

```
clear;
```

Read all log files

```
forwardFiles = dir('./logs/forward*.log');
% Forward motion
for iFile = 1:length(forwardFiles)
    read = importdata(sprintf('./logs/%s', forwardFiles(iFile).name), ...
        ' ', 0);
    forward(iFile).time = read(:,1);
    forward(iFile).x = read(:,2);
    forward(iFile).y = read(:,3);
    forward(iFile).orientation = read(:,4);
    forward(iFile).duration = forward(iFile).time(end) ...
        - forward(iFile).time(1);
end

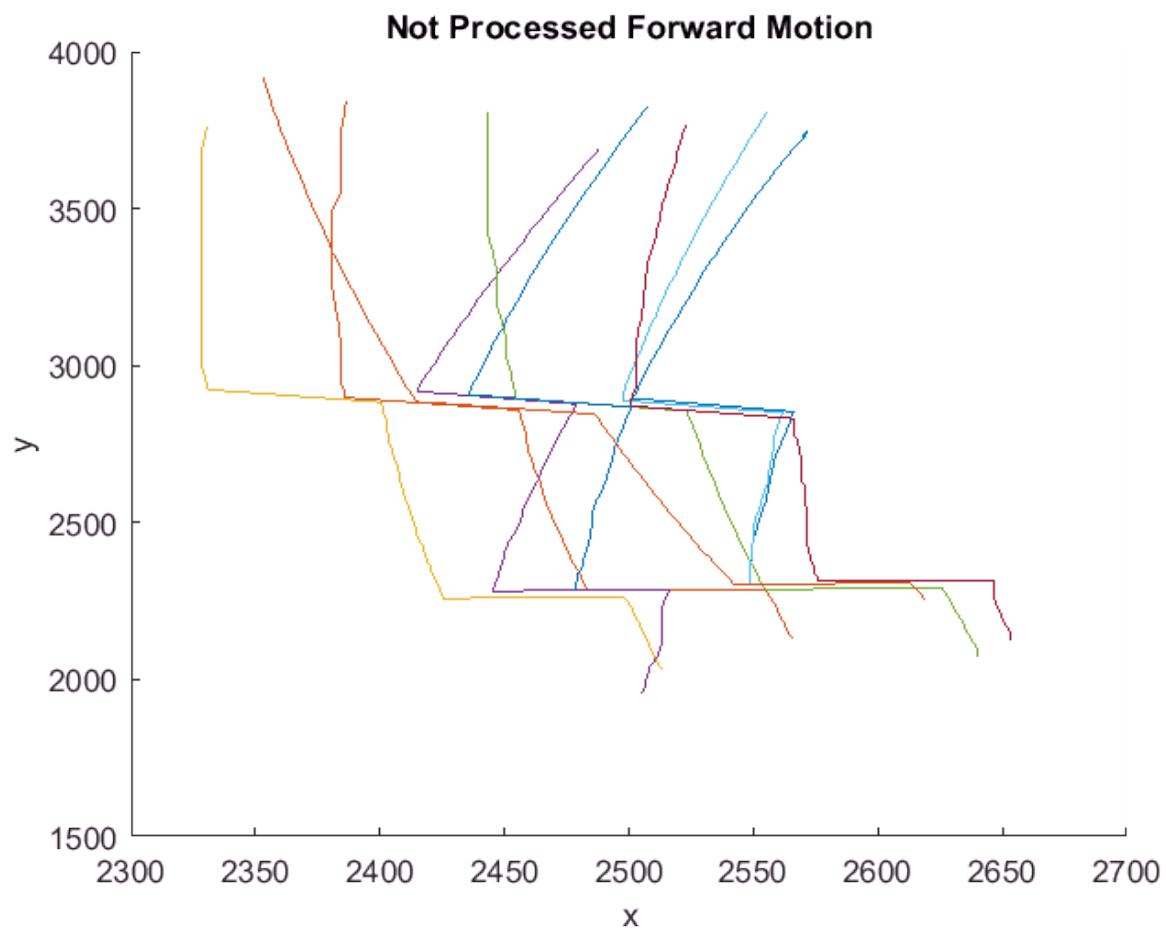
% Right turn
read = importdata('./logs/right_turn.log', ' ', 0);
right.time = read(:,1);
right.x = read(:,2);
right.y = read(:,3);
right.orientation = read(:,4);
right.duration = right.time(end) - right.time(1);

% left turn
read = importdata('./logs/left_turn.log', ' ', 0);
left.time = read(:,1);
left.x = read(:,2);
left.y = read(:,3);
left.orientation = read(:,4);
left.duration = left.time(end) - left.time(1);
```

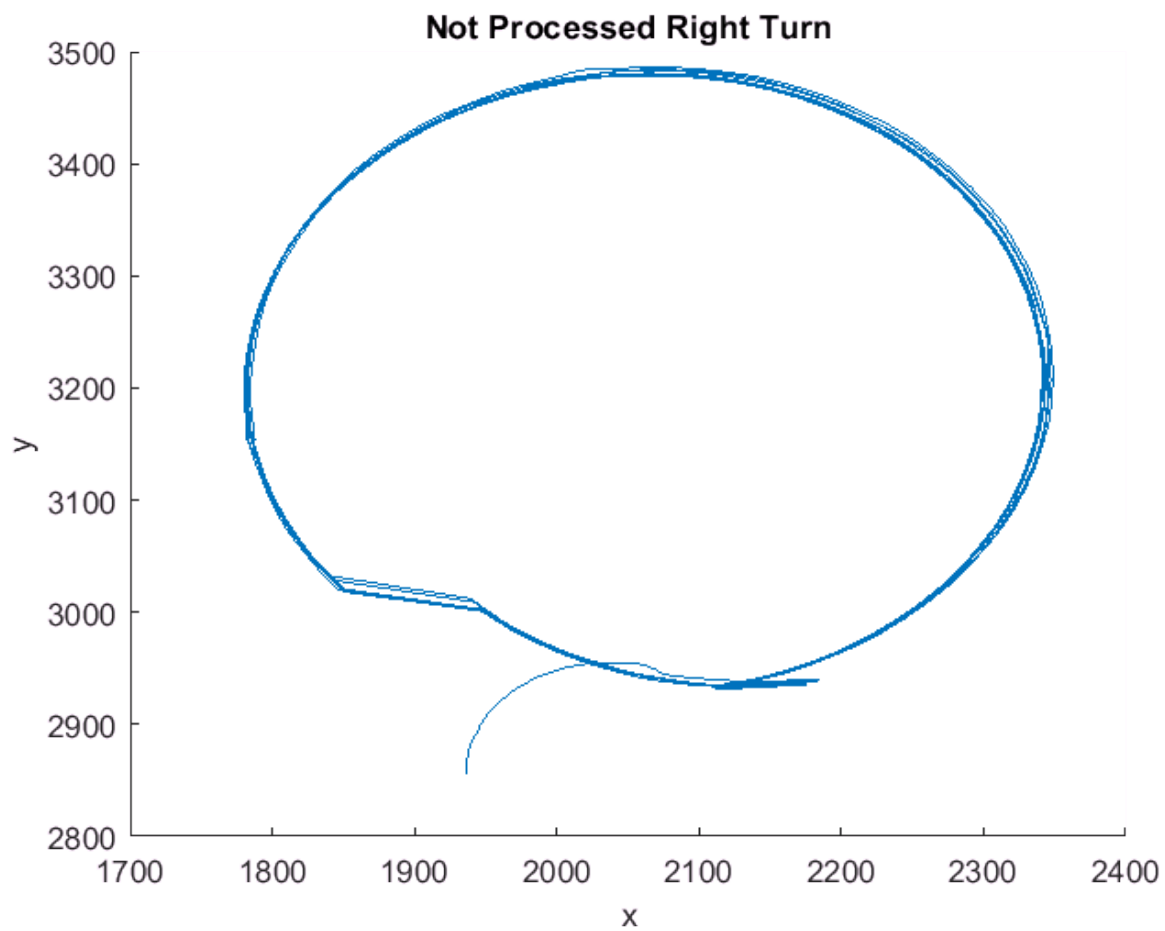
Motion Vizualizations

Plot forward motion

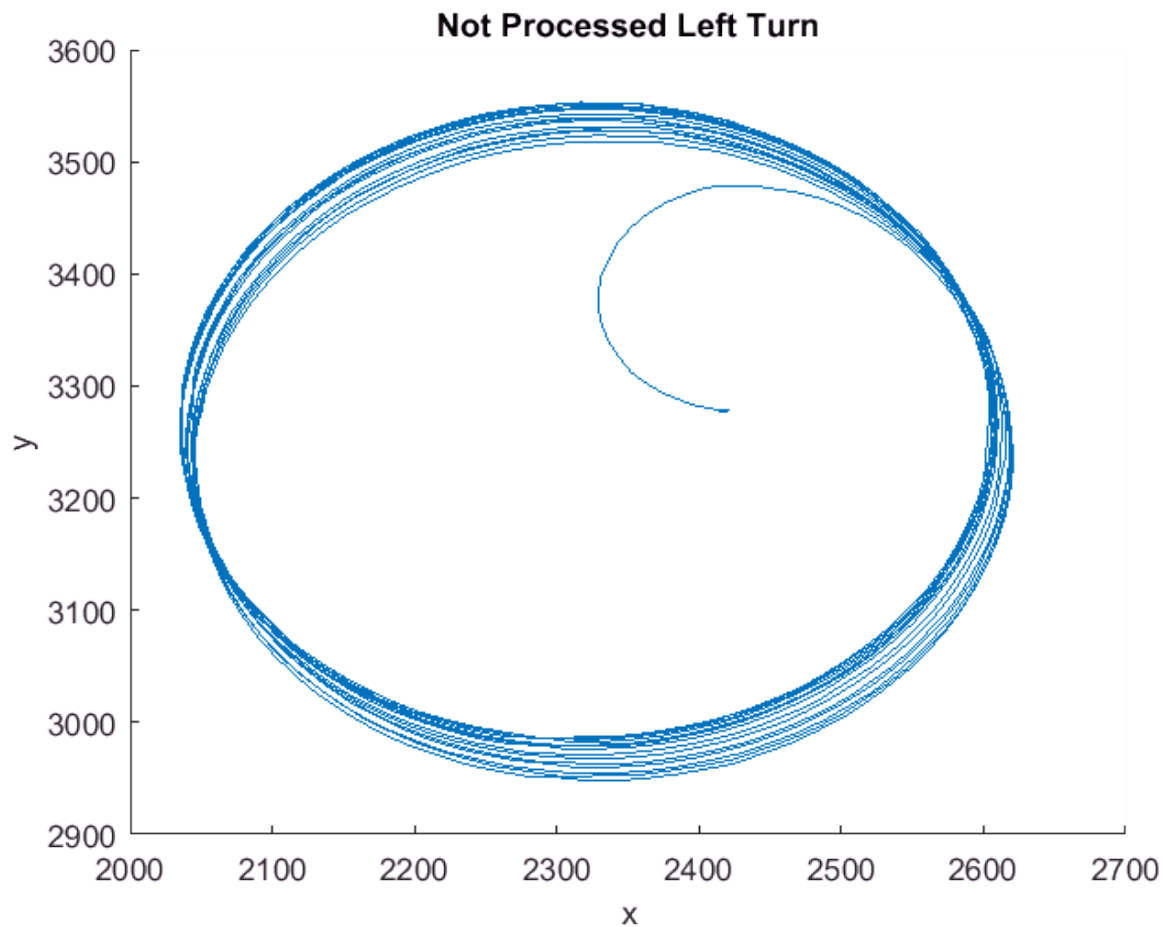
```
figure(1);clf;hold on;
for i=1:length(forward)
    plot(forward(i).x, forward(i).y);
end
title('Not Processed Forward Motion');
xlabel('x');
ylabel('y');
```



```
% Plot right turn  
figure(2);clf;hold on;  
plot(right.x, right.y);  
title('Not Processed Right Turn');  
xlabel('x');  
ylabel('y');
```



```
% Plot left turn
figure(3); clf; hold on;
plot(left.x, left.y);
title('Not Processed Left Turn');
xlabel('x');
ylabel('y');
```



Split left turns into single experiments

```
sequenceDuration = 5000;%round(left.duration/20);
index = 1;
iExperiment = 1;
for sample = 1:length(left.time)
    if left.time(sample) >= left.time(1) + iExperiment * sequenceDuration
        index = 1;
        iExperiment = iExperiment + 1;
    end
    leftExperiment(iExperiment).time(index) = left.time(sample);
    leftExperiment(iExperiment).x(index) = left.x(sample);
    leftExperiment(iExperiment).y(index) = left.y(sample);
    leftExperiment(iExperiment).orientation(index) = ...
        left.orientation(sample);
    index = index + 1;
end
```

Make every experiment start from the same start pose

```
for i = 1:length(leftExperiment)
    % Get orientation
    offsetX = leftExperiment(i).x(1);
    offsetY = leftExperiment(i).y(1);
    offsetOrientation = leftExperiment(i).orientation(1);
```

```

for j = 1:length(leftExperiment(i).time)
    % translate
    x = leftExperiment(i).x(j) - offsetX;
    y = leftExperiment(i).y(j) - offsetY;
    orientation = leftExperiment(i).orientation(j) - offsetOrientation;

    % rotate
    leftExperiment(i).x(j) = cos(-offsetOrientation+pi)*x...
        - sin(-offsetOrientation+pi)*y;
    leftExperiment(i).y(j) = sin(-offsetOrientation+pi)*x...
        + cos(-offsetOrientation+pi)*y;
    leftExperiment(i).orientation(j) = orientation;
end
end

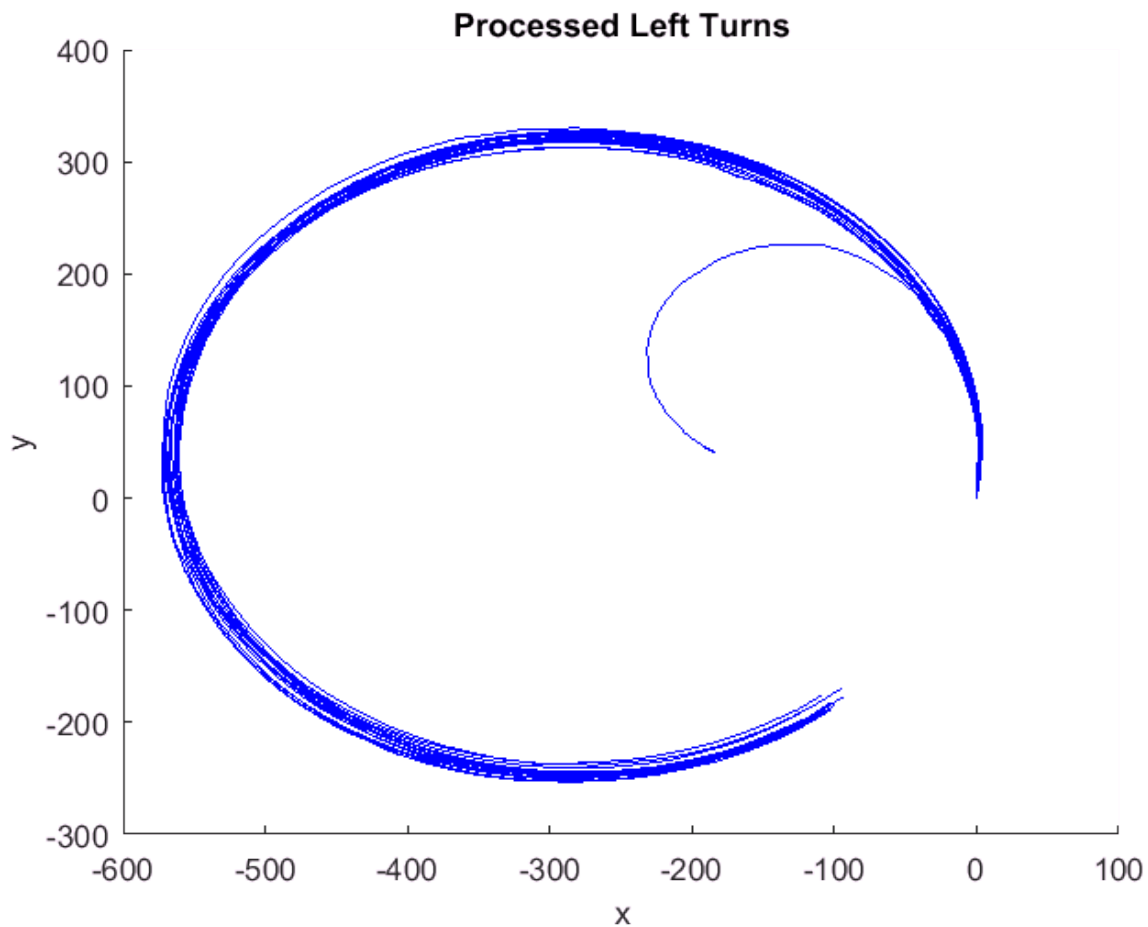
```

Plot Left turns

```

figure(4);clf;hold on;
for experiment = 1:length(leftExperiment)
    plot(leftExperiment(experiment).x, leftExperiment(experiment).y, 'b-');
end
title('Processed Left Turns');
xlabel('x');
ylabel('y');

```



Split right turns into single experiments

```
sequenceDuration = 5000;%round(right.duration/20);
index = 1;
iExperiment = 1;
for sample = 1:length(right.time)
    if right.time(sample) >= right.time(1) + iExperiment * sequenceDuration
        index = 1;
        iExperiment = iExperiment + 1;
    end
    rightExperiment(iExperiment).time(index) = right.time(sample);
    rightExperiment(iExperiment).x(index) = right.x(sample);
    rightExperiment(iExperiment).y(index) = right.y(sample);
    rightExperiment(iExperiment).orientation(index) = ...
        right.orientation(sample);
    index = index + 1;
end
```

Make every experiment start from the same start pose

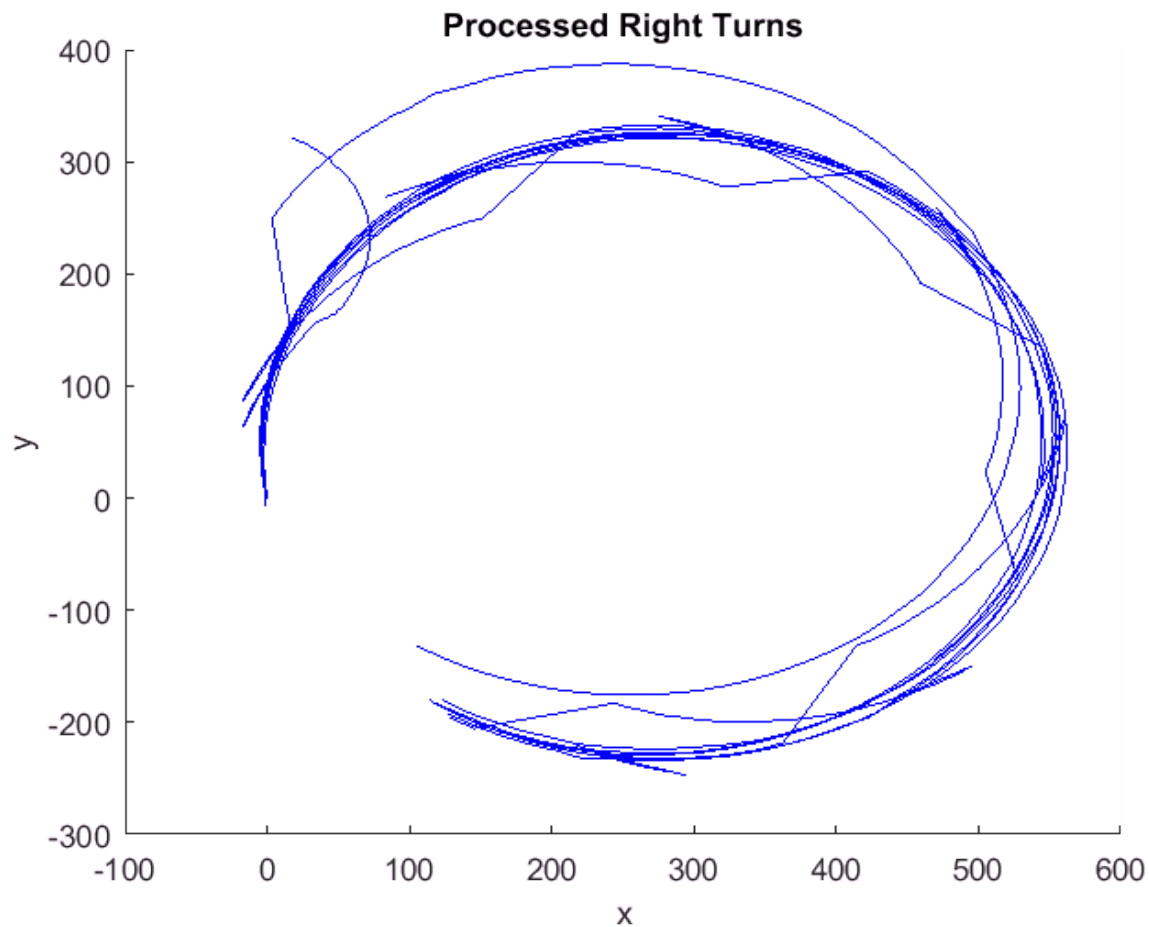
```
for i = 1:length(rightExperiment)
    % Get orientation
    offsetX = rightExperiment(i).x(1);
    offsetY = rightExperiment(i).y(1);
    offsetOrientation = rightExperiment(i).orientation(1);

    for j = 1:length(rightExperiment(i).time)
        % translate
        x = rightExperiment(i).x(j) - offsetX;
        y = rightExperiment(i).y(j) - offsetY;
        orientation = ...
            rightExperiment(i).orientation(j) - offsetOrientation;

        % rotate
        rightExperiment(i).x(j) = cos(-offsetOrientation+pi)*x...
            - sin(-offsetOrientation+pi)*y;
        rightExperiment(i).y(j) = sin(-offsetOrientation+pi)*x...
            + cos(-offsetOrientation+pi)*y;
        leftExperiment(i).orientation(j) = orientation;
    end
end
```

Plot Right turns

```
figure(5);clf;hold on;
for experiment = 1:length(rightExperiment)
    plot(rightExperiment(experiment).x, rightExperiment(experiment).y, 'b-');
end
title('Processed Right Turns');
xlabel('x');
ylabel('y');
```



Plot Forward Motion

Split forwards into single experiments

```

for i = 1:length(forward)
    sequenceDuration = 5000;%round(forward(i).duration/20);
    index = 1;
    iExperiment = 1;
    for sample = 1:length(forward(i).time)
        if forward(i).time(sample) >=...
            forward(i).time(1) + iExperiment * sequenceDuration
            index = 1;
            iExperiment = iExperiment + 1;
        end
        forwardExperiment(i).sequence(iExperiment).time(index) =...
            forward(i).time(sample);
        forwardExperiment(i).sequence(iExperiment).x(index) =...
            forward(i).x(sample);
        forwardExperiment(i).sequence(iExperiment).y(index) =...
            forward(i).y(sample);
        forwardExperiment(i).sequence(iExperiment).orientation(index) =...
            forward(i).orientation(sample);
        index = index + 1;
    end
end

```

```

% Make every experiment start from the same start pose
for j = 1:length(forwardExperiment(i))
    % Get orientation
    offsetX = forwardExperiment(i).sequence(j).x(1);
    offsetY = forwardExperiment(i).sequence(j).y(1);
    offsetOrientation = ...
        forwardExperiment(i).sequence(j).orientation(1);

    for k = 1:length(forwardExperiment(i).sequence(j).time)
        % translate
        x = forwardExperiment(i).sequence(j).x(k) - offsetX;
        y = forwardExperiment(i).sequence(j).y(k) - offsetY;
        orientation = ...
            forwardExperiment(i).sequence(j).orientation(k)...
            - offsetOrientation;

        % rotate
        forwardExperiment(i).sequence(j).x(k) =...
            cos(-offsetOrientation+pi)*x...
            - sin(-offsetOrientation+pi)*y;
        forwardExperiment(i).sequence(j).y(k) =...
            sin(-offsetOrientation+pi)*x...
            + cos(-offsetOrientation+pi)*y;
        forwardExperiment(i).sequence(j).orientation(k) = orientation;
    end
end
end

```

Plot Forward Motion

```

figure(6);clf;hold on;
for i = 1:length(forwardExperiment)
    for j=1:length(forwardExperiment(i).sequence)
        plot(forwardExperiment(i).sequence(j).x,...
            forwardExperiment(i).sequence(j).y, 'b-');
    end
end
axis([-5 5 0 90]);
title('Processed Forward Motions');
xlabel('x');
ylabel('y');

```


Processed Forward Motions

