Contents

- Script for Data Analysis of Smartwatch-Controlled Robot
- Plotting Linear Position of the Robot from Motion Capture data
- Plotting Orientation of the Robot from Motion Capture data
- Plotting Linear Velocity of the Wrist on axis X, from Smartwatch Data
- Plotting Angular Velocity of the Wrist about axis Z, from Smartwatch Data
- 2D Plots for Visualization of Trajectories
- Plots of the behaviour of the driver's wrist in different conditions of motion

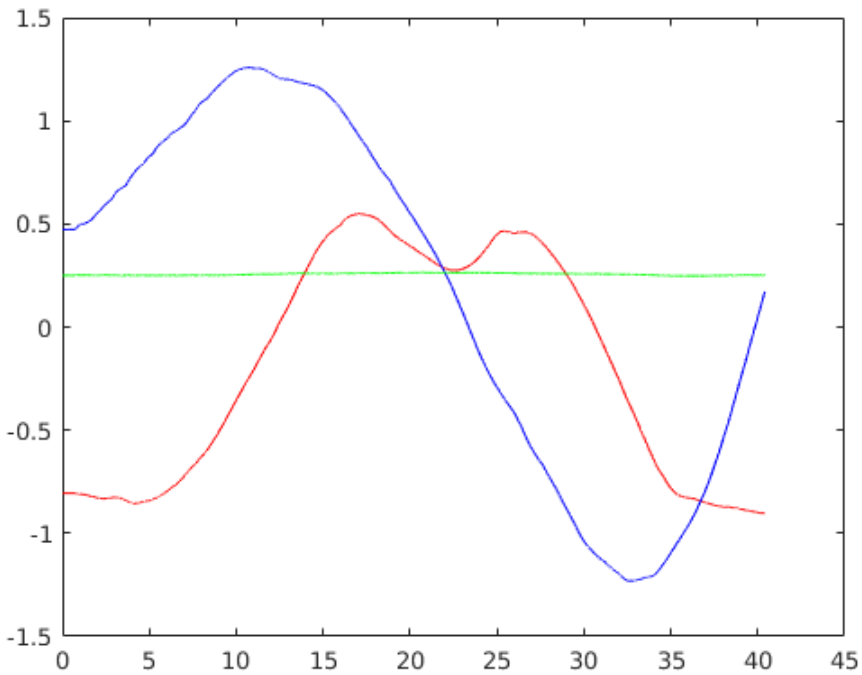
Script for Data Analysis of Smartwatch-Controlled Robot

This is the initial script chosen for data analysis, the user must select as current folder the one which contains the .mat files, then in this version the user can choose which file wants to analyze; by selecting the proper .mat file the data structures containing data of Motion Capture and Smartwatch sensors will be loaded in workspace

```
% To obtain the results, just load into workspace one of the .mat files
% which contain an experimental data set.
% We only kept 9 sets beacause we had to choose between the most regular
% trajectories and so the most reliable data.
```

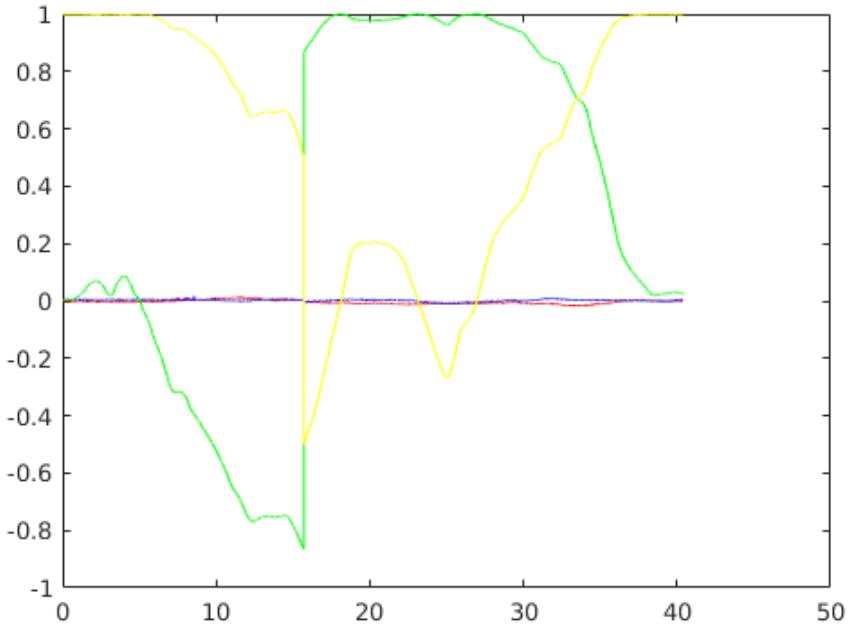
Plotting Linear Position of the Robot from Motion Capture data

```
plot(dataMoCap.time,dataMoCap.pos_x,'color','r','DisplayName','pos_X'); hold on;
plot(dataMoCap.time,dataMoCap.pos_y,'color','b','DisplayName','pos_Y'); hold on;
plot(dataMoCap.time,dataMoCap.pos_z,'color','g','DisplayName','pos_Z');
```



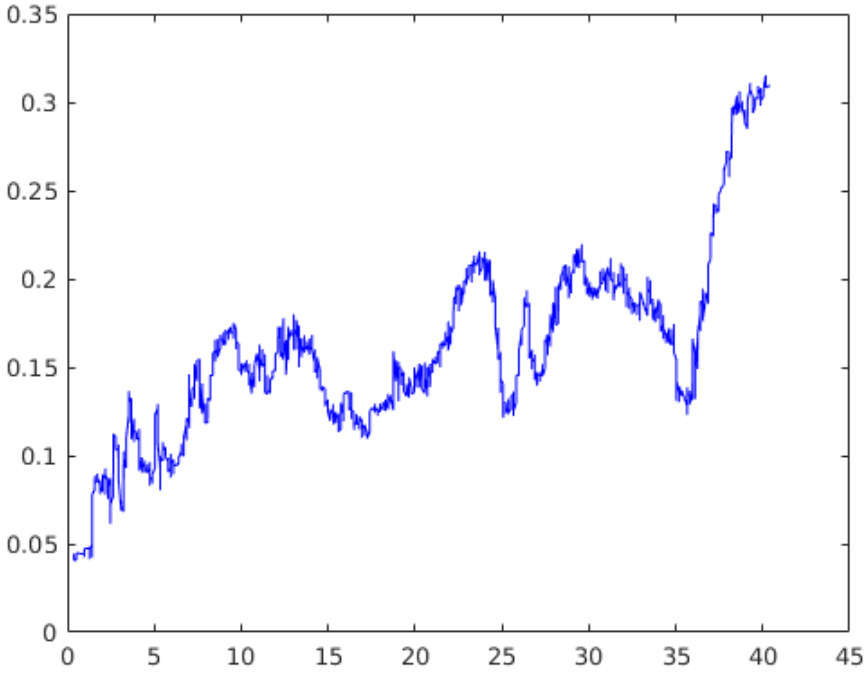
Plotting Orientation of the Robot from Motion Capture data

```
figure;
plot(dataMoCap.time,dataMoCap.ori_x,'color','r','DisplayName','ori_X'); hold on;
plot(dataMoCap.time,dataMoCap.ori_y,'color','b','DisplayName','ori_Y'); hold on;
plot(dataMoCap.time,dataMoCap.ori_z,'color','g','DisplayName','ori_Z'); hold on;
plot(dataMoCap.time,dataMoCap.ori_w,'color','y','DisplayName','ori_W');
```



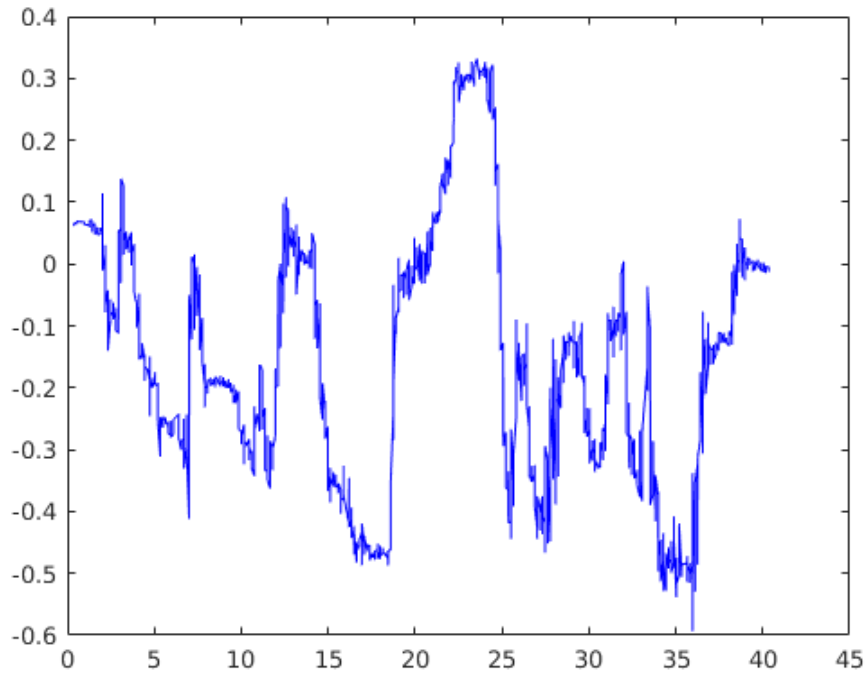
Plotting Linear Velocity of the Wrist on axis X, from Smartwatch Data

```
figure;
plot(dataCommand.time,dataCommand.l_vel_x,'color','b','DisplayName','Accel_X');
```



Plotting Angular Velocity of the Wrist about axis Z, from Smartwatch Data

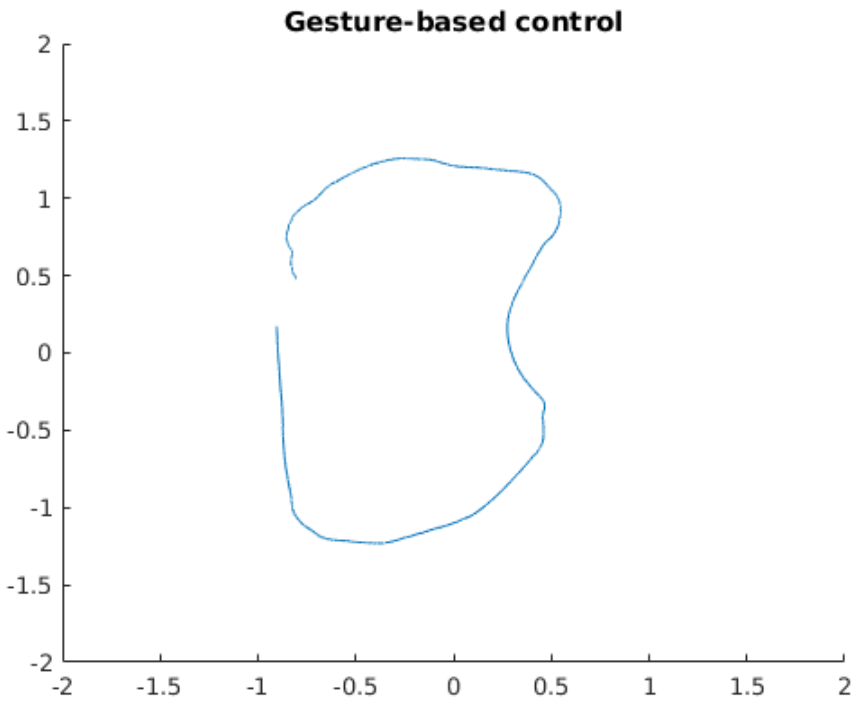
```
figure;
plot(dataCommand.time,dataCommand.a_vel_z,'color','b','DisplayName','AngVel_Z');
```



2D Plots for Visualization of Trajectories

```
%Using the script block given in Lab; Implementation of instructions for
```

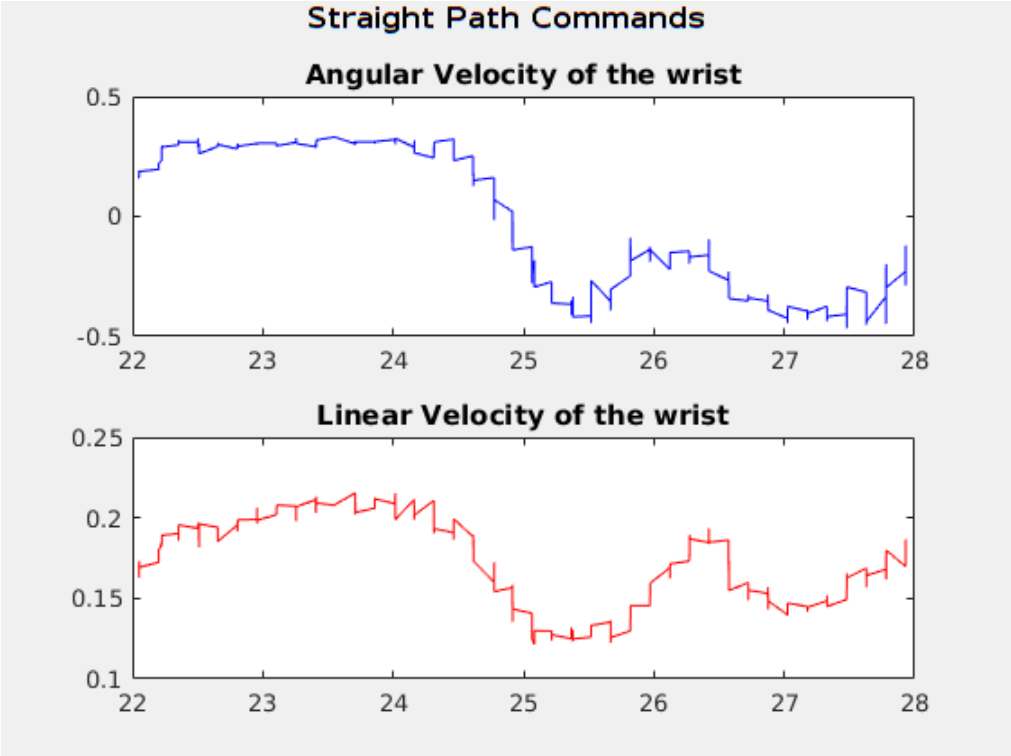
```
%graphical representation of trajectories followed by the robot during the tests.

figure;

    hold on;

    plot(dataMoCap.pos_x,dataMoCap.pos_y);

    title('Gesture-based control')
    axis([-2 2 -2 2])
```

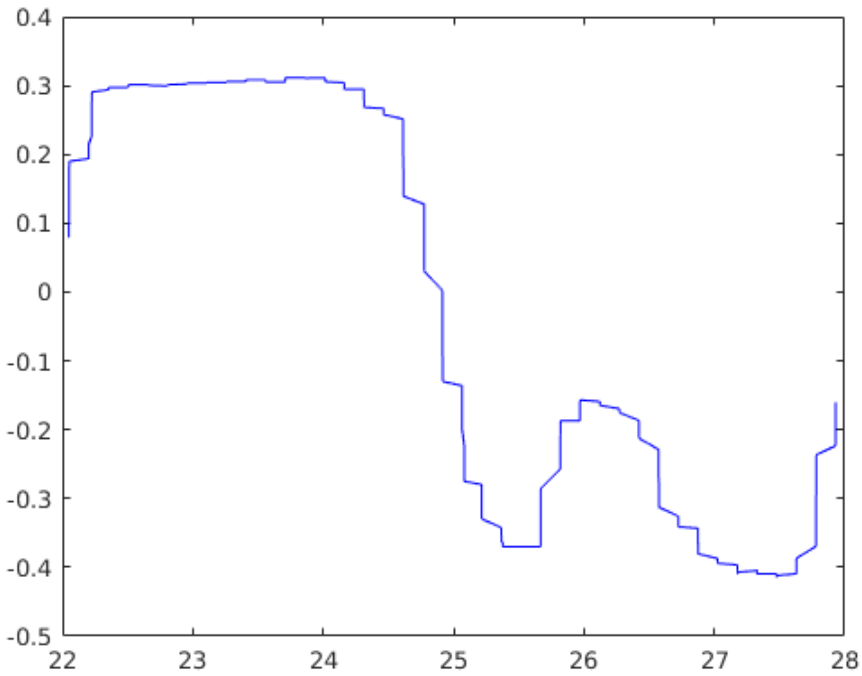


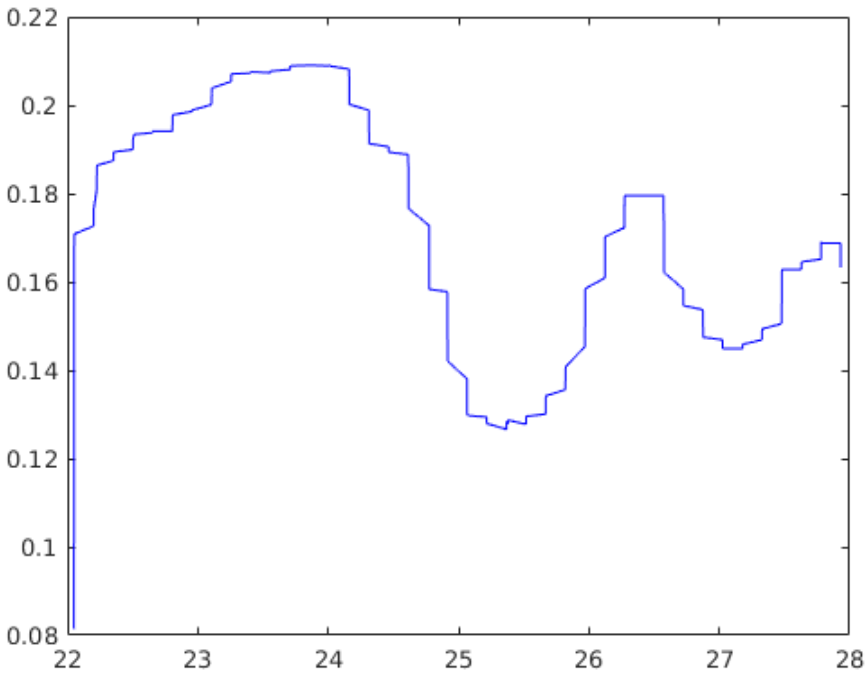
Plots of the behaviour of the driver's wrist in different conditions of motion

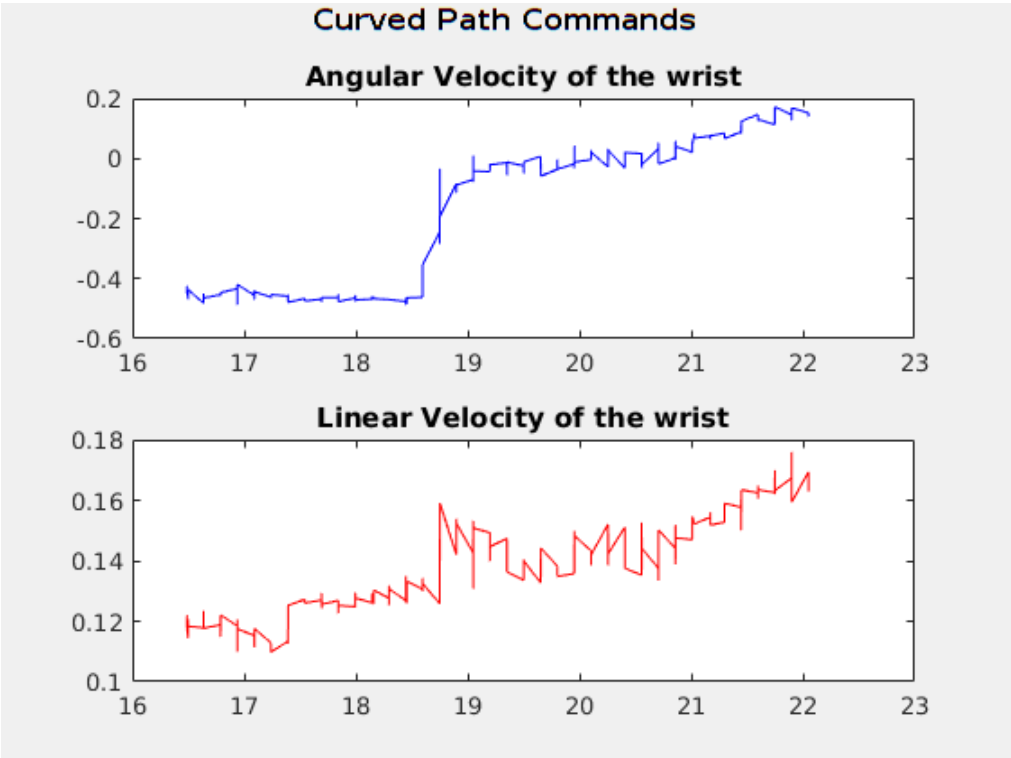
```
% Now we are evaluating the time range in which the robot is moving along
% an approximate straight line and in this range we provide plots of the
% motions registered from the smartwatch.

% This loop is used to save into arrays, for each data structure,
% the Linear and Angular velocities for different parts of the track.
% We took as example the range for the very first track, 'lap1_group1
% volunteer_1'.
time_rettilineo = zeros(241,1);
a_vel_z_rettilineo = zeros(241,1);
l_vel_x_rettilineo = zeros(241,1);
for i = 1:1:241 % This is the approximate range of values which define a straight path
   time_rettilineo(i) = dataCommand.time(886+i);
   a_vel_z_rettilineo(i) = dataCommand.a_vel_z(886+i);
   l_vel_x_rettilineo(i) = dataCommand.l_vel_x(886+i);
end

f = figure;
p = uipanel('Parent',f,'BorderType','none');
p.Title = 'Straight Path Commands';
p.TitlePosition = 'centertop';
```

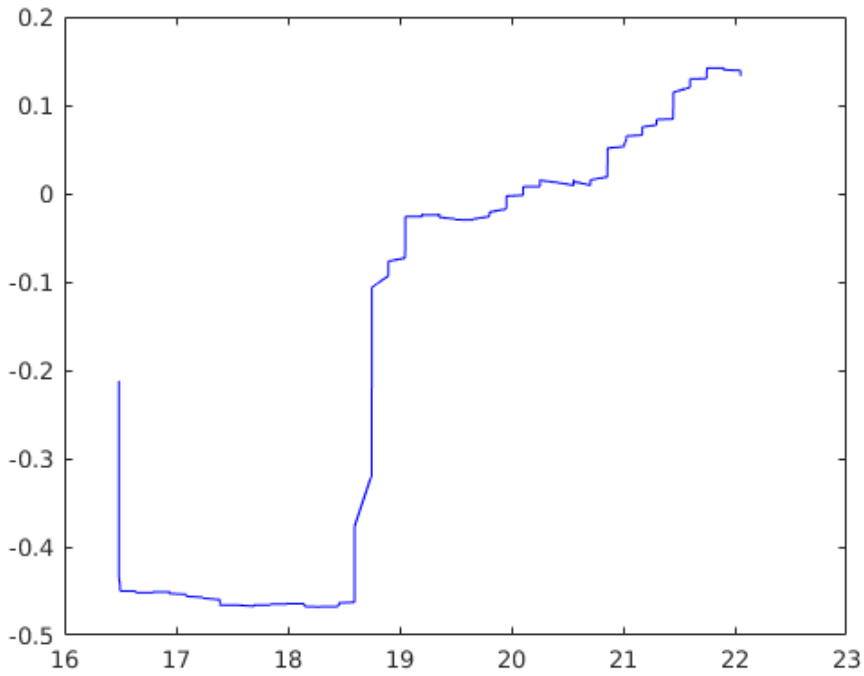
```
p.FontSize = 12;
p.FontWeight = 'bold';
sub1 = subplot(2,1,1,'Parent',p);
plot(sub1,time_rettilineo,a_vel_z_rettilineo,'color','b','DisplayName','AngVel_Z_rettilineo');
title('Angular Velocity of the wrist');
sub2 = subplot(2,1,2,'Parent',p);
plot(sub2,time_rettilineo,l_vel_x_rettilineo,'color','r','DisplayName','LinVel_X_rettilineo');
title('Linear Velocity of the wrist');

% Here we are using a Median Filter to cut off undesidered values from the
% plots
figure
a_vel_z_filtered = medfilt1(a_vel_z_rettilineo,20);
plot(time_rettilineo,a_vel_z_filtered,'color','b','DisplayName','AngVel_Z_filtered');
figure
l_vel_x_filtered = medfilt1(l_vel_x_rettilineo,20);
plot(time_rettilineo,l_vel_x_filtered,'color','b','DisplayName','LinVel_X_filtered');

% Now we evaluate Smartwatch Data on a curve instead of a straight motion
% This loop is used to save into arrays, for each data structure,
% the Linear and Angular velocities for different parts of the track.

time_curve = zeros(236,1);
a_vel_z_curve = zeros(236,1);
l_vel_x_curve = zeros(236,1);
for i = 1:1:236 % This is the approximate range of values for which is defined a curved path
   time_curve(i) = dataCommand.time(650+i);
   a_vel_z_curve(i) = dataCommand.a_vel_z(650+i);
   l_vel_x_curve(i) = dataCommand.l_vel_x(650+i);
end

f = figure;
p = uipanel('Parent',f,'BorderType','none');
p.Title = 'Curved Path Commands';
p.TitlePosition = 'centertop';
p.FontSize = 12;
p.FontWeight = 'bold';
sub1 = subplot(2,1,1,'Parent',p);
plot(sub1,time_curve,a_vel_z_curve,'color','b','DisplayName','AngVel_Z_curve');
title('Angular Velocity of the wrist');
sub2 = subplot(2,1,2,'Parent',p);
plot(sub2,time_curve,l_vel_x_curve,'color','r','DisplayName','LinVel_X_curve');
title('Linear Velocity of the wrist');

% Here we are using a Median Filter to cut off undesidered values from the
% plots
figure
a_vel_z_filtered = medfilt1(a_vel_z_curve,20);
plot(time_curve,a_vel_z_filtered,'color','b','DisplayName','AngVel_Z_filtered');
figure
l_vel_x_filtered = medfilt1(l_vel_x_curve,20);
plot(time_curve,l_vel_x_filtered,'color','b','DisplayName','LinVel_X_filtered');
```

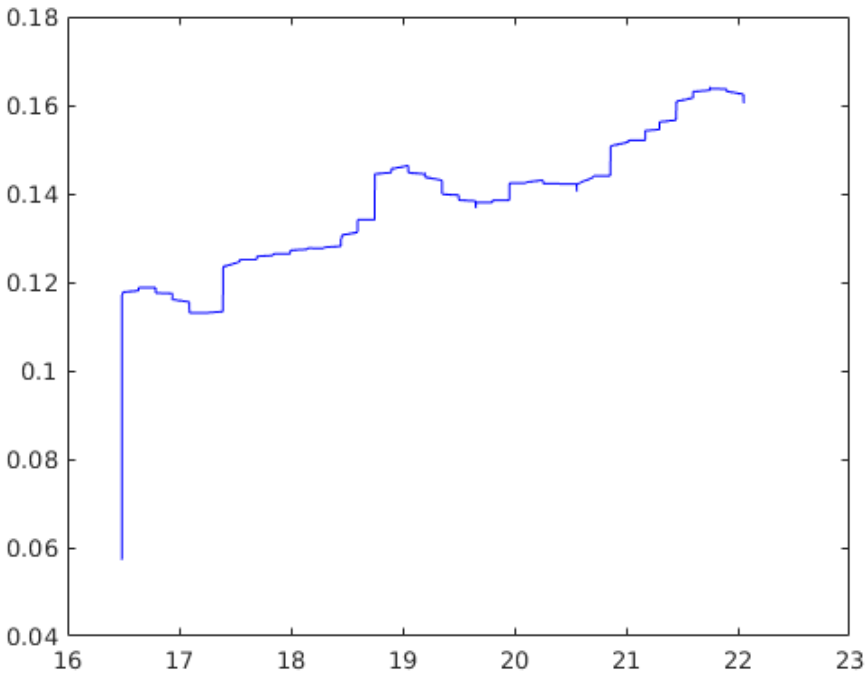

Straight Path Commands
Angular Velocity of the wrist
0.5
0
-0.5
22
23
24
25
26
27
28
Linear Velocity of the wrist
0.25
0.2
0.15
0.1
22
23
24
25
26
27
28


0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
22
23
24
25
26
27
28


0.22
0.2
0.18
0.16
0.14
0.12
0.1
0.08
22
23
24
25
26
27
28


Curved Path Commands
Angular Velocity of the wrist
0.2
0
-0.2
-0.4
-0.6
16
17
18
19
20
21
22
23
Linear Velocity of the wrist
0.18
0.16
0.14
0.12
0.1
16
17
18
19
20
21
22
23





Published with MATLAB® R2017a