EE141L: Lab 4
Point Clouds

Sept. 21, 2018

1 Point clouds

In this lab you will apply matrix products to processing of 3D geometry. One way of representing 3D geometry is with point clouds (PC). More formally, a PC is a matrix

$$\mathbf{P} = [\mathbf{P}_1, \mathbf{P}_2, \cdots, \mathbf{P}_n] \in \mathbb{R}^{3\times n}, \tag{1}$$

where each column

$$\mathbf{P}_i = \begin{bmatrix} p_{i1} \\ p_{i2} \\ p_{i3} \end{bmatrix} \tag{2}$$

has the xyz coordinates of a point in 3D.

Point clouds are the data that's captured by Microsoft kinect[1]. PC are what autonomous vehicles[2] see, PCs are the data you see through the HMD in virtual/augmented reality applications[3].

2 Point cloud processing

2.1 Re-scaling

A scaling matrix is a diagonal matrix

$$\mathbf{S} = \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{bmatrix} \tag{3}$$

Notice that if applied to a point

$$\mathbf{S}\mathbf{P}_i = \begin{bmatrix} s_1 p_{i1} \\ s_2 p_{i2} \\ s_3 p_{i3} \end{bmatrix} \tag{4}$$

[1] https://channel9.msdn.com/coding4fun/kinect/Processing-for-Kinect-for-Windows-v2
[2] https://medium.com/towards-data-science/tracking-pedestrians-for-self-driving-cars-ccf588acd170
[3] https://hololens.reality.news/new/2/

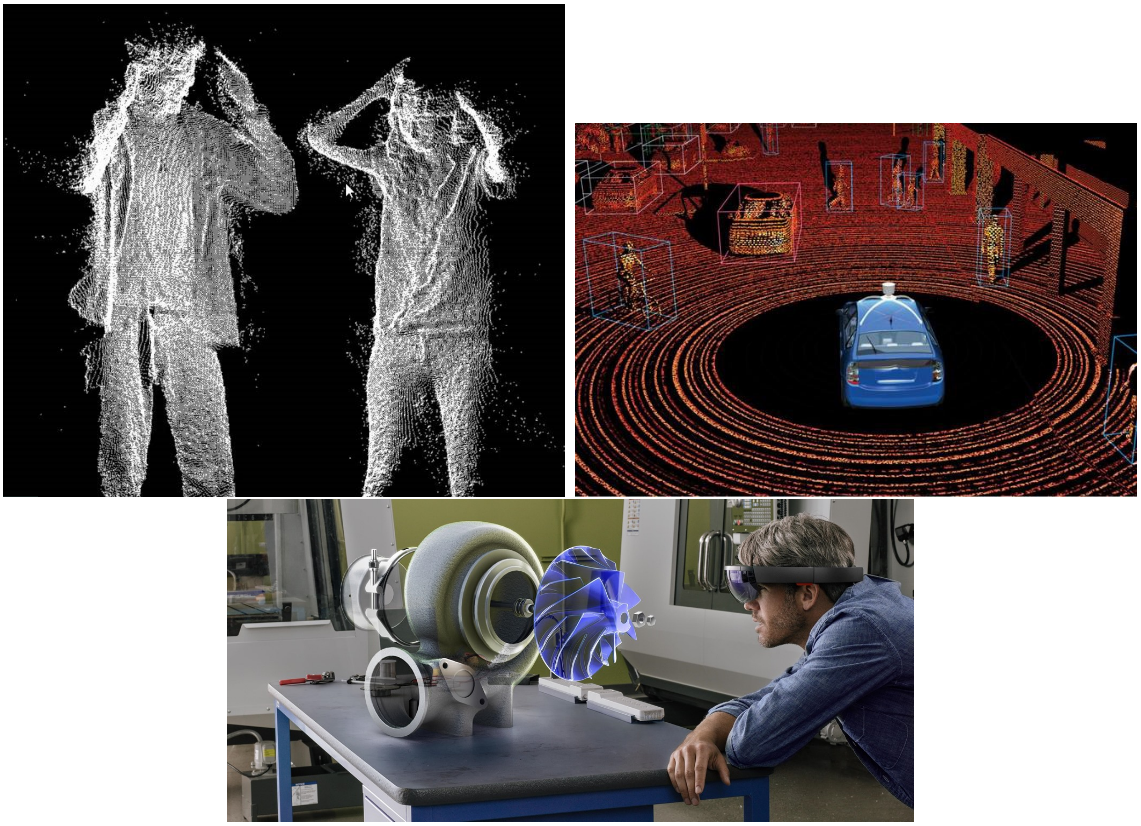

Figure 1: Examples of Point Clouds.

the x,y,z coordinates will be multiplied by s_1, s_2, s_3 respectively. Applying the scaling matrix to the whole point cloud, can be done by multiplying each column by $\mathbf{S}$. This can be achieved by just doing matrix-matrix multiplication

$$\mathbf{SP} = [\mathbf{SP}_1, \mathbf{SP}_2, \cdots, \mathbf{SP}_n] \tag{5}$$

Problem 1. Load the mat file bunny.mat onto Matlab and run the script script.m to display the PC.

1. what happens when $s_1 = s_2 = s_3 = s$, and you vary $s \in [0.01, 5]$
2. Apply a scaling matrix with $s_1 = s_2 = 1$, and s_3 taking values $0, 0.2, 5$ and -1. Explain what the different values of s_3 do to the bunny.

2.2 Rotation

Other simple implemented with matrices are rotations. You are given a function that implements the following operation

$$\mathbf{P}_{rot} = \mathbf{R}_1(\theta)\mathbf{P} \tag{6}$$

for a particular rotation matrix of the form

$$\mathbf{R}_1(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{7}$$

Problem 2. Apply the function rotate_first_axis to the bunny point cloud.

1. Visualize the rotated point cloud, identify the rotation axis.

2. Implement matlab functions rotate_second_axis and rotate_third_axis that implement rotations along the other two axis, write down the corresponding rotation matrices $\mathbf{R}_2(\theta), \mathbf{R}_3(\theta)$ in your report. Apply the functions to the bunny point cloud with different angles (at least 2), include pictures of the resulting point clouds in your report.

3. Intuitively, rotating along an axis by an angle θ, and then again applying rotation along the **same** axis by an angle ϕ should be equivalent to applying one rotation with angle $\theta + \phi$, mathematically that means

$$\mathbf{R}_i(\phi)[\mathbf{R}_i(\theta)\mathbf{P}] = \mathbf{R}_i(\theta + \phi)\mathbf{P} \tag{8}$$

   for $i = 1, 2, 3$, any $\mathbf{P}$. Pick one of the functions you implemented before, and verify this for different values of θ, ϕ.

4. Now we will see that the order in which rotations are performed matters. Verify that the following statement is true

$$\mathbf{R}_1(\phi)[\mathbf{R}_2(\theta)\mathbf{P}] \neq \mathbf{R}_2(\theta)[\mathbf{R}_1(\phi)\mathbf{P}] \tag{9}$$

   by finding an example where the two rotated point clouds are different if you change the ordering.

3 System identification

In practice, linear systems are not given in matrix form, but as an algorithm, or process. The matlab function pointcloud_modification implements a linear system. It takes as input a vector in $\mathbb{R}^3$ and outputs another vector in $\mathbb{R}^3$ by computing the following transformation

$$\mathbf{p}_{out} = \mathbf{A}\mathbf{p}_{in}, \tag{10}$$

where $\mathbf{A}$ is an unknown 3×3 matrix.

Problem 3. In this problem you will find the matrix $\mathbf{A}$

1. Apply the pointcloud_modification function to the bunny point cloud. Describe what happened to the bunny. You can use one of your rotation functions to see better.

2. $\mathbf{e}_i$ is the i-th canonical basis function of $\mathbb{R}^3$, compute the vectors $\mathbf{a}_i = pointcloud\_modification(\mathbf{e}_i)$. Use the them to construct the matrix $\mathbf{A}$.

3. Veryfy that the matrix $\mathbf{A}$ and the function pointcloud_modification have the same effect in points in $\mathbb{R}^3$.