Appendix B: MATLAB script for spectrogram

```
%% get file names

[file_pc,path_pc] = uigetfile('*.bin','Ultrasound file');

%% load ultrasound data

fid = fopen([path_pc,file_pc]);            % data file
data_pc = fread(fid,2e7,'uint16','b');
fclose(fid);
data_info = importdata([path_pc,file_pc(1:end-3),'txt'],' ',0); % additional info file

% get the info about data
dt = data_info(2)*1e-9;
fs = 1/dt;
N_samp = data_info(3);
N_line = length(data_pc)/N_samp;
rfpc = reshape(data_pc,N_samp,N_line);
t = (1:N_samp)*dt;

% filtering
[bpc,apc]=butter(2,[1e6 1e7]*dt/2);
rfpc_f = rfpc-mean(rfpc,2)*ones(1,N_line,1);
rfpc_f = filtfilt(bpc,apc,rfpc_f.*tukeywin(N_samp,0.1));

%....get the echo envelope (abs(hilbert(), display in dB)
rfpc_fh = hilbert(rfpc_f);
rfpc_fhe = abs(rfpc_fh);



afig = figure;
imagesc(1:N_line,t*1e6,20*log10(rfpc_fhe))
colorbar

ylabel('t   (\mus)')
xlabel('Line no')
title(['Ultrasound data ',strrep(file_pc,'_','\_')])
set(gca,'YDir',"normal")
[cmi,cma]=caxis;
caxis([round((cma-50)/5)*5, cma])          % use e.g.,50dB dynamic range

[maxval,indpc_max] = max(sum(abs(rfpc_fh)));
[minval,indpc_min] = min(sum(abs(rfpc_fh)));


[peakX,peakY] = pick_interface(t,rfpc_fhe,[6,21*15],5,10e-6);

figure(afig), hold on
plot(peakX, peakY*1e6,'r.')%'rx')
```

```
function [allTraceX, allTracet] = pick_interface(t, demodDataFilt, smfilt, Peakprom, PeakDist)
% function [allTraceX, allTraceY] = pick_interface(t, demodDataFilt, smfilt, Peakprom, PeakDist)
%
%   function smoothing the image of the dataset and use MatLab's
%   function findpeaks to indentify surfaces/interfaces in the image
%
% t             - time axis
% demodDataFilt - filtered envelope of dataset
% smfilt        - smoothfilter length smfilt(Nx, Ny) (in samples, Nx - lines, Ny time in samples)
% Peakprom      - peak prominence (in dB)
% PeakDist      - minimum peak distance in time (s)
%
% allTraceX - peak positions, X - line no
% allTracet - peak positions, t - time (s)
%
% based on Jørgen Avdals procBinary
%  Tonni F.Johansen, 10.oct.2022
%

N_line = size(demodDataFilt,2);

% smooth filter parameters
Nx = smfilt(1);
Ny = smfilt(2);

fs = 1/diff(t(1:2));    %sampling frequency
PeakDistS = PeakDist*fs;    % peak distance in samples

demodData_af = filter2( ones( Ny,Nx)/(Ny*Nx), abs(demodDataFilt).^2);       % smoothing of the image

peakMask = zeros( size( demodData_af) );
for ii = 1:size( demodData_af,2)
   testSig = 10*log10( demodData_af(:,ii) );
   [pks,locs,w,p] = findpeaks(testSig, 'MinPeakProminence', Peakprom, 'MinPeakDistance', PeakDistS );
   peakMask(locs,ii) = pks;
end
[allTraceY, allTraceX] = ind2sub( size( peakMask), find( peakMask) );

if 1
   figure, hold off, imagesc(1:N_line,t*1e6, 10*log10(demodData_af));
   [cmi,cma]=caxis;
   caxis([cma-15, cma])          % compressed dynbamic range
   hold on, plot( allTraceX, allTraceY/fs*1e6, 'r.'); hold off;

   xlabel('Line no')
   ylabel('t (\mus)')
   colorbar
   set(gca,'YDir',''normal'')
end
allTracet = allTraceY/fs;
```