehat=double(masukan_sehat);
masuk_sehat=reshape(masuk_sehat,1,baris*kolom);

% -----%
% Perhitungan SNR awal sebelum rekonstruksi
% -----%

x_sakit=double(x_masuk);
x_sakit=reshape(x_sakit,1,baris*kolom);
MSE_sakit=sum((masuk_sehat-x_sakit).^2);
PSNR_sakit=10.*log10((255.^2)./(MSE_sakit/(baris*kolom)));
disp(['Nilai SNR awal sebelum rekonstruksi = ',num2str(PSNR_sakit),' dB']);
disp(' ');

```

```

% -----%
% Perhitungan SNR akhir sesudah rekonstruksi
% -----%

sdh_rekonstruksi=reshape(matriks_sdh_sehat,1,baris*kolom);
MSE_rekonstruksi=sum((masuk_sehat-sdh_rekonstruksi).^2);
PSNR_rekonstruksi=10.*log10((255.^2)./(MSE_rekonstruksi/(baris*kolom)));
disp(['Nilai SNR akhir sesudah rekonstruksi = ',num2str(PSNR_rekonstruksi),'
dB']);
disp(' ');

kenaikan_PSNR=PSNR_rekonstruksi-PSNR_sakit;
disp(['Perbaikan SNR sebesar ',num2str(kenaikan_PSNR),' dB']);
disp(' ');

if PSNR_sakit <= PSNR_rekonstruksi
    disp('Rekonstruksi telah berhasil');
else
    disp('Rekonstruksi gagal');
end;

toc; % End timer --> akhir program

figure;
imshow(uint8(matriks_sdh_sehat)),title('Gambar sesudah direkonstruksi');

%Hapus variabel bantu
clear j k

%=====

```