
```
clear

close all

A = load('r07_edfm.mat');
B = load('r07_edfm.mat');
d = A.val(1,:);
d_T = d';

B2 = B.val(2,:);
B3 = B.val(3,:);
B4 = B.val(4,:);
B5 = B.val(5,:);
B_total = B2 + B3 + B4 + B5;
noisy_sig = B_total.*(1/4);
noisy_sig_T = noisy_sig';

Fs = 1000;
Ts = 1/Fs;
order = 4000;           %this is where you can adjust the order
mu = .99;               %this is where you can adjust mu (0.4 < mu <
    0.99)

lms = dsp.LMSFilter(order + 1, 'StepSize', mu, 'Method', 'Normalized
    LMS', 'WeightsOutputPort', true);

[y,e,w] = step(lms, noisy_sig_T, d_T);

figure(1)
plot(noisy_sig(1:2000))
title('noisy signal')

figure(2)
plot(d(1:2000))
title('desired output')

figure(3)
plot(y(1:2000))
title('filtered signal')

figure(4)
plot(e(1:2000))
title('error')

[qrs_amp_raw , qrs_i_raw , delay] = pan_tompkin(y,Fs,1);
[qrs_amp_raw2 , qrs_i_raw2 , delay2] = pan_tompkin(d,Fs,1);

total_loc = 0;
for i = 2:length(qrs_i_raw)
    range = abs(qrs_i_raw(1,i) - qrs_i_raw(1,i-1));
    total_loc = total_loc + range;
```

```

end

total_loc2 = 0;
for i = 2:length(qrs_i_raw2)
    range2 = abs(qrs_i_raw2(1,i) - qrs_i_raw2(1,i-1));
    total_loc2 = total_loc2 + range2;
end

mean_loc = total_loc/(length(qrs_i_raw) - 1);
mean_loc2 = total_loc2/(length(qrs_i_raw2) - 1);
bpm_abdomen = (60*1000)/mean_loc
bpm_direct = (60*1000)/mean_loc2

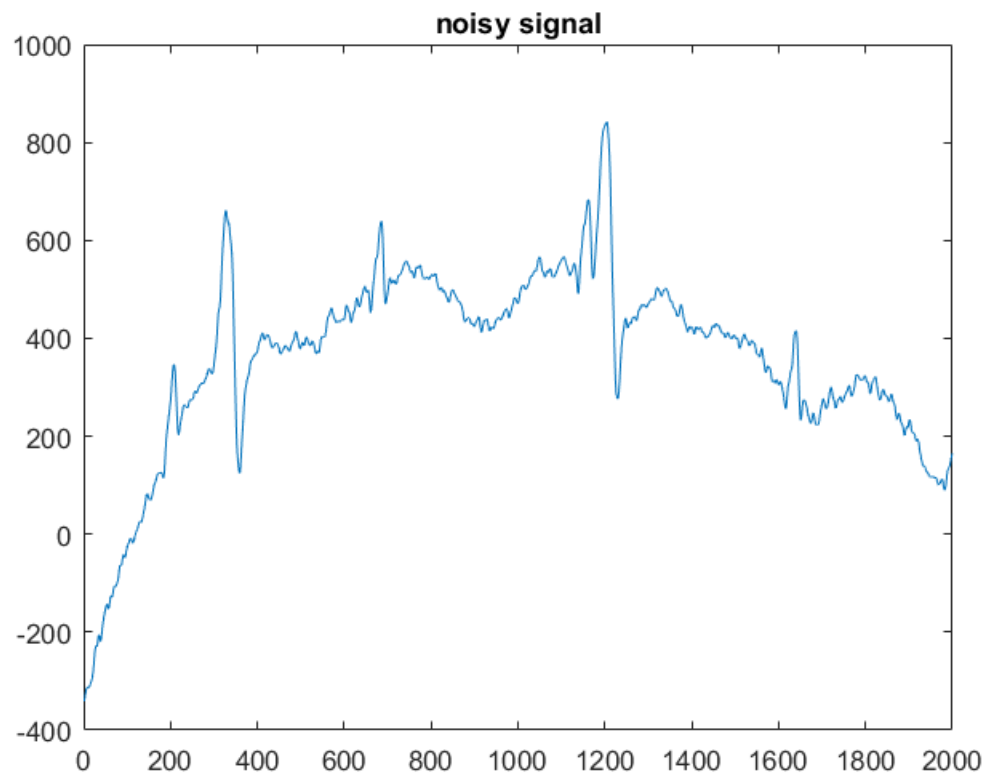
bpm_abdomen =

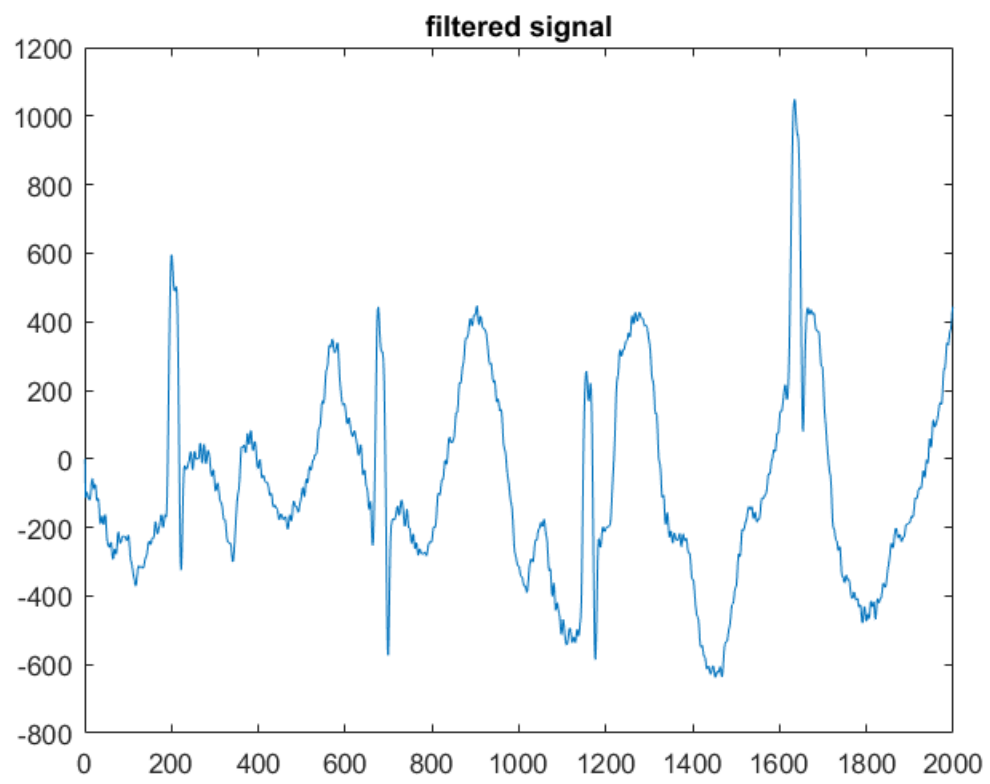
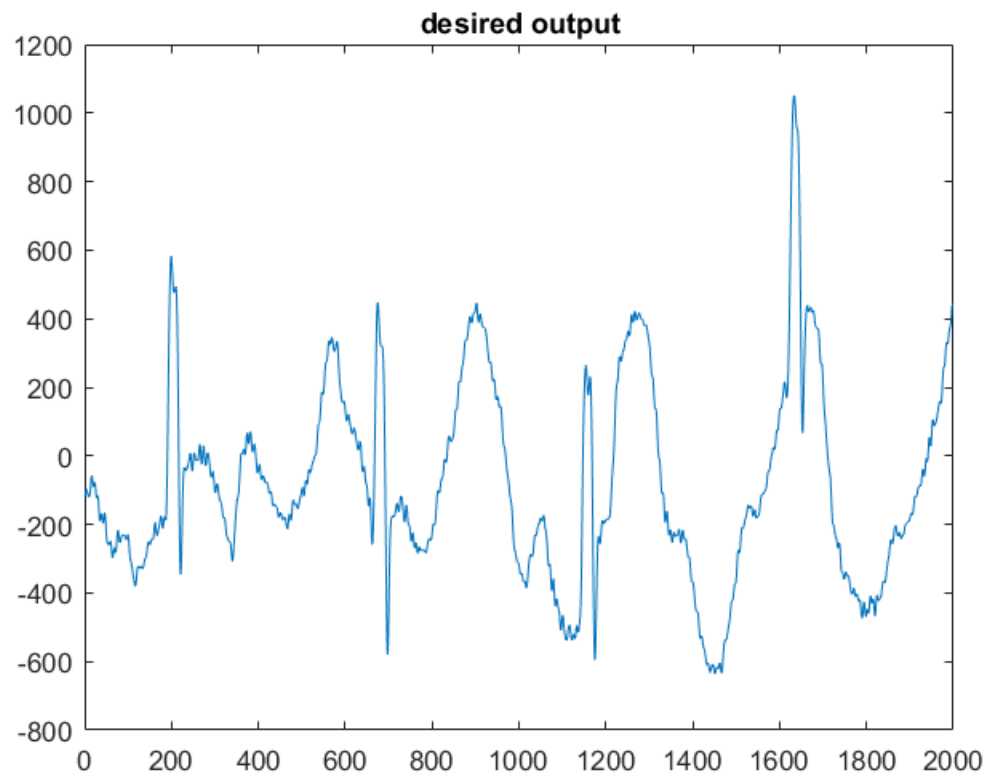
    126.0104

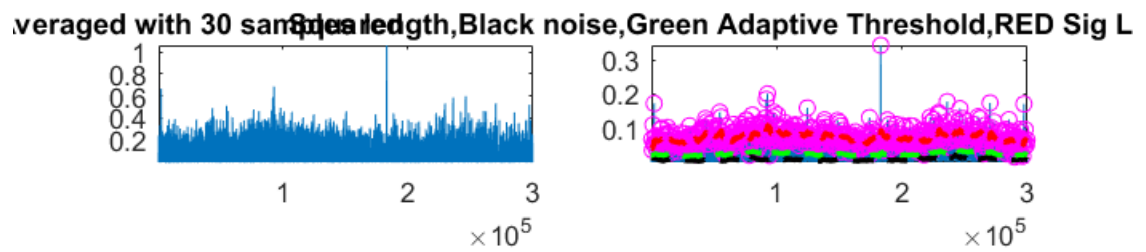
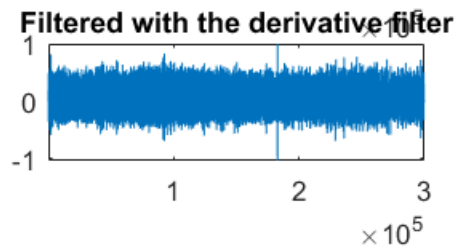
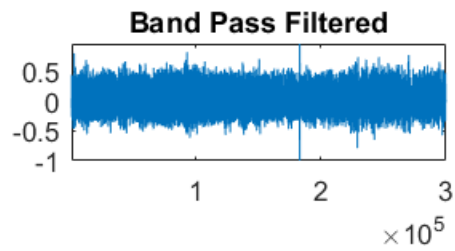
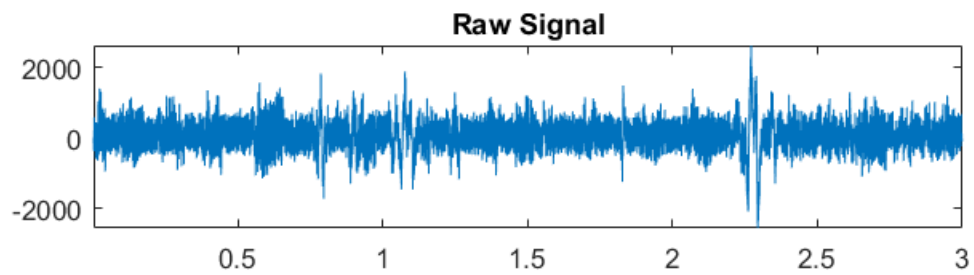
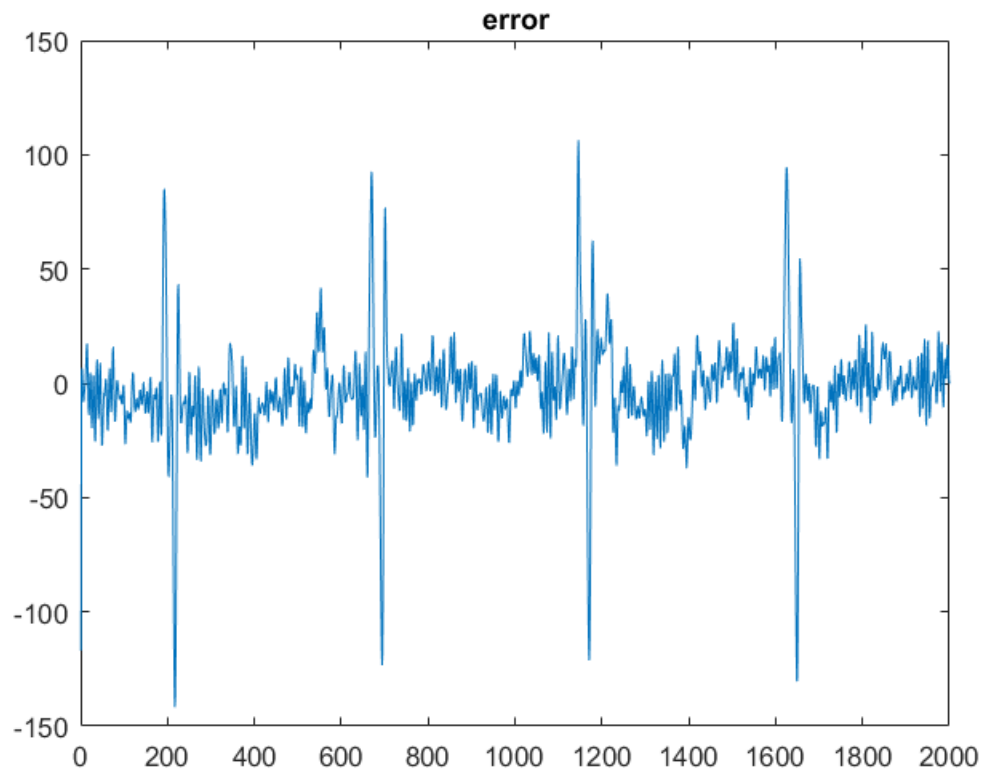
bpm_direct =

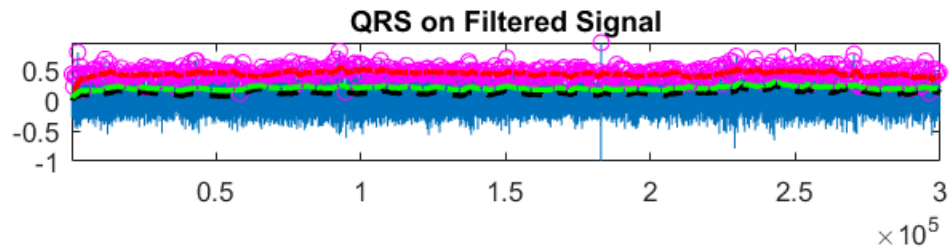
    125.2091

```

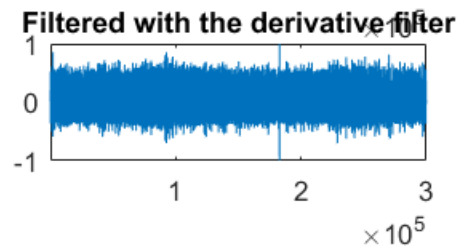
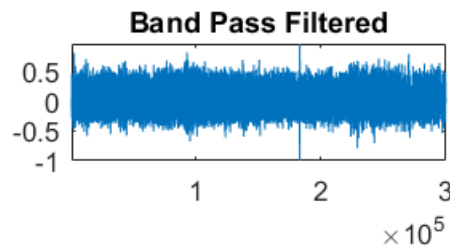
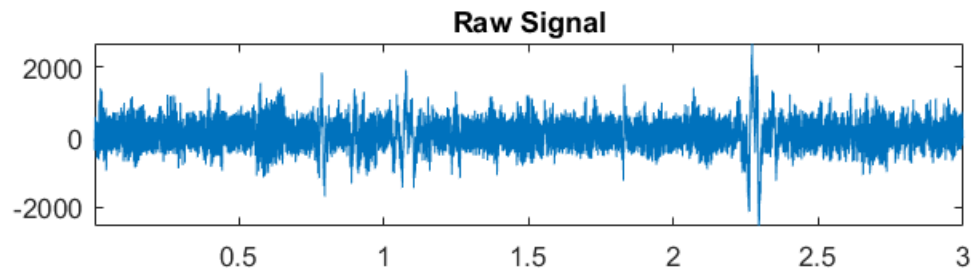
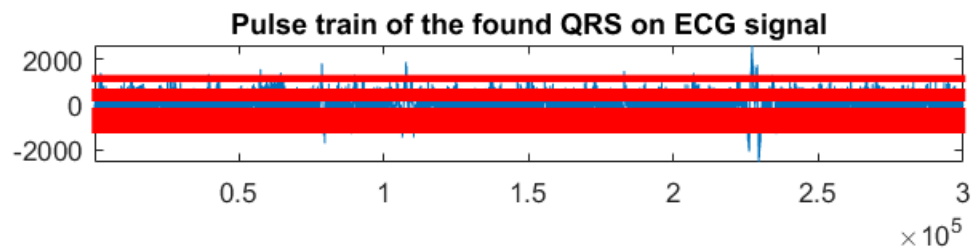
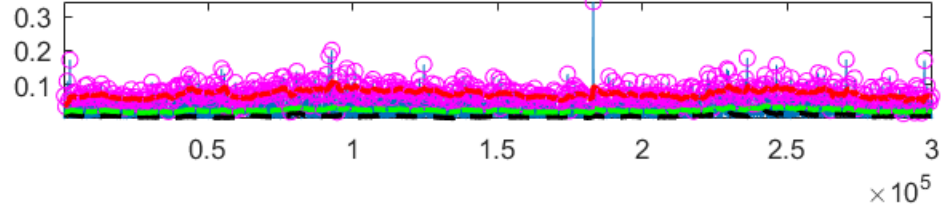




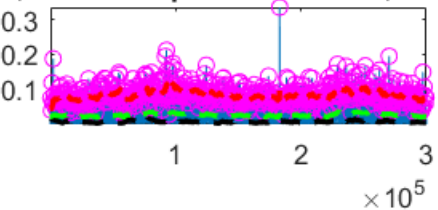
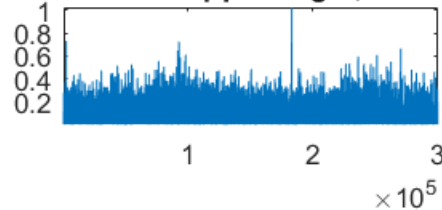


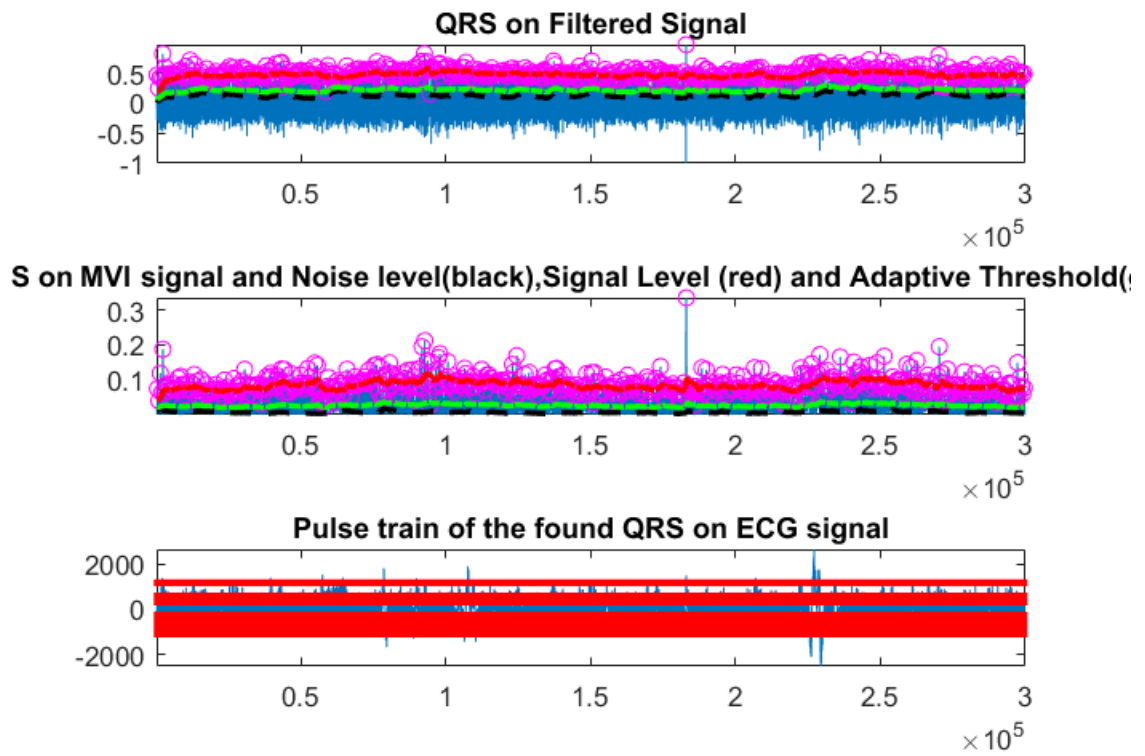


S on MVI signal and Noise level(black),Signal Level (red) and Adaptive Threshold(



veraged with 30 samples length,Black noise,Green Adaptive Threshold,RED Sig L





Published with MATLAB® R2016b