REFERENCES


[58] SudhaPonnarasi, S., and Rajaram, M., 2011, Estimating the Impact of Minutia Extraction Algorithms in Fingerprint Recognition,


clear all
close all
clc
I=(imread('10_2.bmp'));
[X,map] = rgb2ind(I,128);
surf(double(X))
xlabel('Rows')
ylabel('Columns')
zlabel('Intensity profile')
figure
% plane information
imshow(I(:,:,1))
title('Plane-1')
surf(double(I(:,:,1)))
title('Plane-1')
xlabel('Rows')
ylabel('Columns')
zlabel('Intensity profile')
figure
imshow(I(:,:,2))
title('Plane-2')
surf(double(I(:,:,2)))
title('Plane-2')
xlabel('Rows')
ylabel('Columns')
zlabel('Intensity profile')
imshow(I(:,:,3))
title('Plane-1')

figure
surf(double(I(:,:,3)))
title('Plane-3')
xlabel('Rows')
ylabel('Columns')
zlabel('Intensity profile')
figure
imshow(I)
title('True color image')

figure
X=X(1:256,1:256);
ncol = size(map,1);
[ca1,ch1,cv1,cd1] = dwt2(X,'db1');
[rough_param] = calculate_roughness_L1A(ca1);
L1 = rough_param;

figure
[rough_param] = calculate_roughness_L1H(ch1);
L1 = [L1 rough_param];

figure
[rough_param] = calculate_roughness_L1V(cv1);
L1 = [L1 rough_param];

figure
[rough_param] = calculate_roughness_L1D(cd1);
L1 = [L1 rough_param];

close all
[A1] = feature_extraction_wavelet(ca1,ch1,cv1,cd1);
cod_ca1 = wcodemat(ca1,ncol);
figure
surf(cod_ca1)
ylabel('Rows=128')
xlabel('Columns=128')
zlabel('Coefficients')
title('Approximation Level-1')
figure
cod_ch1 = wcodemat(ch1,nbcol);
surf(cod_ch1)
ylabel('Rows=128')
xlabel('Columns=128')
zlabel('Coefficients')
title('Horizontal Level-1')
figure
cod_cv1 = wcodemat(cv1,nbcol);
surf(cod_cv1)
ylabel('Rows=128')
xlabel('Columns=128')
zlabel('Coefficients')
title('Vertical Level-1')
figure
cod_cd1 = wcodemat(cd1,nbcol);
surf(cod_cd1)
ylabel('Rows=128')
xlabel('Columns=128')
zlabel('Coefficients')
title('Diagonal Level-1')
figure
image([cod_ca1,cod_ch1;cod_cv1,cod_cd1]);
title('First level decomposition')
figure
% Perform second step decomposition :
% decompose approx. cfs of level 1.
[ca2,ch2,cv2,cd2] = dwt2(ca1,'db1');
[rough_param]=calculate_roughness_L2A(ca2);
L2=rough_param;
figure
[rough_param]=calculate_roughness_L2H(ch2);
L2=[L2 rough_param];
figure
[rough_param]=calculate_roughness_L2V(cv2);
L2=[L2 rough_param];
figure
[rough_param]=calculate_roughness_L2D(cd2);
L2=[L2 rough_param];
[A2]=feature_extraction_wavelet(ca2,ch2,cv2,cd2)
cod_ca2 = wcodemat(ca2,nbcol);
surf(cod_ca2)
ylabel('Rows=64')
xlabel('Columns=64')
zlabel('Coefficients')
title('Approximation Level-2')
figure
cod_ch2 = wcodemat(ch2,nbcol);
surf(cod_ch2)
ylabel('Rows=64')
xlabel('Columns=64')
zlabel('Coefficients')
title('Horizontal Level-2')
figure
cod_cv2 = wcodemat(cv2,nbcol);
surf(cod_cv2)
ylabel('Rows=64')
xlabel('Columns=64')
zlabel('Coefficients')
title('Vertical Level-2')
figure
cod_cd2 = wcodemat(cd2,nbcol);
surf(cod_cd2)
ylabel('Rows=64')
xlabel('Columns=64')
zlabel('Coefficients')
title('Diagonal Level-2')
figure
image([[cod_ca2,cod_ch2;cod_cv2,cod_cd2],cod_ch1;cod_cv1,cod_cd1]);
title('Second level decomposition')
close all

%%
%%%third level decomposition
figure
[ca3,ch3,cv3,cd3] = dwt2(ca2,'db1');
[rough_param]=calculate_roughness_L3A(ca3);
L3=rough_param;
figure
[rough_param]=calculate_roughness_L3H(ch3);
L3=[L3 rough_param];
figure
[rough_param]=calculate_roughness_L3V(cv3);
L3=[L3 rough_param];
figure
[rough_param]=calculate_roughness_L3D(cd3);
L3=[L3 rough_param];
[A3]=feature_extraction_wavelet(ca3,ch3,cv3,cd3)
cod_ca3 = wcodemat(ca3,nbcol);
surf(cod_ca3)
ylabel('Rows=32')
xlabel('Columns=32')
zlabel('Coefficients')
title('Approximation Level-3')
figure
cod_ch3 = wcodemat(ch3,nbcol);
surf(cod_ch3)
ylabel('Rows=32')
xlabel('Columns=32')
zlabel('Coefficients')
title('Horizontal Level-3')
figure
cod_cv3 = wcodemat(cv3,nbcol);
surf(cod_cv3)
ylabel('Rows=32')
xlabel('Columns=32')
zlabel('Coefficients')
title('Vertical Level-3')
figure
cod_cd3 = wcodemat(cd3,nbcol);
surf(cod_cd3)
ylabel('Rows=32')
xlabel('Columns=32')
zlabel('Coefficients')
title('Diagonal Level-3')
figure
image([[cod_ca3,cod_ch3;cod_cv3,cod_cd3],cod_ch2;cod_cv2,cod_cd2],
cod_ch1;cod_cv1,cod_cd1));
title('Third level decomposition')
close all
figure
[ca4,ch4,cv4,cd4] = dwt2(ca3,'db1');
[rough_param]=calculate_roughness_L4A(ca4);
L4=rough_param;
figure
[rough_param]=calculate_roughness_L4H(ch4);
L4=[L4 rough_param];
figure
[rough_param]=calculate_roughness_L4V(cv4);
L4=[L4 rough_param];
figure
[rough_param]=calculate_roughness_L4D(cd4);
L4=[L4 rough_param];
[A4]=feature_extraction_wavelet(ca4,ch4,cv4,cd4)
cod_ca4 = wcodemat(ca4,nbcol);
surf(cod_ca4)
ylabel('Rows=16')
xlabel('Columns=16')
zlabel('Coefficients')
title('Approximation Level-4')
figure
cod_ch4 = wcodemat(ch4,nbcol);
surf(cod_ch4)
ylabel('Rows=16')
xlabel('Columns=16')
zlabel('Coefficients')
title('Horizontal Level-4')
figure
cod_cv4 = wcodemat(cv4,nbcol);
surf(cod_cv4)
ylabel('Rows=16')
xlabel('Columns=16')
zlabel('Coefficients')
title('Vertical Level-4')
figure
cod_cd4 = wcodemat(cd4,nbcol);
surf(cod_cd4)
ylabel('Rows=16')
xlabel('Columns=16')
zlabel('Coefficients')
title('Diagonal Level-4')
figure
image([[[[cod_ca4,cod_ch4;cod_cv4,cod_cd4],cod_ch3;cod_cv3,cod_cd3],cod_ch2;cod_cv2,cod_cd2],cod_ch1;cod_cv1,cod_cd1])
title('Fourth level decomposition')
close all
figure
[ca5,ch5,cv5,cd5] = dwt2(ca4,'db1');
[rough_param]=calculate_roughness_L5A(ca5);
L5=rough_param;
figure
[rough_param]=calculate_roughness_L5H(ch5);
L5=[L5 rough_param];
figure
[rough_param]=calculate_roughness_L5V(cv5);
L5=[L5 rough_param];
figure
[rough_param]=calculate_roughness_L5D(cd5);
L5=[L5 rough_param];
[A5]=feature_extraction_wavelet(ca5,ch5,cv5,cd5)
savecoeffi_A1 A2 A3 A4 A5
save A1.txt A1 -ascii
save A2.txt A2 -ascii
save A3.txt A3 -ascii
save A4.txt A4 -ascii
save A5.txt A5 -ascii
cod_ca5 = wcodemat(ca5,nbcol);
surf(cod_ca5)
ylabel('Rows=8')
xlabel('Columns=8')
zlabel('Coefficients')
title('Approximation Level-5')
figure
cod_ch5 = wcodemat(ch5,nbcol);
surf(cod_ch5)
ylabel('Rows=8')
xlabel('Columns=8')
zlabel('Coefficients')
title('Horizontal Level-5')
figure
cod_cv5 = wcodemat(cv5,nbcol);
surf(cod_cv5)
ylabel('Rows=8')
xlabel('Columns=8')
zlabel('Coefficients')
title('Vertical Level-5')
figure
cod_cd5 = wcodemat(cd5,nbcol);
surf(cod_cd5)
ylabel('Rows=8')
xlabel('Columns=8')
zlabel('Coefficients')
title('Diagonal Level-5')
figure
image(
[[[cod_ca5,cod_ch5;cod_cv5,cod_cd5],
cod_ch4;cod_cv4,cod_cd4],
cod_ch3;cod_cv3,cod_cd3],
cod_ch2;cod_cv2,cod_cd2],
cod_ch1;cod_cv1,cod_cd1])
title('Fifth level decomposition')
close all
% storing patterns
figure
plot(Max_min_ca(1,:),'-r')
hold on
plot (Max_min_ca(2,:),'-g')
legend('Maximum','Minimum')
Title('Approximation at all five levels of decomposition')
ylabel('Coefficients')
xlabel('Levels 1-5')
figure
Max_min_ch=[A1(7:8,:) A2(7:8,:) A3(7:8,:) A4(7:8,:)];
plot (Max_min_ch(1,:),'-r')
hold on
plot (Max_min_ch(2,:),'-g')
legend('Maximum','Minimum')
Title('Horizontal at all five levels of decomposition')
ylabel('Coefficients')
xlabel('Levels 1-5')
figure
Max_min_cv=[A1(11:12,:) A2(11:12,:) A3(11:12,:) A4(11:12,:)];
plot (Max_min_cv(1,:),'-r')
hold on
plot (Max_min_cv(2,:),'-g')
legend('Maximum','Minimum')
Title('Vertical at all five levels of decomposition')
ylabel('Coefficients')
xlabel('Levels 1-5')
figure
plot (Max_min_cd(1,:),'-r')
hold on
plot (Max_min_cd(2,:),'-g')
legend('Maximum','Minimum')
Title('Diagonal at all five levels of decomposition')
ylabel('Coefficients')
xlabel('Levels 1-5')
save features Max_min_ca
function [A]=feature_extraction_wavelet(ca,ch,cv,cd)
A1=mean(ca);%mean Approximation
A2=std(ca);%std
% A3=norm(ca);%norm
A4=max(ca); %maximum
A5=min(ca);%minimum
A6=mean(ch);,%mean Detail
A7=std(ch);%std
% A8=norm(ch);%norm
A9=max(ch); %maximum
A10=min(ch);%minimum
A11=mean(cv);%mean Detail
A12=std(cv);%std
% A13=norm(cv);%norm
A14=max(cv); %maximum
A15=min(cv);%minimum
A16=mean(ch);,%mean Detail
A17=std(cd);%std
% A18=norm(cd);%norm
A19=max(cd); %maximum
A20=min(cd);%minimum
Aone=[A1;A2;A4;A5];
Atwo=[A6;A7;A9;A10];
Athree=[A11;A12;A14;A15];
Afour=[A16;A17;A19;A20];
A=[Aone;Atwo;Athree;Afour];
function [rough_param]=calculate_roughness_L1A(images)
    [a,b,c]=size(images);
    PSF = fspecial('gaussian',100,20);%answer(1),answer(2));
    i=1;
    waviness(:,:,i) = imfilter(images(:,:,i),PSF,'conv');
    roughness(:,:,i)=images(:,:,i)-waviness(:,:,i);
    n=a*b;
    fori=1:c
        vectorR(:,i)=reshape(roughness(:,:,i), [1,n]);
        vectorW(:,i)=reshape(waviness(:,:,i), [1,n]);
        RaR(i)=sum(sum(abs(roughness(:,:,i))))/n; %average roughness
        RqR(i)=sqrt(mean(vectorR(:,i).^2)); %RMS roughness
        RskR(i)=sum(sum(roughness(:,:,i).^3))/(n*RqR(i)^3); %skewness
        RkuR(i)=sum(sum(roughness(:,:,i).^4))/(n*RqR(i)^4); % kurtosis
        RaW(i)=sum(sum(abs(waviness(:,:,i))))/n; %average roughness
        RqW(i)=sqrt(mean(vectorW(:,i).^2)); %RMS roughness
        RskW(i)=sum(sum(waviness(:,:,i).^3))/(n*RqW(i)^3); %skewness
        RkuW(i)=sum(sum(waviness(:,:,i).^4))/(n*RqW(i)^4); % kurtosis
    end
    rough_param(:,1)=RaR;
    rough_param(:,2)=RqR;
    rough_param(:,3)=RskR;
    rough_param(:,4)=RkuR;
    rough_param(:,5)=RaW;
    rough_param(:,6)=RqW;
    rough_param(:,7)=RskW;
    rough_param(:,8)=RkuW;
    saveall_datarough_param roughness waviness images
loadall_data
figure
imagesc(waviness(:,:,1)), colormap(gray)
title ('Waviness Level-1, Approximation')
figure
imagesc(roughness(:,:,1)), colormap(gray)
title ('Roughness Level-1 Approximation')
figure
plot(mean(waviness(:,:,1)),'-r')
hold on
plot(mean(roughness(:,:,1)), 'g')
legend('Roughness','Waviness')
title(' Level-1 Approximation')