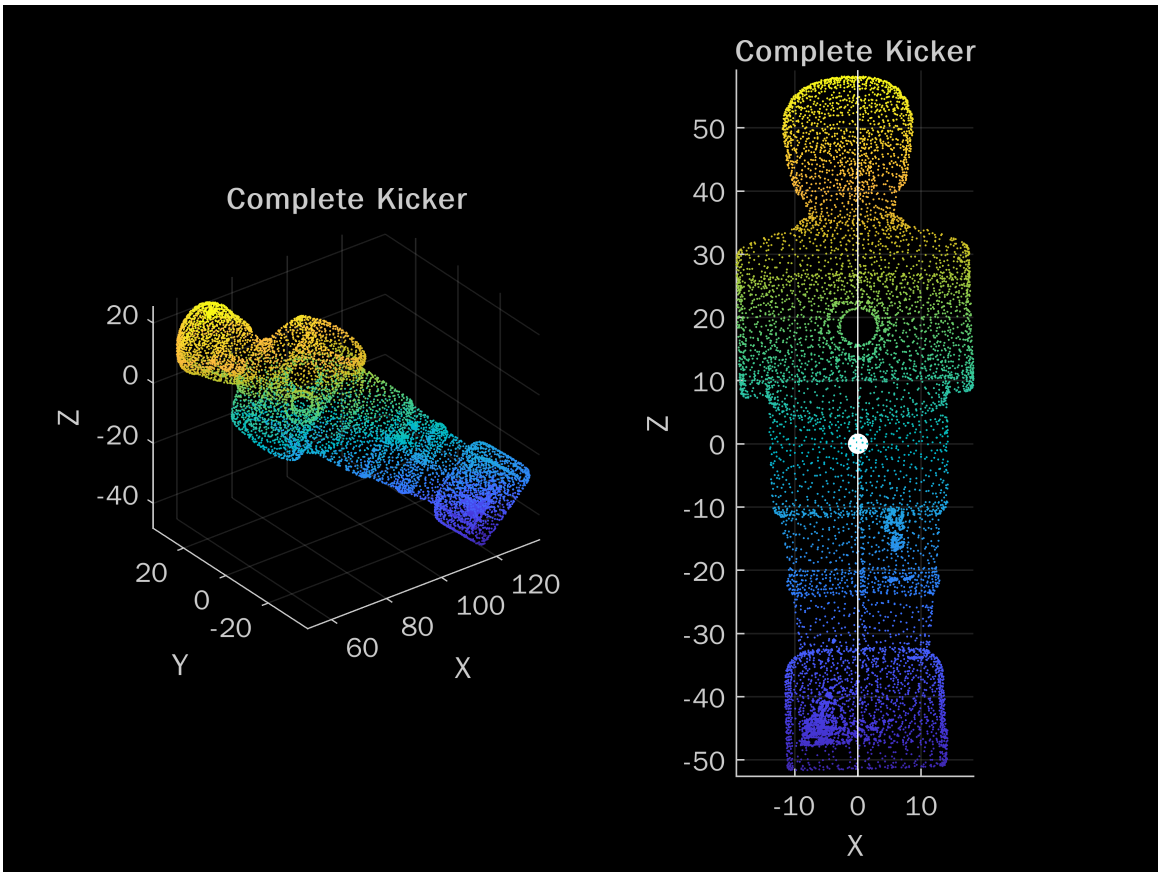
Bilateral Symmetry plane detection of a Complete point cloud

This file covers a method to extract the bilateral symmetry plane of a point cloud. The arbitrarily oriented point cloud is aligned with the largest eigen vector along z-axis using the function 'rotatePointCloudAlongZ' . The Center of Gravity of the point cloud is shifted to (0,0,0) thus enabling a easy detection of the symmetry plane.

```
clear all; clf; clc;

points = readtable("Kicker_complete.xlsx");
points = points{:,1:3};

ax1 = subplot(1,2,1);
pcshow(points,"MarkerSize",7,"VerticalAxis","Z")
% set(h, 'Tag','left')
title("Complete Kicker Unoriented");
xlabel("X")
ylabel("Y")
zlabel("Z")
hold(ax1, "on")


points_aligned = rotatePointCloudAlongZ(points, "x", "",0);
x = points_aligned(:,1);
y = points_aligned(:,2);
z = points_aligned(:,3);

% h = get(gcf, "Children");

ax1 = subplot(1,2,2);
pcshow(points_aligned,"MarkerSize",7,"VerticalAxis","Z")
% set(h, 'Tag','left')
title("Complete Kicker, oriented with symmetry plane");
xlabel("X")
ylabel("Y")
zlabel("Z")
hold(ax1, "on")

limz = [min(z)-1 min(z)-1
        max(z)+1 max(z)+1];

limy = [min(y)-1 max(y)+1
        min(y)-1 max(y)+1];
limx = [0 0
        0 0];
% h = findobj('Tag','left');
% set(h,'NextPlot','add')
plot3(0, 0, 0,'o','color','w',"MarkerSize",7,'MarkerFaceColor',"white")
% h = findobj('Tag','left');
% set(h,'NextPlot','add')
```

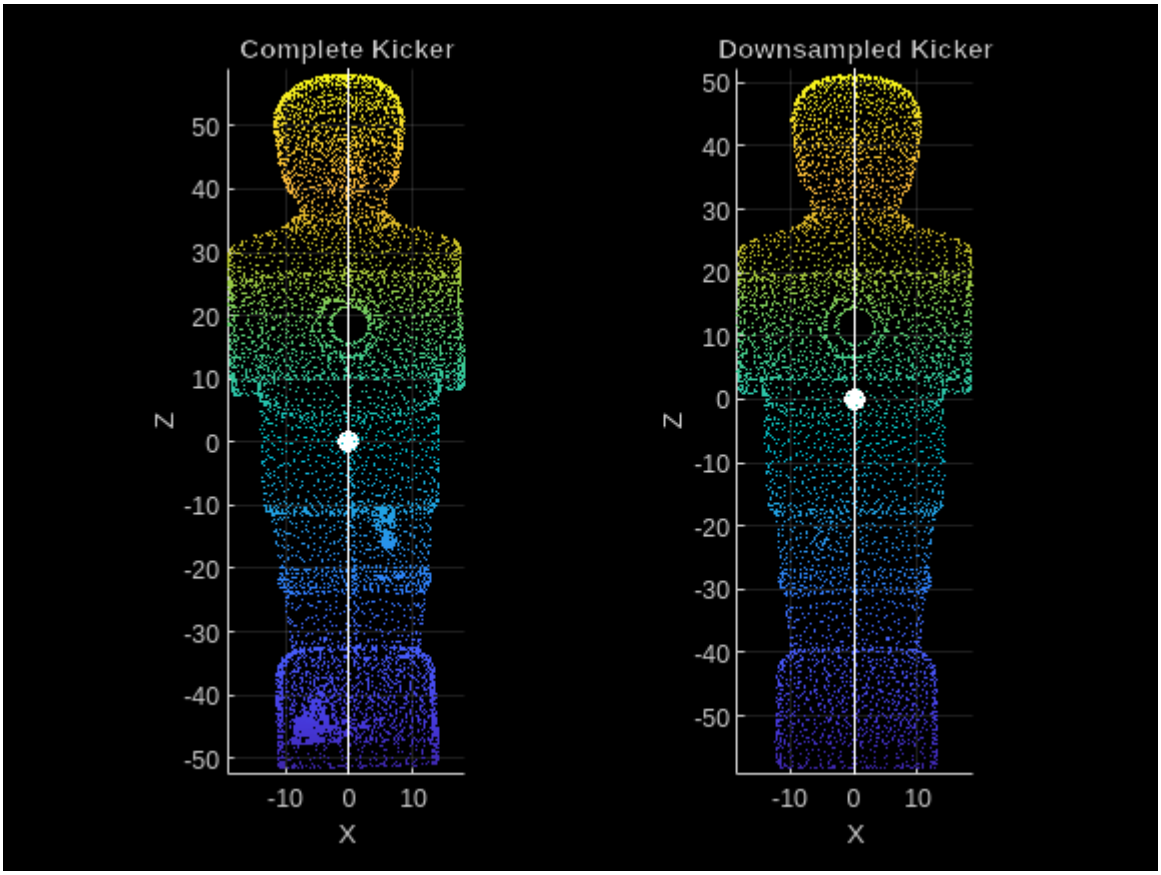
```
surf(limx, limy, limz, "LineStyle","-", "FaceAlpha",0.55,"EdgeColor",[1,1,1] )

subplot(1,2,2)
view([0 0])
```



It can be seen from the figure that the detected symmetry plane has a rotational offset along the Y axis. This is due to the point clusters produced during the laser scanning.

```
points = readtable("Kicker_complete.xlsx");
points = points{:,1:3};

points = pointCloud(points);

gridStep = 1;
points_downsampled = pcdownsample(points, "gridAverage", gridStep);
points_downsampled = points_downsampled.Location;

points_downsampled = rotatePointCloudAlongZ(points_downsampled, "x","",0);
x = points_downsampled(:,1);
y = points_downsampled(:,2);
z = points_downsampled(:,3);


% h = subplot(1,2,2);
```

```
% set(h, 'Tag','right')
ax2 = subplot(1,2,2);
pcshow(points_downsampled,"MarkerSize",7,"VerticalAxis","Z")
title("Downsampled Kicker");
xlabel("X")
ylabel("Y")
zlabel("Z")
hold(ax2, "on")

limz = [min(z)-1 min(z)-1
        max(z)+1 max(z)+1];
limy = [min(y)-1 max(y)+1
        min(y)-1 max(y)+1];
limx = [0 0
        0 0];

plot3(0, 0, 0,'o','color','w',"MarkerSize",7,'MarkerFaceColor',"white")

surf(limx, limy, limz, "LineStyle","-", "FaceAlpha",0.55,"EdgeColor",[1,1,1] )
hold(ax1, "off")
hold(ax2, "off")



subplot(1,2,1)
view([0 0])
subplot(1,2,2)
view([0 0])
```



The points are converted to PointCloud object and are downsampled to remove the clusters of points. The downsampled point cloud is again oriented as shown the figure above, resulting in a accurate bilateral symmetry plane detection.

The result is written to a CSV file and this point cloud is used as a reference to the interpolation of Incomplete point cloud using ICP Registeration.

```
csvwrite("Kicker_complete_Output.xlsx", points_downsampled);
```