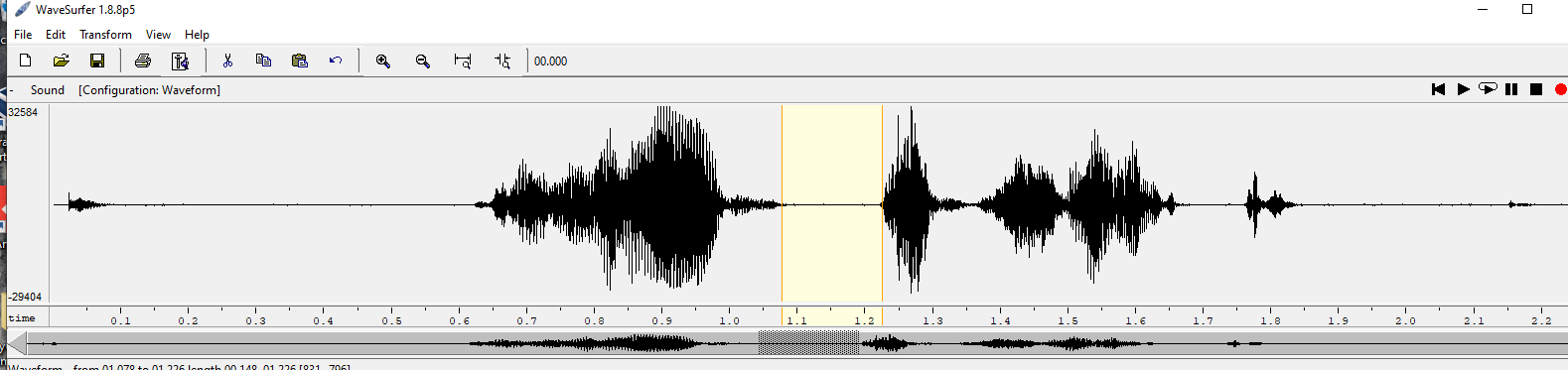
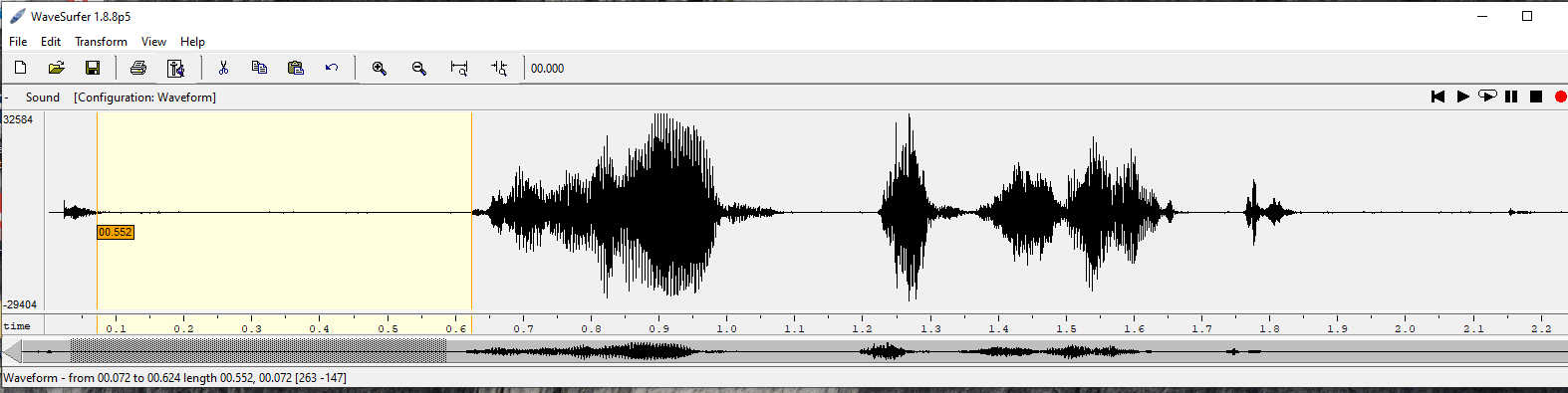


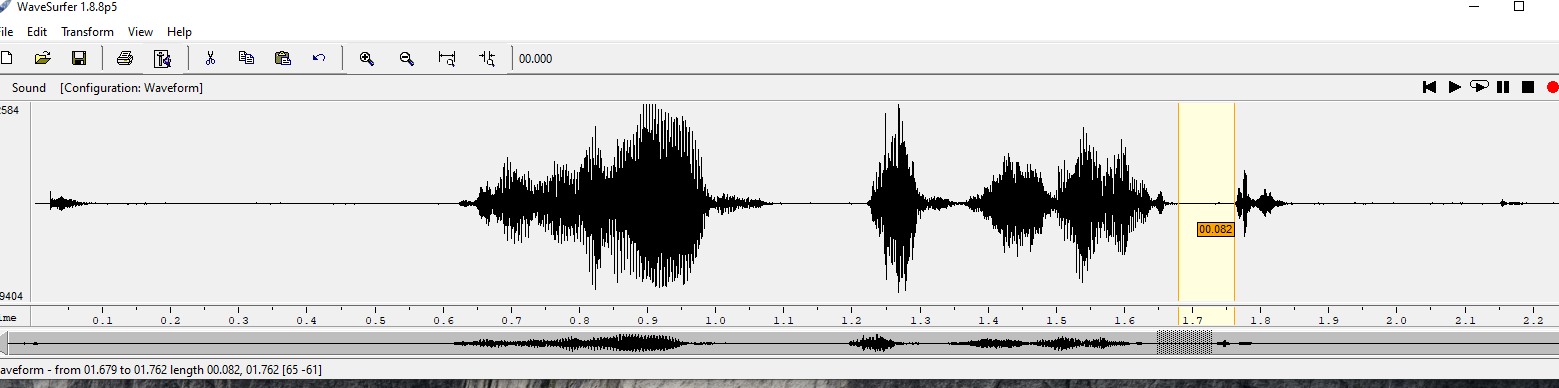
VOICED SIGNAL



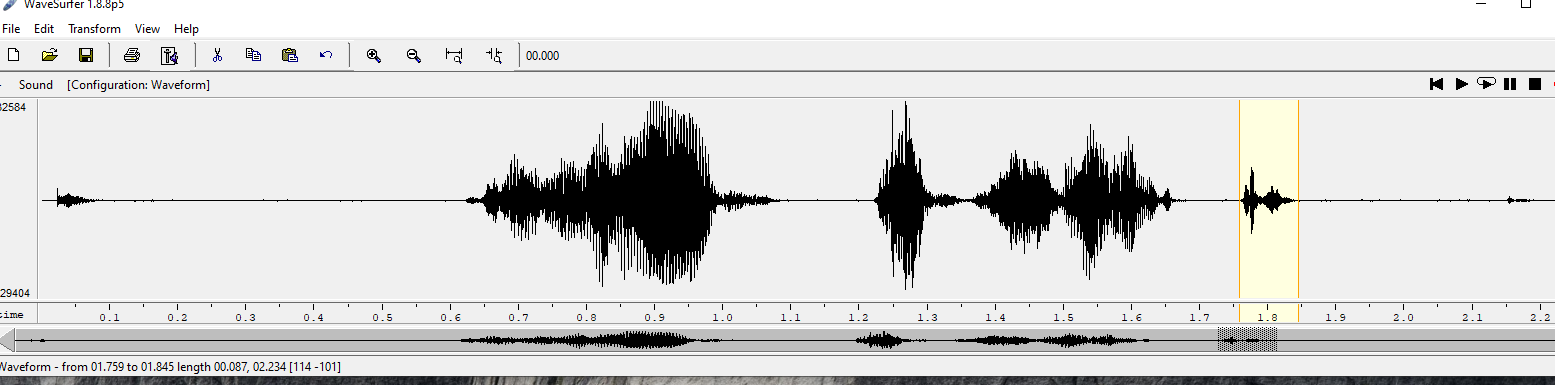
UNVOICED SIGNAL



UNVOICED SIGNAL



UNVOICED SIGNAL



% Let’s load the speech signal first.

[s,fs] = wavread(’H.22.16k.wav’);

frame1 = s(3600:4400);

N1 = length(frame1);

figure; plot(0:1/fs:N1/fs-1/fs, frame1); grid; title(’A voiced frame’); xlabel(’time’);

% Listen if you want % soundsc(frame1, fs);

E1 = (1/N1)\*sum(frame1.^2);

frame2 = s(4800:5500);

N2 = length(frame2); figure;

plot(0:1/fs:N2/fs-1/fs, frame2); grid;

title(’An unvoiced frame’);

xlabel(’time’);

% Listen if you want %

soundsc(frame2, fs);

E2 = (1/N2)\*sum(frame2.^2);

stem(1, E1, ’+’, ’LineWidth’, 3); grid; hold on; stem(2, E2, ’rx’, ’LineWidth’, 3); hold off; axis([0 3 0 max(E1,E2)]); title(’Energy comparison’); xlabel(’Index’);

ylabel(’Energy Value’);

legend(’Energy of Voiced frame’, ’Energy of Unvoiced Frame’);

figure; subplot(2,1,1);

plot(frame1); grid;

title(’Voiced frame’); subplot(2,1,2); plot(frame2); grid;

title(’Unvoiced frame’);

ZCr1 = 0.5\*sum(abs(sign(frame1(2:end))-sign(frame1(1:end-1))));

ZCr2 = 0.5\*sum(abs(sign(frame2(2:end))-sign(frame2(1:end-1))));

; stem(1, ZCr1, ’+’, ’LineWidth’, 3); grid; hold on; stem(2, ZCr2, ’rx’, ’LineWidth’, 3); hold off; axis([0 3 0 max(ZCr1, ZCr2)]); title(’Zero-Crossings comparison’); xlabel(’Index’);

ylabel(’Zero-Crossings Value’);

legend(’Zero Crossings of Voiced frame’, ’Zero Crossings of Unvoiced Frame’);

% Reading the speech signal

[s, fs] = wavread(’sample.wav’);

% Remove mean value

(DC component) s = s - mean(s);

% Signal length

D = length(s); 8

% Frame

length (30 ms, how many samples? ) L = %INSERT CODE HERE % Frame

shift (10 ms, how many samples? ) U = %INSERT CODE HERE

% Windo

w type (Hamming) win = hamming(L);

% Number of frames Nfr = %INSERT CODE HERE % Memory allocation (for speed) energy = zeros(1, Nfr+1); ZCr = zeros(1, Nfr+1); % Loop which calculates the speech features for i = 1:1:(Nfr+1) frame = %INSERT CODE HERE % a frame of speech windowed by the Hamming window energy(i) = %INSERT CODE HERE % calculate energy ZCr(i) = %INSERT CODE HERE % calculate zero crossings T(i) = L/2 + (i-1)\*U; % Next analysis time instant end % THRESHOLDS (you can play with it!) Ethres = mean(energy)/2; ZCRthres = (3/2)\*mean(ZCr) - 0.3\*std(ZCr); % Clssification for i = 1:1:Nfr if % INSERT CONDITION HERE % VOICED VUS(i) = 1.0; elseif % INSERT CONDITION HERE % SILENCE VUS(i) = 0.0; elseif % INSERT CONDITION HERE %

UNVOICED VUS(i) = 0.5; end end %

Interpolation with interp1 VUSi = INSERT CODE HERE Visualize figure; t = 0:1/fs:length(s)/fs-1/fs; plot(t, VUSi); 9 hold on; plot(t, s/max(s), ’r’); hold off; xlabel(’Time (s)’); title(’Energy & Zero-Crossings Rate-based VUS discrimination’);

