# AER 1515 - Perception for Robotics

Assignment 1

**Submission guidelines**: This assignment is due on Friday, October 23, 2020 at 11:59 pm. Please submit only a single your *typed* solutions as a PDF file.

## Coordinate Transformations and Feature Projections

Reference frames and coordinate transformations are of key importance to roboticists. In the context of robot perception, they are used to relate information between different parts of a robot. On a robot with many different sensors, we must be able to relate the information from each sensor to a central coordinate frame for the data to be usable. In this assignment, we will refer to the central coordinate frame as the robot *body* frame.

## 1 LiDAR Feature Coordinate Transformation (20 pts)

Consider the self-driving car in Figure 1. 3D points detected by the LiDAR are expressed in the LiDAR's local reference frame. To determine where the 3D point is in relation to the car, we must apply a coordinate frame transformation.

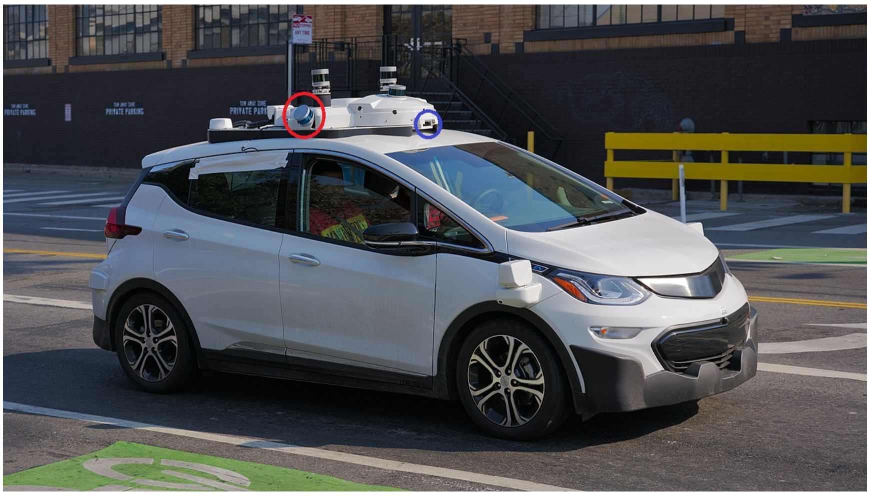

Figure 1: Cruise Automation autonomous vehicle, LiDAR circled in red and camera circled in blue.

The transformation from the LiDAR frame ($L$) to the body frame ($B$) is denoted as $\mathbf{T}_{BL}$. The transformation consists of a rotation matrix $\mathbf{C}_{BL}$ (the rotation of $L$ with respect to $B$) and a translation vector $\mathbf{r}_B^{LB}$ (the vector from $B$ to $L$ expressed in $B$). The full transformation matrix is defined as

$$\mathbf{T}_{BL} = \begin{bmatrix} \mathbf{C}_{BL} & \mathbf{r}_B^{LB} \\ \mathbf{0} & 1 \end{bmatrix}$$

Following a calibration sequence, we have determined the position and rotation of the LiDAR unit. The position of the LiDAR unit follows Figure 2. Distance values are positive if they follow the corresponding body axis and vice versa. The LiDAR is found to be rotated relative to the body frame according to the angles in Table 1. The rotations follow the Tait-Bryan extrinsic rotation format (i.e. 3-2-1 Euler angles). They are applied in the order: $z$, $y$, $x$.

| Axis | Angle (°) |
|---|---|
| $z$ | $-90$ |
| $y$ | $-23$ |
| $x$ | $-10$ |

Table 1: Rotations of the LiDAR with respect to the body frame. Rotations are expressed in Tait-Bryan extrinsic format.

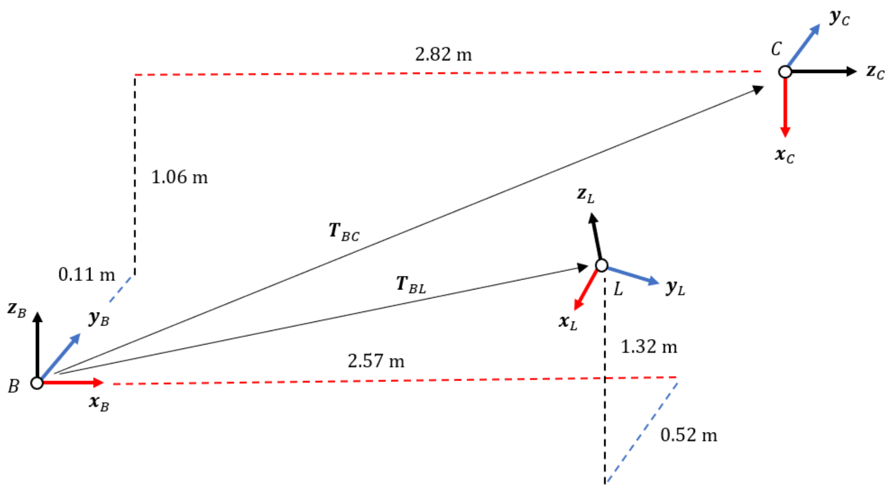


Figure 2: Locations of sensors on an autonomous vehicle.

## Questions

1. Provide the full transformation matrix $\mathbf{T}_{BL}$. Please round values to 3 decimal points. The elementary rotation matrices are provided below. (14 pts)

$$\mathbf{C}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \quad \mathbf{C}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \quad \mathbf{C}_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2. The LiDAR has detected a feature of interest at (3.64, 8.30, 2.45). This point is expressed in the LiDAR's *local* reference frame. Express this point in the body reference frame to 3 decimal places. (3 pts)

3. The transformation matrix from the body frame to the LiDAR frame is denoted as $\mathbf{T}_{LB}$. Given $\mathbf{T}_{LB}$ below and $\mathbf{T}_{BL}$ that you previously found, what do you notice about the relationship between these two matrices? How is this relationship useful in robotics applications?

$$\mathbf{T}_{LB} = \begin{bmatrix} 0 & -0.920 & 0.391 & -0.994 \\ 0.985 & -0.068 & -0.160 & 2.355 \\ 0.174 & 0.385 & 0.907 & -1.443 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# 2 Camera Projections (20 pts)

As we have covered in class, depth cannot be directly recovered when using monocular cameras. To find the depths of detected image features, roboticists often combine the depth information from a LiDAR with the pixel information from a monocular camera. A point in the LiDAR reference frame will be transformed into the camera reference frame by compounding transformation matrices

| Parameter | Value |
|---|---|
| focal length ($x$) | 959.79 |
| focal length ($y$) | 956.93 |
| principle point ($x$) | 696.02 |
| principle point ($y$) | 224.18 |

Table 2: Intrinsic parameters of the monocular camera.

| Parameter | Value |
|---|---|
| $\kappa_1$ | -0.369 |
| $\kappa_2$ | 0.197 |
| $\kappa_3$ | $1.35 \times 10^{-3}$ |
| $\tau_1$ | $5.68 \times 10^{-4}$ |
| $\tau_2$ | -0.068 |

Table 3: Distortion parameters of the monocular camera.

together into a chain (i.e. $\begin{bmatrix} \mathbf{r}_C^{PC} \\ 1 \end{bmatrix} = \mathbf{T}_{CB}\mathbf{T}_{BL}\begin{bmatrix} \mathbf{r}_L^{PL} \\ 1 \end{bmatrix}$). This point can be projected to camera image (pixel) coordinates using a camera projection model.
Assume that the camera can be modelled as an ideal pinhole camera with no skew and plumb bob model distortion. The camera intrinsics are presented in Table 2 and the camera distortion parameters are presented in Table 3. The camera frame is translated according to Figure 2 and is rotated 90° about the pitch axis.

## Questions

1. Provide the full transformation matrix $\mathbf{T}_{BC}$. Please round values to 3 decimal points. (7 pts)
2. Assume that the LiDAR has found another point of interest. After using the transformation from Section 1, the feature is expressed in the body frame as (-3.82895, -1.24462, 5.320). Project this point to the camera's image plane using the pinhole camera model and plumb bob distortion model using the parameters provided in Table 2 and 3. Provide the pixel location of the point, rounded to the nearest full pixel and show all calculations. *Hint*: we suggest not to round intermediate calculations, as any rounding errors will compound (especially with the distortion model) (12 pts)
3. Open the provided image at `data/image/000000.png` (also provided below for reference) in an image editor of your choice (i.e. Inkscape, Gimp, Photoshop, MS Paint) or using Python/OpenCV. What object is at the pixel location above? (1 pt)


Figure 3: `data/image/000000.png` for Question 3 in Section 2

# 3 Putting It All Together (20 pts)

In this task, you will be combining the knowledge from the previous two sections to perform a basic perception task on data from the KITTI dataset. Your task is to process the LiDAR points to extract the depth of each point **relative to the car body frame**. Then, you will determine which points overlap with the camera's field of view and project the points to the image plane. Finally, the points will be plotted over the image with colours corresponding to the depth of each point.

Starter code for Python >=3.5 has been provided to you in a `.zip` file. Additionally, the file contains a folder named `data` which contains a camera image, LiDAR scan, and calibration for the two sensors. Required dependencies to run the starter code is provided in `requirements.txt`. Please include a single image in your write-up with the LiDAR points super-imposed atop the original camera image. Ensure there is a colourmap next to the image which indicates which colour corresponds with which depth value. See Figure 4 for an example.

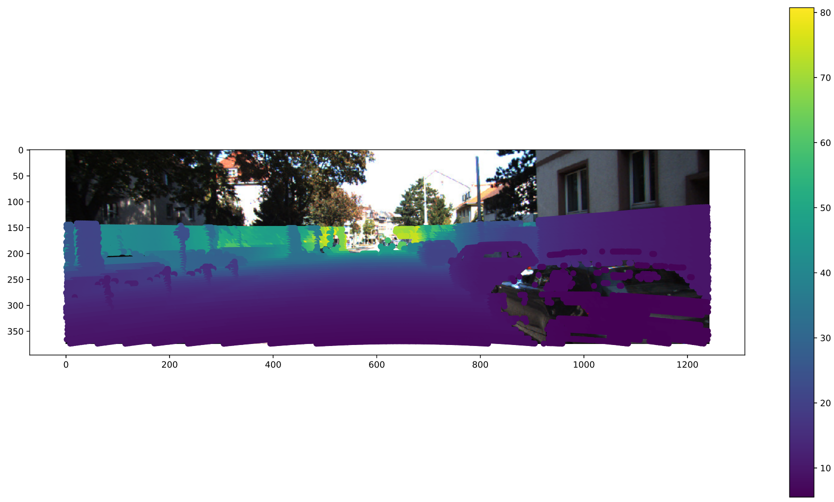

Figure 4: Sample image with projected LiDAR depths overlaid