SPLINE TRANSFORMATION OF UNAFFECTED REGION:

```
% Load a 1D signal
%load 1Ddata
[x,fs]=audioread('1.wav');
x=(x(1:512))';
figure(1),plot(x)
set(gca,'fontsize',14)
title('Original signal')
M=length(x);      % Should be a power of 2

% Choose a fractional spline wavelet transform
alpha=2.4;         % Real value larger than -0.5
tau=0.3;           % Real value between -0.5 and +0.5
type='ortho';      % Options are:  1. 'bspline'
                   %               2. 'ortho'
                   %               3. 'dual'
[FFTanalysisfilters,FFTsynthesisfilters]=FFTfractsplinefilters(M,alpha,tau,type);

% Perform a fractional spline wavelet transform of the signal
J=4;               % Number of decomposition levels
w=FFTwaveletanalysis1D(x,FFTanalysisfilters,J);

% Show a subband
bandnumber=2;
titletext=['Subband ' num2str(bandnumber)];
b0=wextract1D(w,J,bandnumber);
```

```
figure(2),clf,plot(b0),colormap(gray(256))
set(gca,'fontsize',14)
title(titletext)

% Show the whole wavelet transform
figure(3),clf,waveletplot1D(w,J)
set(gca,'fontsize',14)
title({['Fractional Spline Wavelet Transform (' num2str(J) ' levels)'],[type '-type,
alpha=' num2str(alpha) ', tau=' num2str(tau)]})

% Reconstruction of the signal from its wavelet transform
x0=FFTwaveletsynthesis1D(w,FFTsynthesisfilters,J);
disp(['Resynthesis error: ' num2str(max(abs(x0(:)-x(:))))])

% Reconstruction from only one subband: use of the second output parameter
% from the function wextract2D.m
bandnumber=J+1;
titletext=['Image resynthetized from the lowpass band at depth ' num2str(J)];
[b0,w0]=wextract1D(w,J,bandnumber);
x0=FFTwaveletsynthesis1D(w0,FFTsynthesisfilters,J);
figure(1),a=axis;
figure(4),clf,plot(x0),colormap(gray(256))
set(gca,'fontsize',14)
title(titletext)
axis(a)

% Plot of the synthesis scaling function and the wavelet
```

```
[x,y1]=fractsplinefunction(alpha,tau,type,10);

[x,y2]=fractsplinewaveletfunction(alpha,tau,type,10);

figure(5),plot(x,y1,'b',x,y2,'r')

set(gca,'fontsize',14)

legend('scaling function','wavelet')

title([type '-type, alpha=' num2str(alpha) ', tau=' num2str(tau)])
```

SPLINE TRANSFORMATION OF AFFECTED REGION:

```
% Load a 1D signal
%load 1Ddata
[x,fs]=audioread('2.wav');
x=(x(1:512))';
figure(1),plot(x)
set(gca,'fontsize',14)
title('Original signal')
M=length(x);      % Should be a power of 2

% Choose a fractional spline wavelet transform
alpha=2.4;         % Real value larger than -0.5
tau=0.3;           % Real value between -0.5 and +0.5
type='ortho';      % Options are:  1. 'bspline'
                   %               2. 'ortho'
                   %               3. 'dual'
[FFTanalysisfilters,FFTsynthesisfilters]=FFTfractsplinefilters(M,alpha,tau,type);

% Perform a fractional spline wavelet transform of the signal
J=4;               % Number of decomposition levels
w=FFTwaveletanalysis1D(x,FFTanalysisfilters,J);

% Show a subband
bandnumber=2;
titletext=['Subband ' num2str(bandnumber)];
b0=wextract1D(w,J,bandnumber);
figure(2),clf,plot(b0),colormap(gray(256))
set(gca,'fontsize',14)
title(titletext)
```

```
% Show the whole wavelet transform
figure(3),clf,waveletplot1D(w,J)
set(gca,'fontsize',14)
title({['Fractional Spline Wavelet Transform (' num2str(J) ' levels)'],[type '-type,
alpha=' num2str(alpha) ', tau=' num2str(tau)]})

% Reconstruction of the signal from its wavelet transform
x0=FFTwaveletsynthesis1D(w,FFTsynthesisfilters,J);
disp(['Resynthesis error: ' num2str(max(abs(x0(:)-x(:))))])

% Reconstruction from only one subband: use of the second output parameter
% from the function wextract2D.m
bandnumber=J+1;
titletext=['Image resynthetized from the lowpass band at depth ' num2str(J)];
[b0,w0]=wextract1D(w,J,bandnumber);
x0=FFTwaveletsynthesis1D(w0,FFTsynthesisfilters,J);
figure(1),a=axis;
figure(4),clf,plot(x0),colormap(gray(256))
set(gca,'fontsize',14)
title(titletext)
axis(a)

% Plot of the synthesis scaling function and the wavelet
[x,y1]=fractsplinefunction(alpha,tau,type,10);
[x,y2]=fractsplinewaveletfunction(alpha,tau,type,10);
figure(5),plot(x,y1,'b',x,y2,'r')
set(gca,'fontsize',14)
legend('scaling function','wavelet')
title([type '-type, alpha=' num2str(alpha) ', tau=' num2str(tau)])
```

DIAGNOSIS:

```
% read two images
Image1 = audioread(uigetfile); % Image 1
Image2 = audioread(uigetfile); % Image 2
%  convert images to type double (range from from 0 to 1 instead of from 0 to 255)
Imaged1 = (Image1);
Imaged2 = (Image2);
% reduce three channel [ RGB ]  to one channel [ grayscale ]
```

```
Imageg1 = (Imaged1);
Imageg2 = (Imaged2);
% Calculate the Normalized Histogram of Image 1 and Image 2
hn1 = imhist(Imageg1)./numel(Imageg1);
hn2 = imhist(Imageg2)./numel(Imageg2);
 %subplot(2,2,1);plot(Image1)
 %subplot(2,2,2);subimage(Image2)
  subplot(2,1,1);plot(hn1)
  title('Signal comparison');
 subplot(2,1,2);plot(hn2)
 title('Signal variation');
% Calculate the histogram error
f = sum((hn1 - hn2).^2);
disp(f) %display the result to console
```