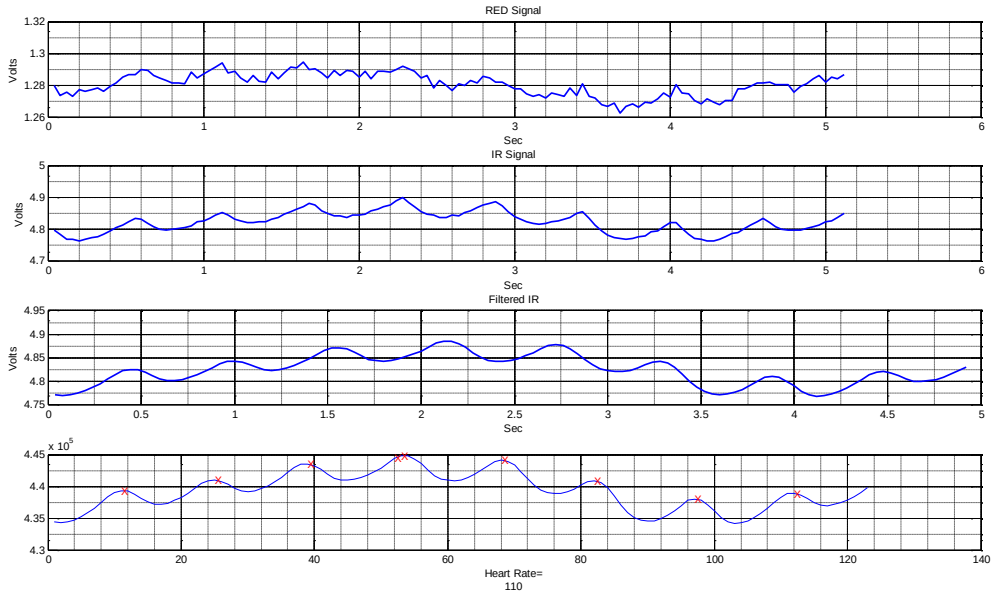
Odredjivanje pulsa iz vremenskog i frekventnog domena

IR PPG signala

1. VREMENSKI DOMEN

a. Nacrta se IR PPG signal u vremenskom domenu.
b. Odredi se rastojanje izmedju pikova PPi u sekundama.
c. Nadje se srednja vrijednost tog rastojanja PPsr u sekundama.
d. Nadje se frekvencija pulsa u sekundima kao fp=1/PPsr.
e. Nadje se frekvencija pulsa u minutima kao HR(otkucaja/min)=60*fp;
f. U prilogu je dat MATLAB program i odgovarajuci fajlovi odabrani sa frekvencijom 25Hz, RED (prva kolona) i IR(druga kolona)
g. Funkcija za pronalazenje lokalnih pikova je samo ilustrativna. STUDENT KOJI NAPRAVI POUZDANU FUNKCIJU KOJA CE BITI OTPORNA NA SMETNJE I RADICE BEZ OBZIRA NA FREKVENCIJU ODABIRANJA I VELICINU PROZORA DOBIJA POLA POENA NA KOLOKVIJUMU.
h. Upotrebljava se fajl data_9.dat, 25Hz frekvencija odabiranja i 20bitna rezolucija.



%% **HR from PPG in time domain**
% by Radovan Stojanovic
% THIS ONE IS NOT OPTIMIZED. DO NOT USE FOR COMMERCIAL
%% Start
clear all
%% Load sample data
clear all
fs=25; %sampling rate 25Hz
data_st = 150; % Data start
data_length = 128; % Data length

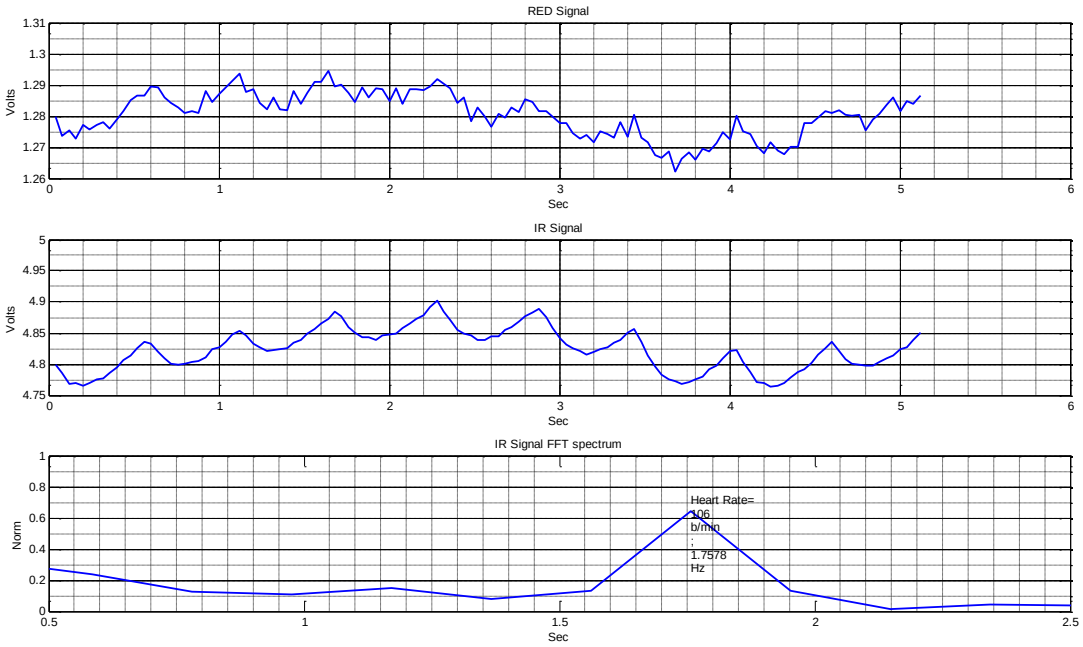
```
MAX_N=255*255*7; % 20 bit convertor for normalisation
load data_9.dat
X1=data_9(data_st:data_st+data_length-1,1);
X2=data_9(data_st:data_st+data_length-1,2);
%% Data input for Heart calculation
y1 = X1; %RED
y2 = X2; %IR
x=(1:1:data_length)/fs; %x axe in seconds

% Moving average filter during window fs/5
for i=1:(length(y2)-fs/5)
   local_sum=0;
   for j=1:fs/5
      local_sum=local_sum+y2(i+j);
   end
      y(i)=local_sum/(fs/5);
end

%plot signals in time domain
subplot(411) %plot RED
plot(x,y1*5/MAX_N,'LineWidth',2);
title('RED Signal');
xlabel('Sec');
ylabel('Volts');
xlabel('Sec');
grid minor

subplot(412) %plot IR
plot(x,y2*5/MAX_N,'LineWidth',2);
title('IR Signal');
xlabel('Sec');
ylabel('Volts');
xlabel('Sec');
grid minor

N_y=length(y);
x_y=(1:1:N_y)/fs;

subplot(413) %plot filtered IR

plot(x_y,y*5/MAX_N,'LineWidth',2);

title('Filtered IR');

xlabel('Sec');

ylabel('Volts');
```

```
xlabel('Sec');
text(2,2,'Heart Rate=')
grid minor
% FIND PEAKS NOT SURE THAT CORRECT WORKS, but it is principle of the work?
step=0;
w=round(0.5*fs) %window for finding a local peaks
delta_w=round(0.5*w) %for additional filtering distance between peaks
peak_i=1;
%plot diagram for ilustration of peaks finding
subplot(414);
plot(y);
grid minor
while step<=(length(y)-w)
   for i=1:w
      yy(i) = y(i+step);
   end

   local_i_max = 1;
   local_max = yy(local_i_max);

   for i=2:w
   if yy(i)>local_max
      local_i_max = i;
      local_max=yy(i);
   end
   end
pk(peak_i)=step+local_i_max; %pk is position of the peaks, indeces of the peaks
```

```
text(pk(peak_i),y(pk(peak_i)),'X','color','red'); %put X on found peaks
peak_i=peak_i+1;
step =step + w;
end
%second filtering of the peaks that are too close or too far
len_pk=length(pk);
for i=1:len_pk-1
   diff(i)=pk(i+1)-pk(i);
end
j=1;
for i=1:length(diff)
   if(diff(i)>=w-delta_w && diff(i)<=w+delta_w/5 )
      diff1(j)=diff(i);
      j=j+1;
   end
end
par_sum=0;
for i=1:length(diff1)
   par_sum=par_sum+diff1(i);
end
par_sum=par_sum/length(diff1);
hr_av=(fs/par_sum)*60
xlabel({'Heart Rate=',num2str(ceil(hr_av))});
%% END
```

2. FREKVETNI DOMEN

a. Nacrta se IR PPG signal u vremenskom domenu
b. Nadje se spektar signala u frekventnom domenu FFT
c. Nadje se dominantna frekvencija u vremenskom intervalu pulsa fp 0.5Hz-2.5Hz
d. Nadje se frekvencija pulsa u minutima kao HR(otkucaja/min) 60*fp
e. U prilogu je dat MATLAB program i odgovarajuçi fajlovi odabrani sa frekvencijom 25Hz, RED (prva kolona) i IR(druga kolona)
f. Upotrebljava se fajl data_9.dat, 25Hz frekvencija odabiranja i 20bitna rezolucija.




```
%% Exploit PPG signal to get heart rate from spectrum of IR
% by Radovan Stojanovic, stox@ucg.ac.me
% NOT FOR COMMERCIAL USE
%% Start
clear all
%% Load sample data, extract segment and normalize
clear all
fs=25; %sampling rate 25Hz
data_st = 150; % Data start in frame that read
data_length = 128; % Data length
MAX_N=255*255*7; % 20 bit A/D converter of data that read normalisation to
voltage
load data_9.dat
X1=data_9(data_st:data_st+data_length-1,1); %first colon
X2=data_9(data_st:data_st+data_length-1,2); %second colon

y1 = X1; %RED
y2 = X2; %IR
x=(1:1:data_length)/fs; %x axe in seconds

% Filtering moving average filter during window fs/5 for
for i=1:(length(y2)-fs/5)
    local_sum=0;
    for j=1:fs/5
```

```
        local_sum=local_sum+y2(i+j);
    end
        y(i)=local_sum/(fs/5);
end
%plot signals
subplot(311) %plot RED
plot(x,y1*5/MAX_N,'LineWidth',2);
title('RED Signal');
xlabel('Sec');
ylabel('Volts');
xlabel('Sec');
grid minor

subplot(312) %plot IR
plot(x,y2*5/MAX_N,'LineWidth',2);
title('IR Signal');
xlabel('Sec');
ylabel('Volts');
xlabel('Sec');
grid minor


FFT_size=128;
fm=fs/2; %max frequency by Nyquist
fc1=0.5; %lower heart rate frequency 30beats/min - 0.5 hz
fc2=2.5; %upper heart rate frequency 150Hz 2.5 hz
ind1=round(FFT_size*fc1/(fm)); %index in fft that corespond to lower freq
ind2=round(FFT_size*fc2/(fm)); %index in fft that corespond to higher freq


%% FFT Transform IR
NFFT = FFT_size; % Next power of 2 from length of y
Y2=abs(fft(y2-mean(y2),NFFT));
f2 = fm*linspace(0,1,NFFT/2+1);
MAX_Y2=max(Y2(1:NFFT/2+1)); %for normalisation to 1

subplot(313)
plot(f2,Y2(1:NFFT/2+1)/MAX_Y2,'LineWidth',2);
axis([fc1 fc2 0 1]) %normalised spectrum
title('IR Signal FFT spectrum');
xlabel('Sec');
ylabel('Norm');
xlabel('Sec')
grid minor


%find max value of abs FFT and index of this maximum value
max_am=0;
max_ind=0;
for i=ind1:ind2-1
    if Y2(i)>max_am
        max_am=Y2(i);
        max_ind=i;
    end
end
f_hr=(max_ind-1)*fs/FFT_size; %heart rate in Hz, Matlab started from 1 and
FFT from 0 (DC)
HR=f_hr*60; %HR in beats per sec
%subplot(414) %plot filtered IR
```

```
text(f_hr,0.5,{'Heart
Rate=',num2str(ceil(HR)),'b/min',';',num2str(f_hr),'Hz'})
%draw results

%% END
```