Contents


```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Code for local predictions of gaps and overlaps
% Author : Siddharth Pantoji @ TU Delft
% Description :
% This script uses fused data from all 4 sensors and do local level/tape
% level predictions to be used for online corrections
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

load neccesary data

```
clc
clear all
close all

% Get to storage folder for 4 sensors data
cd 'O:\Siddharth Experiment 31 straight lines 26th July\Manipulated timetables\Struct with all data'

% load all mats.
mat = dir('*.mat');
for q = 1:length(mat)
    load(mat(q).name);
end
clear q mat Data Fit_T_Logistic Fit_C_Extremevalue Fit_B_Normal Fit_A_Normal
```

Timestamps visualisation for a two adjacent run

```
% dont need everytime
i = 16;
figure(i);
hold on
plot(DataTimeTables(i).Tracker.TimeStamp,DataTimeTables(i).Tracker.Y_normalised);
plot(DataTimeTables(i).Tracker.TimeStamp,(DataTimeTables(i).Tracker.X_mm_/200));% mag reduced to visualise
plot(DataTimeTables(i).Tracker.TimeStamp,(DataTimeTables(i).Tracker.Velocity/31.5));% mag reduced to visualise
plot(DataTimeTables(i).LLS_A.Time,DataTimeTables(i).LLS_A.TapeWidth_Arc1);
plot(DataTimeTables(i).Camera.TimeStamps,DataTimeTables(i).Camera.TapeCenterLine);
plot(DataTimeTables(i).LLS_B.Time,DataTimeTables(i).LLS_B.TapeWidthAfterCompaction_Rotated);

x(1) = RunCrucialTimeStamps.RunStartTime(i);
x(2) = RunCrucialTimeStamps.PlatueauStartTime(i);
x(3) = RunCrucialTimeStamps.PlatueauEndTime(i);
x(4) = RunCrucialTimeStamps.CutOffStartTime(i);
x(5) = RunCrucialTimeStamps.CutOffEndTime(i);
x(6) = RunCrucialTimeStamps.Run1mEndTime(i);
x(7) = RunCrucialTimeStamps.RunEndTime(i);
plot(x,[0 0 0 0 0 0 0],'-o');
```

```
legend('Tracker','X mm','Velocity','LLS A','Camera','LLS B','Crucial Timestamps')

clear x

i = 17;
figure(i);
hold on
plot(DataTimeTables(i).Tracker.TimeStamp,DataTimeTables(i).Tracker.Y_normalised);
plot(DataTimeTables(i).Tracker.TimeStamp,(DataTimeTables(i).Tracker.X_mm_/200));% mag reduced to visualise
plot(DataTimeTables(i).Tracker.TimeStamp,(DataTimeTables(i).Tracker.Velocity/31.5));% mag reduced to visualise
plot(DataTimeTables(i).LLS_A.Time,DataTimeTables(i).LLS_A.TapeWidth_Arc1);
plot(DataTimeTables(i).Camera.TimeStamps,DataTimeTables(i).Camera.TapeCenterLine);
plot(DataTimeTables(i).LLS_B.Time,DataTimeTables(i).LLS_B.TapeWidthAfterCompaction_Rotated);

x(1) = RunCrucialTimeStamps.RunStartTime(i);
x(2) = RunCrucialTimeStamps.PlatueauStartTime(i);
x(3) = RunCrucialTimeStamps.PlatueauEndTime(i);
x(4) = RunCrucialTimeStamps.CutOffStartTime(i);
x(5) = RunCrucialTimeStamps.CutOffEndTime(i);
x(6) = RunCrucialTimeStamps.Run1mEndTime(i);
x(7) = RunCrucialTimeStamps.RunEndTime(i);
plot(x,[0 0 0 0 0 0 0],'-o');

legend('Tracker','X mm','Velocity','LLS A','Camera','LLS B','Crucial Timestamps')

clear x
```

figure specs PAPER

```
figure(99);
```

plot gap between n and n+1th tape

```
i = 16 ;% gap between 16 and 17
len = height(MeasuredGapData(i).Gap.Gap_Rotated(1:end));
% X values for constant velocity X = 62, X = 902 mm
% X values for const velocity and excluding LLS A shift X = 372.452, X = 902 mm
% plot will be made for X = 400, X = 900 mm

% made a mistake here.
% tapewidth has been shifted in distance
% so can start comparisions at X = 62 itsself???
% ONLY NOLAN can declutter this!!!

% proportioning for choosing traverse values
PlateauStartIndex_traverse = round(62/1000 * len);
PlateauEndIndex_traverse = round(900/1000 * len);
indexrange = PlateauEndIndex_traverse - PlateauStartIndex_traverse;
Gap400_900 = MeasuredGapData(i).Gap.Gap_Rotated(PlateauStartIndex_traverse:PlateauEndIndex_traverse);
X_for_Gap = linspace(62,900,indexrange+1);
for ii = 1 : length(X_for_Gap)
    Gap_upper_edge(ii) = 6.35/2 + 3.075 + (Gap400_900(ii)/2);
    Gap_lower_edge(ii) = 6.35/2 + 3.075 - (Gap400_900(ii)/2);
end
% plot gap as a band
c = [1 0.8 0.8]
fillBetweenAreaCurve(X_for_Gap, Gap_lower_edge, Gap_upper_edge)
```

Get relevant tape distance alignment indices (DataStreamsJoinedInterpolated) for the 3 streams

```
%%(Indices work in const velocity only)

% Constant velocity between timestamp - plateau start and plateau end
% RunCrucialTimeStamps.PlatueauStartTime(i)
% RunCrucialTimeStamps.PlatueauEndTime(i)

% first estimate from CAD and tracker data and CAD
LLSA_Distance_firstestimate = 417.452 - 107;
```

```
% X range for Tracker 400 900
Xrange_T = [62 900];
TimeRange_T = [DataTimeTables(16).Tracker.Properties.RowTimes(54) DataTimeTables(16).Tracker.Properties.RowTimes(473)];

% X range for Camera
camera_distance_shift = 5;
X_C = Xrange_T - camera_distance_shift;

camera_time_shift = camera_distance_shift/200;
TimeRange_C = TimeRange_T - seconds(camera_time_shift);

% X range for LLS A
LLS_distance_shift = LLSA_Distance_firstestimate;
X_A = Xrange_T - LLS_distance_shift;

LLS_time_shift = LLS_distance_shift/200;
TimeRange_A = TimeRange_T - seconds(LLS_time_shift);
```

Shifting 3 streams in time to get them distance aligned

(valid for const velocity regime only) Sync is changed from truetime to distance-aligned (LLS B ignored)

```
% clear CameraData_Truedistance_16 LLSAData_Truedistance_16

%LLS A
LLSAData = DataTimeTables(16).LLS_A;

    for i = 1:height(LLSAData)
    LLSAData.Time(i) = LLSAData.Time(i) - seconds(LLS_time_shift);
    end

LLSAData_Truedistance_16 = LLSAData;
clear LLSAData;

%Camera
CameraData = DataTimeTables(16).Camera;

    for i = 1:height(CameraData)
    CameraData.TimeStamps(i) = CameraData.TimeStamps(i) - seconds(camera_time_shift);
    end

CameraData_Truedistance_16 = CameraData;

clear CameraData;



%LLS A
LLSAData = DataTimeTables(17).LLS_A;

    for i = 1:height(LLSAData)
    LLSAData.Time(i) = LLSAData.Time(i) - seconds(LLS_time_shift);
    end

LLSAData_Truedistance_17 = LLSAData;
clear LLSAData;

%Camera
CameraData = DataTimeTables(17).Camera;

    for i = 1:height(CameraData)
    CameraData.TimeStamps(i) = CameraData.TimeStamps(i) - seconds(camera_time_shift);
    end

CameraData_Truedistance_17 = CameraData;

clear CameraData;
```

fusing data from time tables which have been shifted to accurately match distance

timestamps

```
for i = 16:16
    clear a b c d
    a = DataTimeTables(i).Tracker;
    b = CameraData_Truedistance_16;
    c = LLSAData_Truedistance_16;
    DataStreamsJoinedInterpolated_distancealigned(i).Run = synchronize(a,b,c,'union','linear');
    clear a b c d i
end


for i = 17:17
    clear a b c d
    a = DataTimeTables(i).Tracker;
    b = CameraData_Truedistance_17;
    c = LLSAData_Truedistance_17;
    DataStreamsJoinedInterpolated_distancealigned(i).Run = synchronize(a,b,c,'union','linear');
    clear a b c d i
end
```

Visulaising shifts in streams to ensure sanity!

```
timestmp = DataStreamsJoinedInterpolated_distancealigned(16).Run.Properties.RowTimes;

figure(1000);
hold on
plot(timestmp,DataStreamsJoinedInterpolated_distancealigned(16).Run.Y_normalised);
plot(timestmp,(DataStreamsJoinedInterpolated_distancealigned(16).Run.X_mm_/200));% mag reduced to visualise
plot(timestmp,(DataStreamsJoinedInterpolated_distancealigned(16).Run.Velocity/31.5));% mag reduced to visualise
plot(timestmp,DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeWidth_Arc1);
plot(timestmp,DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeCenterLine);
```

plot nth tape

position = tracker + camera

```
avgwidthincrease = mean((DataTimeTables(16).LLS_B.TapeWidthAfterCompaction_Rotated))-(mean(DataTimeTables(16).LLS_A.TapeWidth_Arc1));

% for i = 630 : 1224
    for i = 225 : 1221

% position + camera + geometry A + avg of compaction contribution =
Tape16_upper_edge(i) = ...
    DataStreamsJoinedInterpolated_distancealigned(16).Run.Y_normalised(i)...
    + DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeCenterLine(i)...
    + 3.0296...
    + (avgwidthincrease/2);
    % (DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeWidth_Arc1(i)/2) + ...

Tape16_lower_edge(i) = ...
    DataStreamsJoinedInterpolated_distancealigned(16).Run.Y_normalised(i)...
    + DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeCenterLine(i)...
    - 3.0296...
    - (avgwidthincrease/2);
    % (DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeWidth_Arc1(i)/2) - ...

end

X_for_Tape16 = linspace(62,900,1221-225+1);
```

plot n+1th tape

```
avgwidthincrease = mean((DataTimeTables(17).LLS_B.TapeWidthAfterCompaction_Rotated))-(mean(DataTimeTables(17).LLS_A.TapeWidth_Arc1));

% for i = 619 : 1208
for i = 214 : 1208
```

```
% position + camera + geometry A + avg of compaction contribution =
Tape17_upper_edge(i) = ...
    DataStreamsJoinedInterpolated_distancealigned(17).Run.Y_normalised(i)...
    + DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeCenterLine(i)...
    + (DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeWidth_Arc1(i)/2)...% 3.0296...
    + (avgwidthincrease/2)...
    + 12.5;
    % (DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeWidth_Arc1(i)/2) + ...

Tape17_lower_edge(i) = ...
    DataStreamsJoinedInterpolated_distancealigned(17).Run.Y_normalised(i)...
    + DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeCenterLine(i)...
    - (DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeWidth_Arc1(i)/2)...%3.0296...
    - (avgwidthincrease/2)...
    + 12.5;
    % (DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeWidth_Arc1(i)/2) - ...

end

X_for_Tape17 = linspace(62,900,1208-214+1);
```

Get all arrays to be the asme length for convenience

```
X_for_all = linspace(62,900,2000);

Gap400_900_normalised = interp1(X_for_Gap, Gap400_900, X_for_all);

%16 black
Tape16_lower_edge_normalised = interp1(X_for_Tape16, Tape16_lower_edge(225:end),X_for_all);
Tape16_upper_edge_normalised = interp1(X_for_Tape16, Tape16_upper_edge(225:end),X_for_all);

%17 blue
Tape17_lower_edge_normalised = interp1(X_for_Tape17, Tape17_lower_edge(214:end),X_for_all);
Tape17_upper_edge_normalised = interp1(X_for_Tape17, Tape17_upper_edge(214:end),X_for_all);
```

Figures

```
figure

hold on

plot(X_for_all,Tape16_lower_edge_normalised,Color="k");
plot(X_for_all,Tape16_upper_edge_normalised,Color="k");

plot(X_for_all,Tape17_lower_edge_normalised,Color="b");
plot(X_for_all,Tape17_upper_edge_normalised,Color="b");

% adding gap on both sides of the gap centerline
Gapcenterline = (Tape17_lower_edge_normalised + Tape16_upper_edge_normalised)/2;
for i=1:length(X_for_all)
    Gap_upper_edge_normalised(i) = Gapcenterline(i) + (Gap400_900_normalised(i)/2);
    Gap_lower_edge_normalised(i) = Gapcenterline(i) - (Gap400_900_normalised(i)/2);
end

plot(X_for_all,Gapcenterline,'-');
plot(X_for_all,Gap_lower_edge_normalised,Color="r");
plot(X_for_all,Gap_upper_edge_normalised,Color="r");

hold off
figure
hold on
plot(X_for_all,Gap400_900_normalised,'b')
plot(X_for_all,(Tape17_lower_edge_normalised - Tape16_upper_edge_normalised),'o')
```

Understanding data

```
figure
hold on
plot(DataStreamsJoinedInterpolated_distancealigned(16).Run.X_mm_,DataStreamsJoinedInterpolated_distancealigned(16).Run.Y_normalised)
```

```
plot(DataStreamsJoinedInterpolated_distancealigned(16).Run.X_mm_,DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeCenterLine)
plot(DataStreamsJoinedInterpolated_distancealigned(16).Run.X_mm_,DataStreamsJoinedInterpolated_distancealigned(16).Run.TapeWidth_Arc1)

plot(DataStreamsJoinedInterpolated_distancealigned(17).Run.X_mm_,DataStreamsJoinedInterpolated_distancealigned(17).Run.Y_normalised+2)
plot(DataStreamsJoinedInterpolated_distancealigned(17).Run.X_mm_,DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeCenterLine+2)
plot(DataStreamsJoinedInterpolated_distancealigned(17).Run.X_mm_,DataStreamsJoinedInterpolated_distancealigned(17).Run.TapeWidth_Arc1+2)
```

Understand data

```
j=1;

figure; hold on;
for i=1:31
    plot(DataTimeTables(i).Camera.TapeCenterLine + j);
    j=j+1;
end

j=-6.15;

figure; hold on;
for i=1:30
    plot(MeasuredGapData(i).Gap.Gap_Rotated + j);
    j=j+2;
end
```

Appendix / trash

figure fillBetweenAreaCurve(X_for_Tape16, Tape16_lower_edge(630:end), Tape16_upper_edge(630:end)) figure fillBetweenAreaCurve(X_for_Tape17, Tape17_lower_edge(619:end), Tape17_upper_edge(619:end))

Gap400_900

figure hold on %gap plot(X_for_Gap, Gap_upper_edge,Color="r") plot(X_for_Gap, Gap_lower_edge,Color="r")

%16 black plot(X_for_Tape16, Tape16_lower_edge(630:end),Color="k") plot(X_for_Tape16, Tape16_upper_edge(630:end),Color="k")

%17 blue plot(X_for_Tape17, Tape17_lower_edge(619:end),Color="b") plot(X_for_Tape17, Tape17_upper_edge(619:end),Color="b")

hold off

vq16 = interp1(X_for_Tape17, Tape17_lower_edge(619:end),X_for_Gap) vq17 = interp1(X_for_Tape16, Tape16_upper_edge(630:end),X_for_Gap)

% figure(9999) % %16 black % plot(X_for_Gap, vq16,Color="k") % plot(X_for_Tape16, Tape16_upper_edge(630:end),Color="k") % % %17 blue % plot(X_for_Tape17, Tape17_lower_edge(619:end),Color="b") % plot(X_for_Gap, vq17,Color="b")

figure hold on plot(X_for_Gap,Gap400_900) plot(X_for_Gap,vq16-vq17) hold off

---

Published with MATLAB® R2024a