1

ROB101-OPTIMIZATION

LiDAR-Camera Extrinsic Calibration

Bruce JK Huang and Jessy W. Grizzle

BipedLab @ Robotics Institute

University of Michigan, Ann Arbor

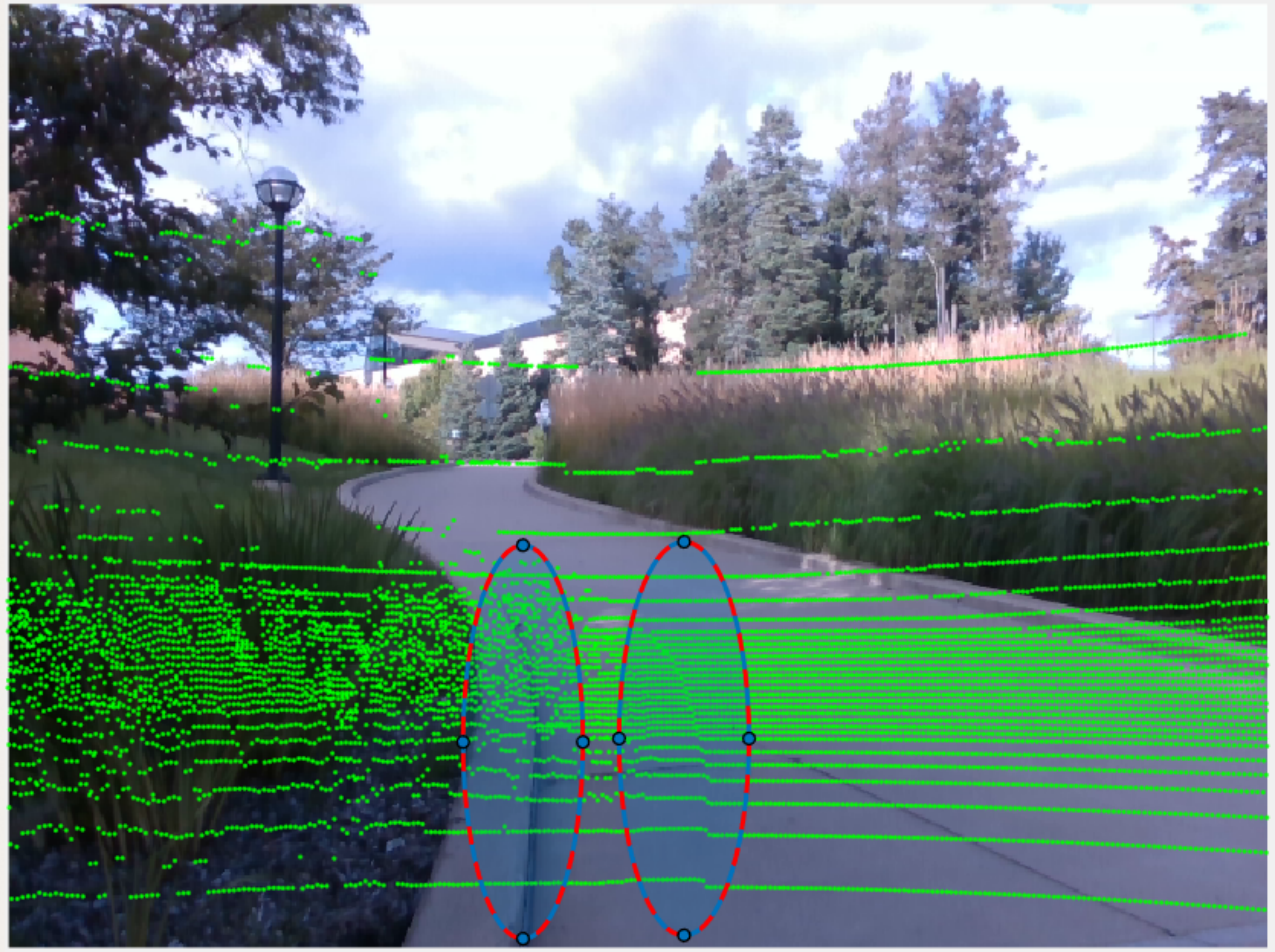
November 16, 2021

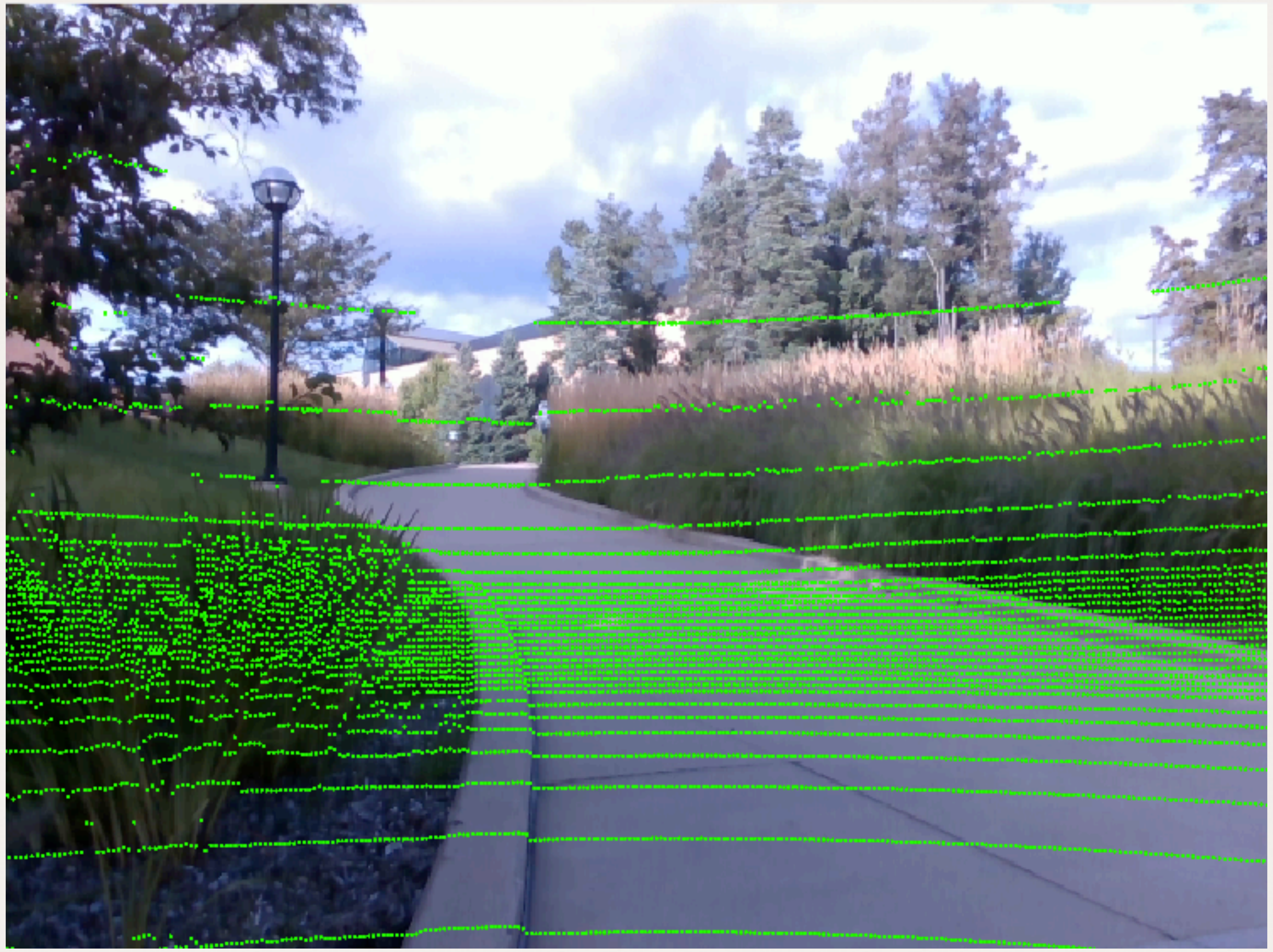
ROBOTICS

Paper: Improvements to Target-Based 3D LiDAR to Camera Calibration

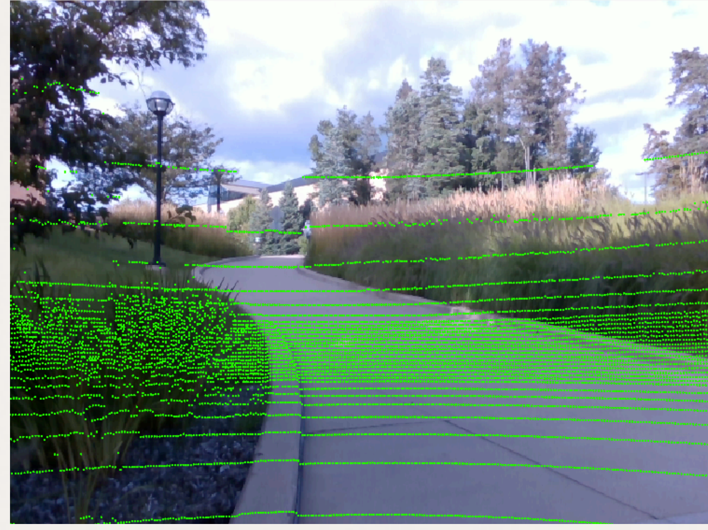
BIPED ROBOTICS LAB@MICHIGAN

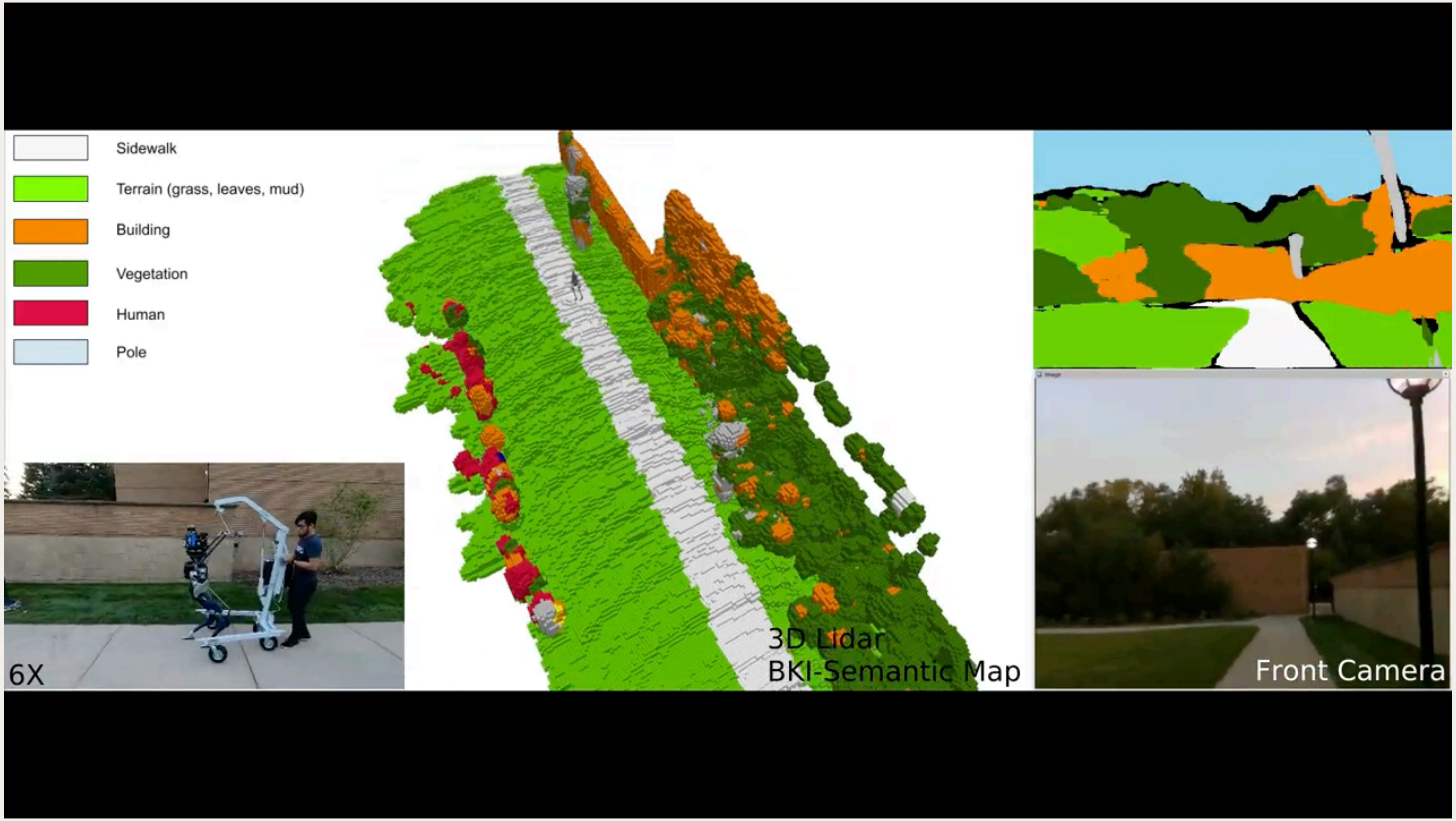
ACCURATE AND PRECISE!!





IDEA: OVERLAYING POINTS TO IMAGE

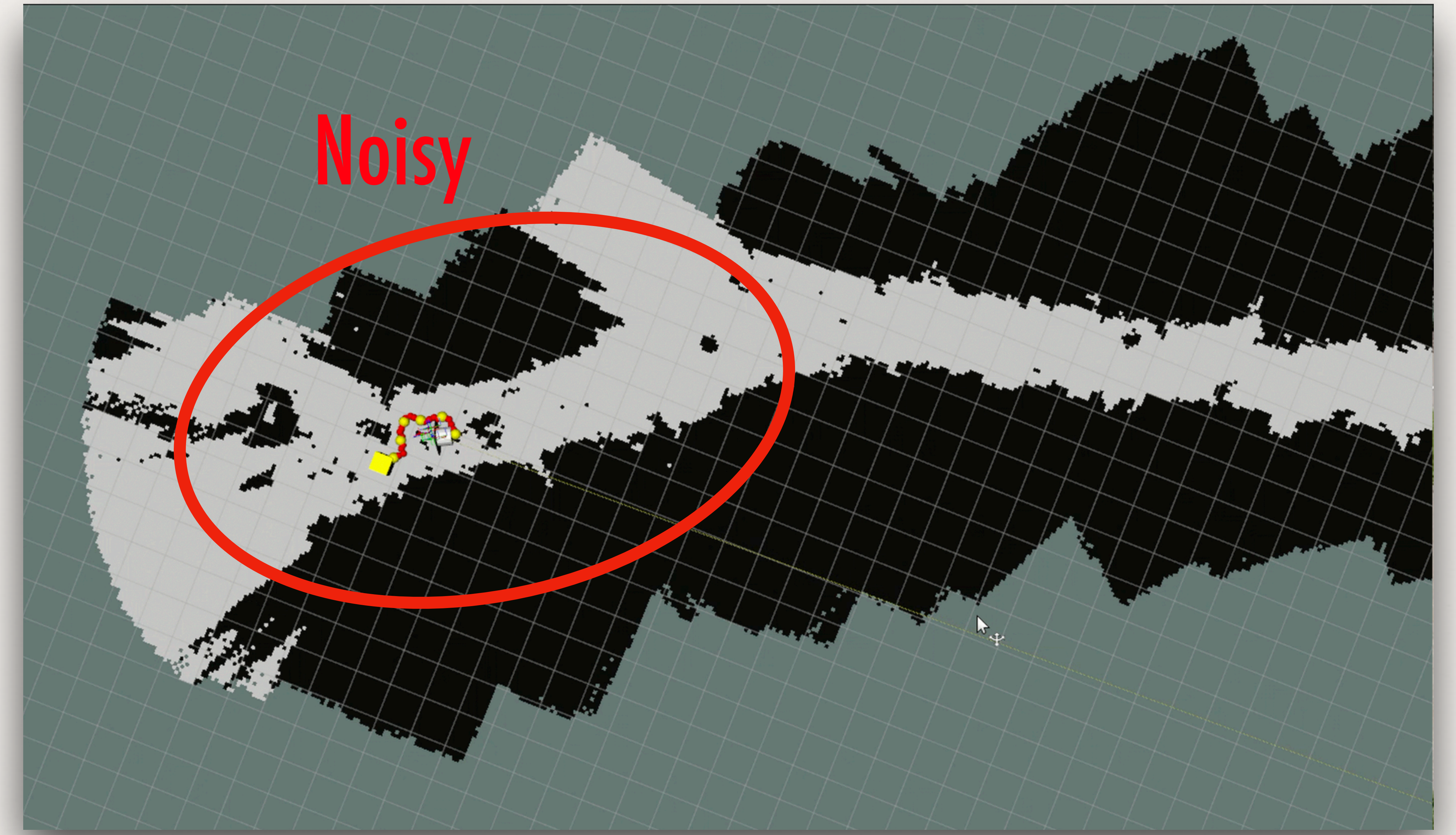




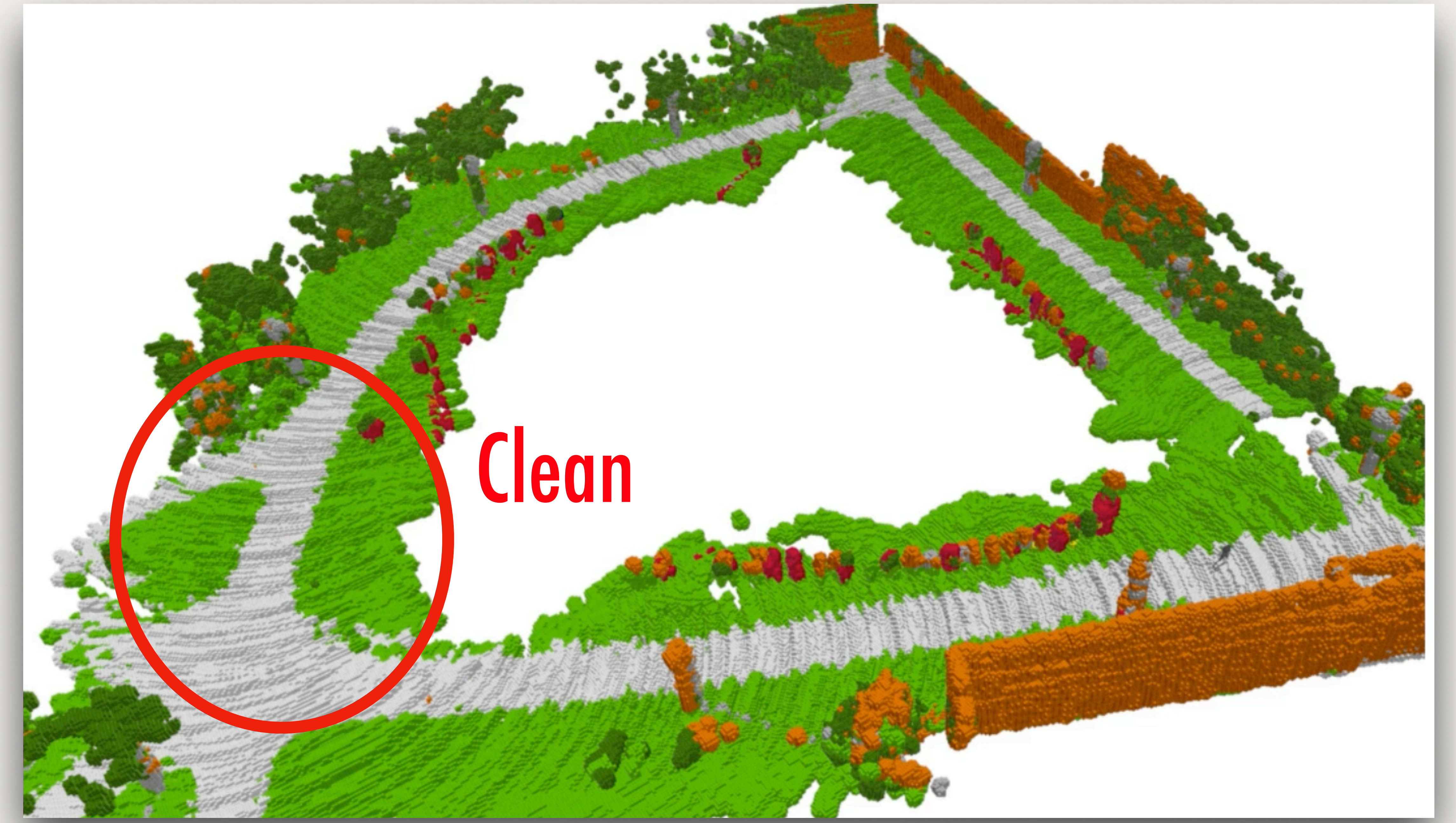
ROBOTICS

BIPED ROBOTICS LAB@MICHIGAN

MOTIVATION AND WHY



Without LiDAR (Only use RGB-D camera)



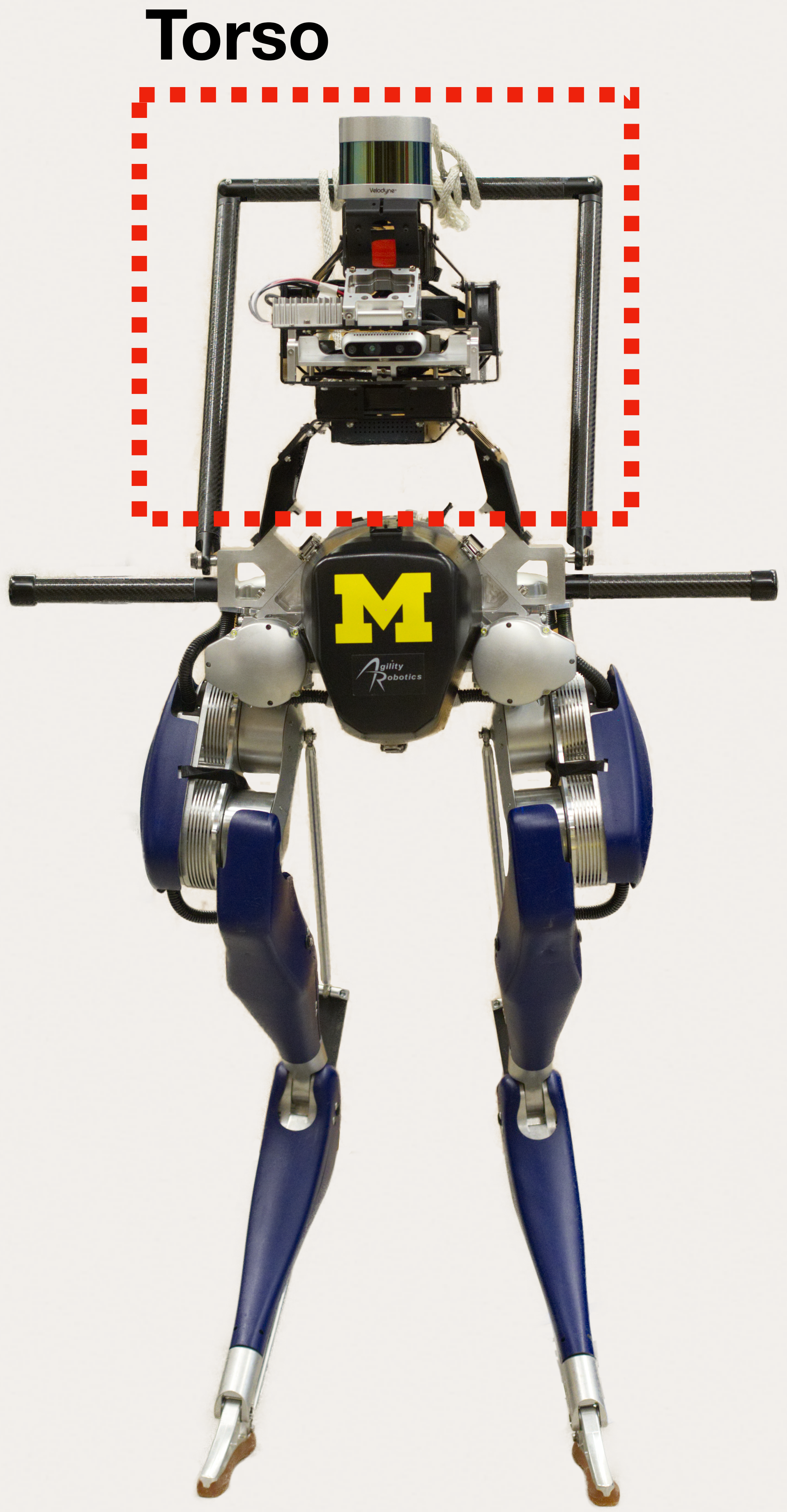
With LiDAR and camera calibrated

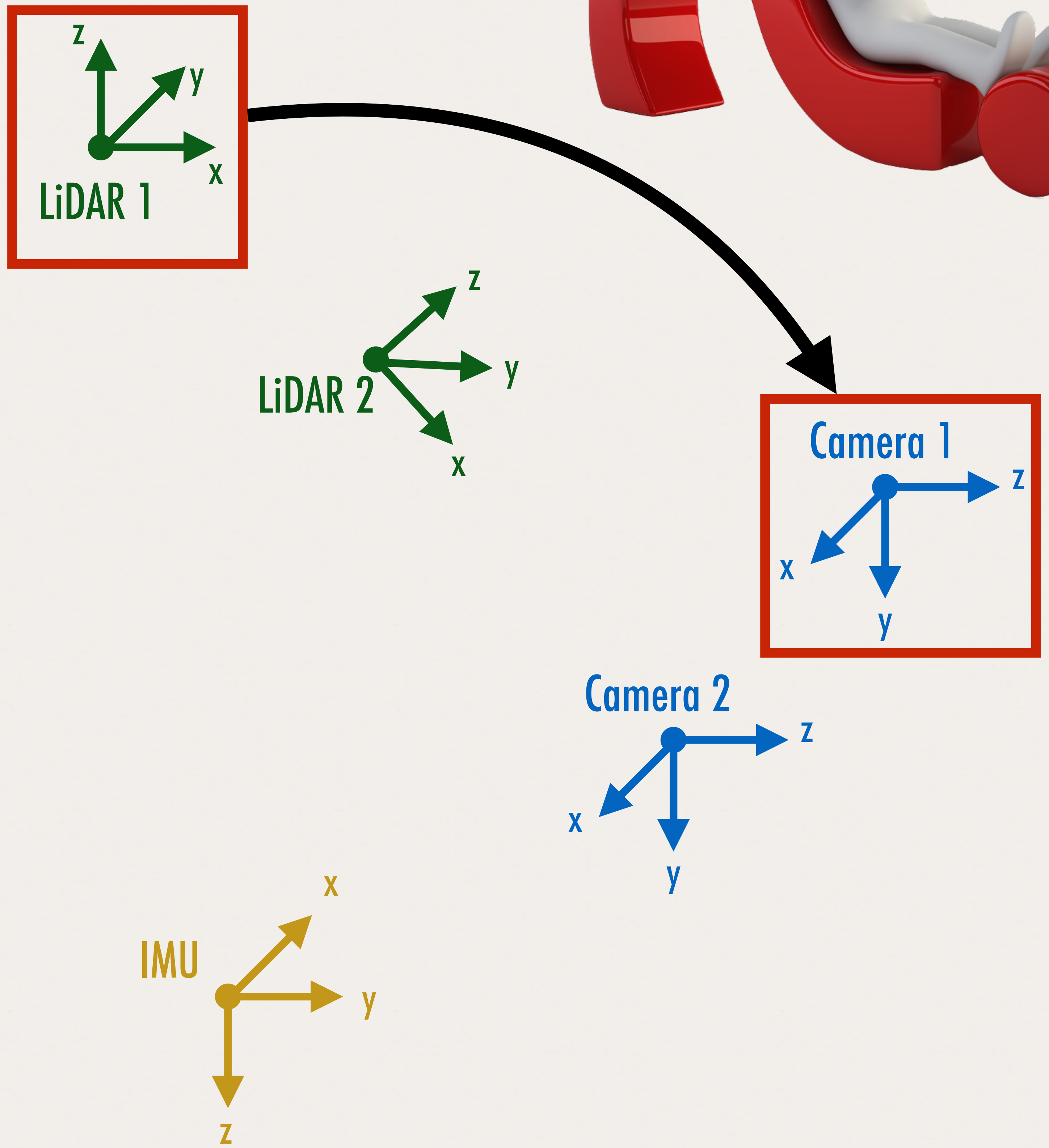
MOTIVATION AND WHY

Without LiDAR (Only use RGB-D camera)

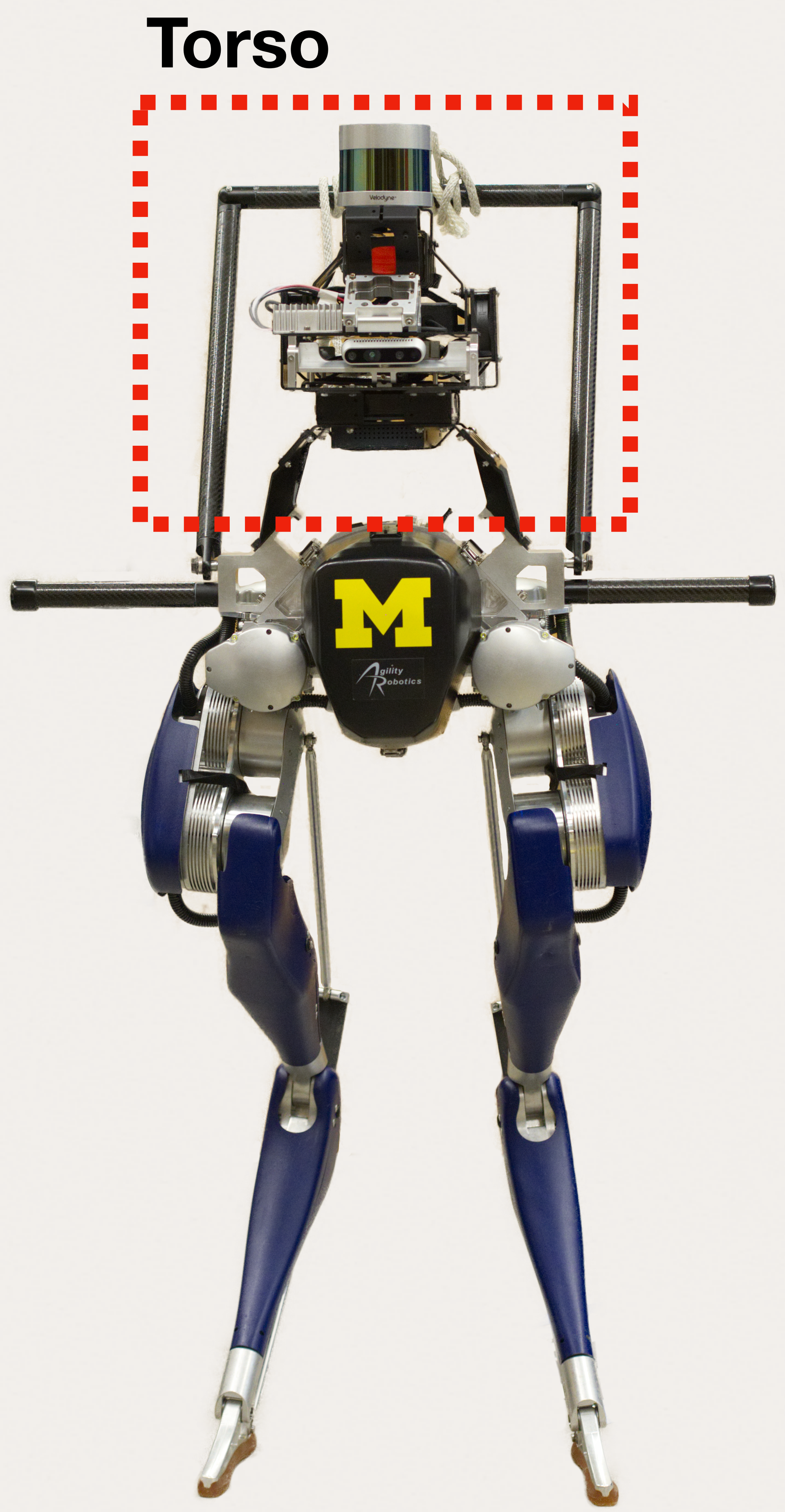
With LiDAR and camera calibrated

WHAT IS THE PROBLEM

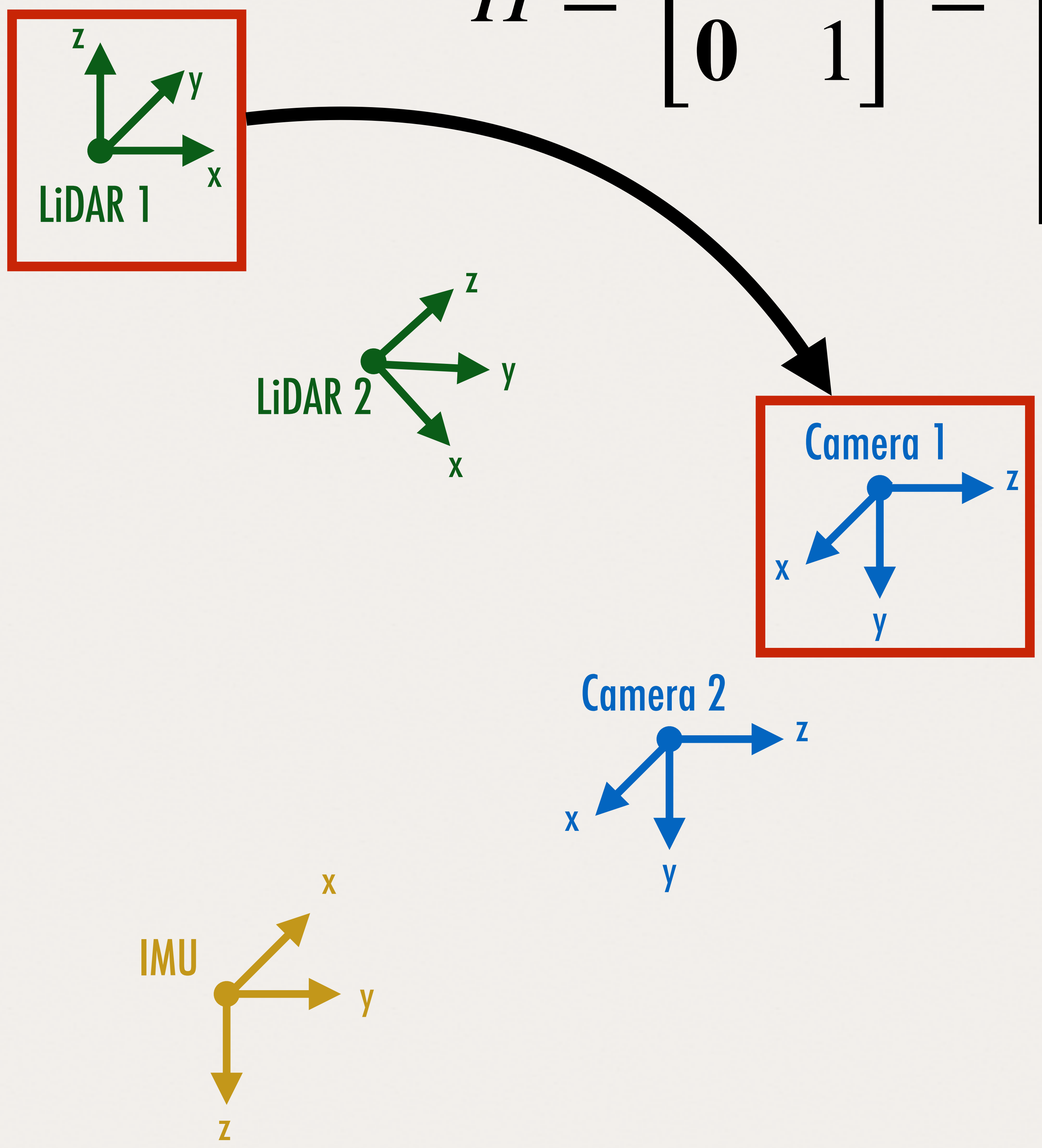




RIGID-BODY TRANSFORMATION

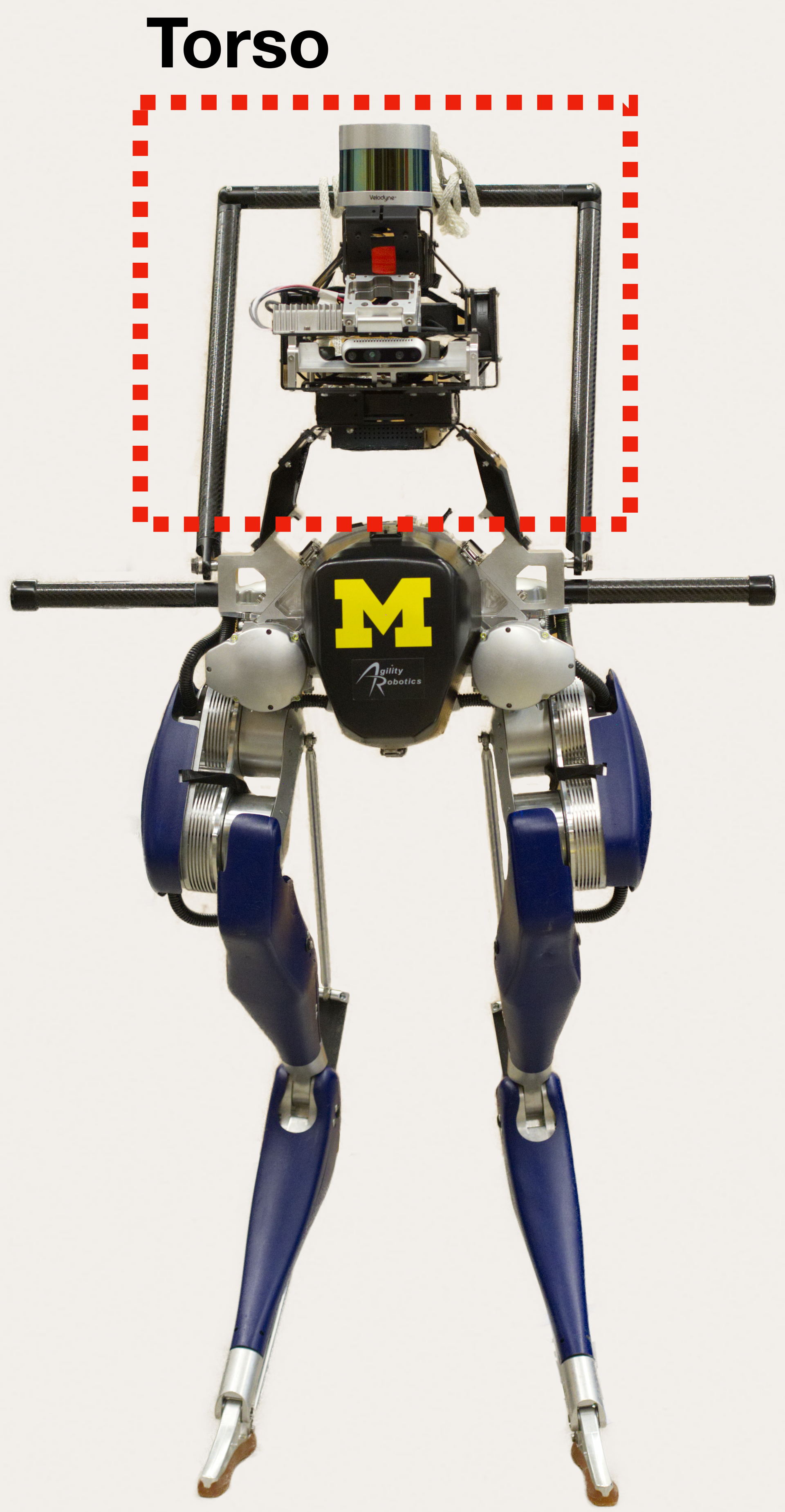


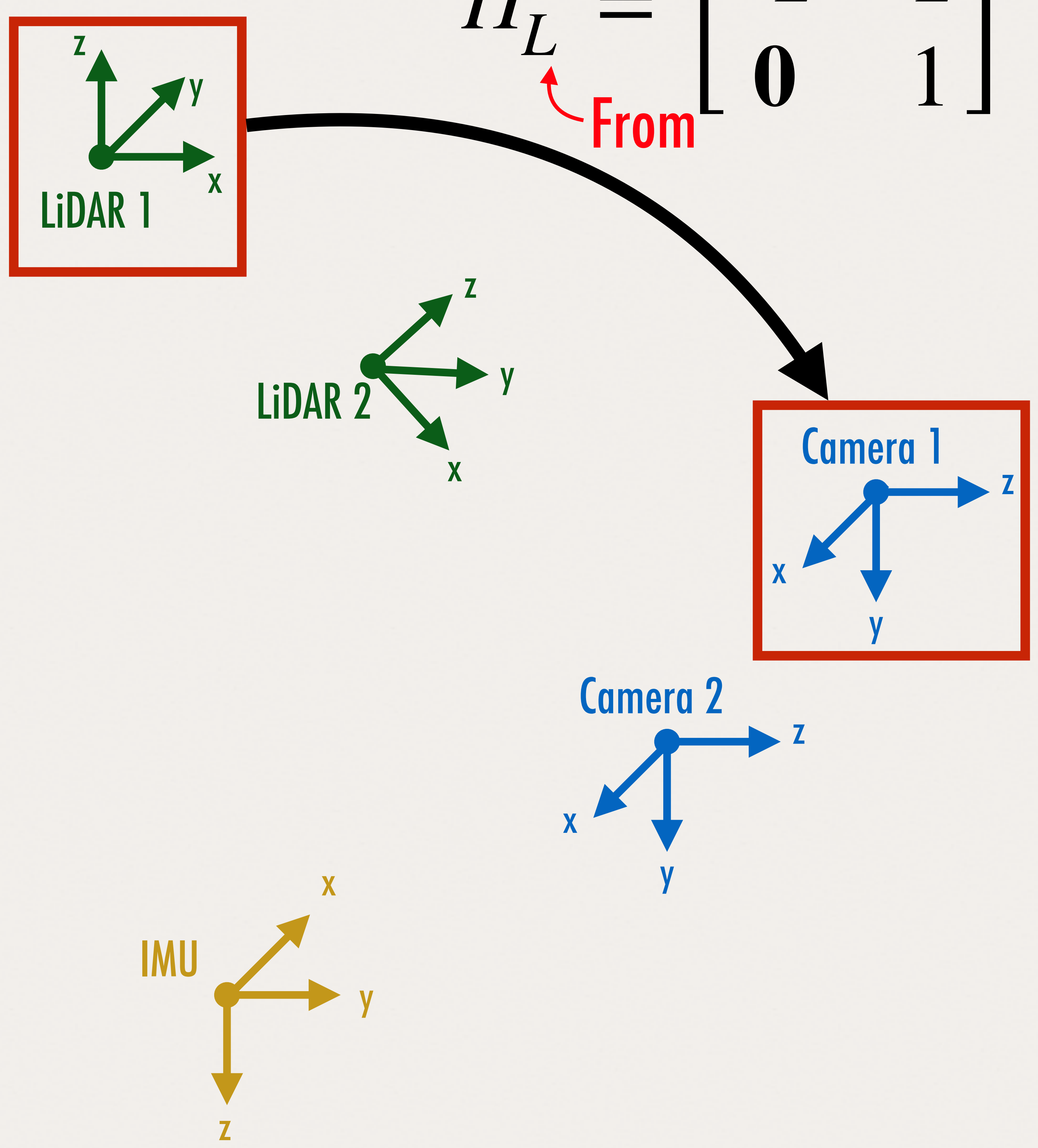
$$H = \begin{bmatrix} R & t \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Homogeneous transformation used in Project 1

RIGID-BODY TRANSFORMATION





$$H_L^C = \begin{bmatrix} R_L^C & t_L^C \\ \mathbf{0} & 1 \end{bmatrix}$$

WHAT DO WE NEED?

- Ingredients:
  - LiDAR features (X)
  - Camera features (Y)
- Goal:
  - Overlaying points from the LiDAR to an image from the camera (R_L^C, t_L^C)

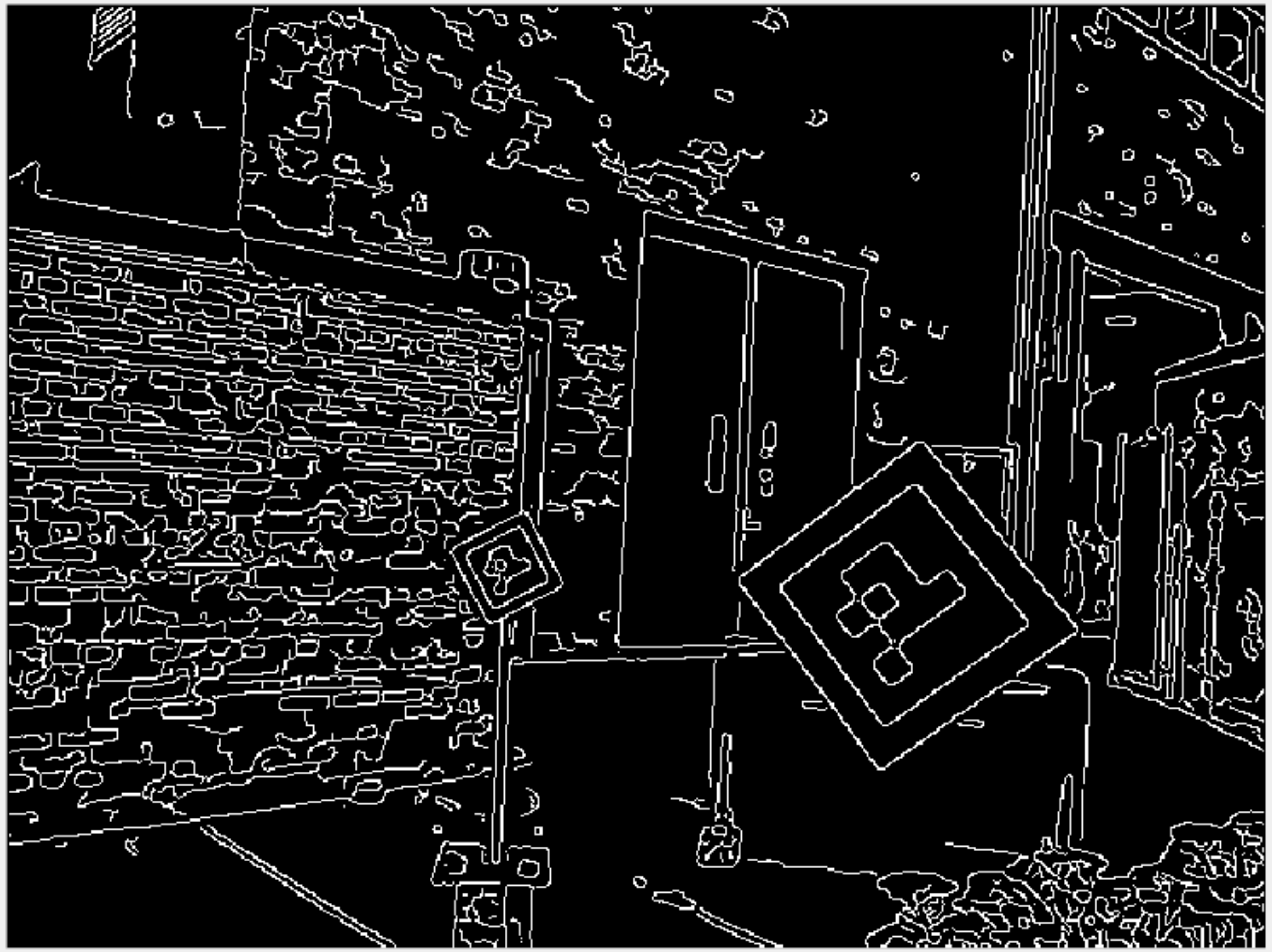
WHAT DO WE NEED?

- Ingredients:
  - LiDAR features (X)
  - Camera features (Y)

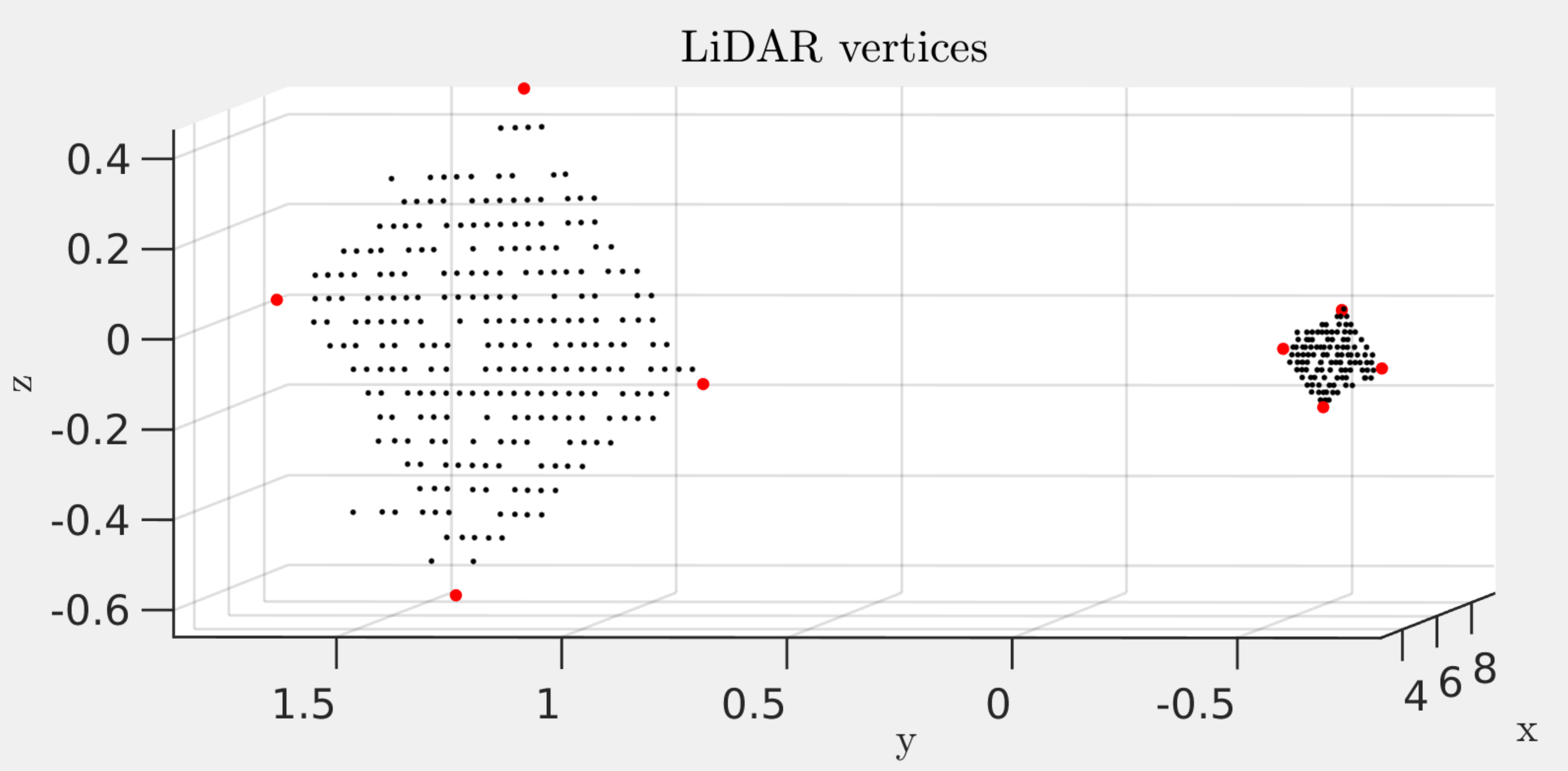
  } Specific structures in an environment

- Goal:
  - Overlaying points from the LiDAR to an image from the camera (R_L^C, t_L^C)

FEATURES: EDGES



FEATURES: CORNERS



WHAT DO WE NEED?

- Ingredients:
  - LiDAR features (X): 3D ✓
  - Camera features (Y): 2D ✓
  - 3D to 2D?
- Goal:
  - Overlaying points from the LiDAR to an image from the camera (R_L^C, t_L^C)

WHAT DO WE NEED?

- Ingredients:
  - LiDAR features (X): 3D
  - Camera features (Y): 2D
  - 3D to 2D?
- Goal:
  - Overlaying points from the LiDAR to an image from the camera (R_L^C, t_L^C)

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\, min_{R,t} f(R, t; X, Y)$$

Project points, minimize distance

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\, min_{R,t} f(R, t; X, Y)$$: Project points, minimize distance

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\, min_{R,t} f(R, t; X, Y)$$: Project points, minimize distance

$$:= arg\, min_{R,t} \qquad \Pi\left(X_i; R, t\right)$$

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\,min_{R,t} f(R, t; X, Y)$$: Project points, minimize distance

$$:= arg\,min_{R,t} \quad \Pi\left(X_i; R, t\right) - Y_i$$

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\,min_{R,t} f(R, t; X, Y)$$: Project points, minimize distance

$$:= arg\,min_{R,t} \quad \|\Pi\left(X_i; R, t\right) - Y_i\|_2^2$$

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\, min_{R,t} f(R, t; X, Y) \quad \text{: Project points, minimize distance}$$

$$:= arg\, min_{R,t} \sum_{i=1}^{4n} \|\Pi\left(X_i; R, t\right) - Y_i\|_2^2$$

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\, min_{R,t} f(R, t; X, Y) \quad \text{: Project points, minimize distance}$$

$$:= arg\, min_{R,t} \sum_{i=1}^{4n} \|\Pi\left(X_i; R, t\right) - Y_i\|_2^2$$

$$H_{k+1} = H_k - s[\nabla f(H_k; X, Y)]^\top \text{: the gradient descent}$$

LiDAR features (X)
Camera features (Y)

HOW TO FORMULATE

Problem:

$$\left(R_L^{C^*}, t_L^{C^*}\right) := arg\,min_{R,t} f(R, t; X, Y) \quad \text{: Project points, minimize distance}$$

$$:= arg\,min_{R,t} \sum_{i=1}^{4n} \|\Pi\left(X_i; R, t\right) - Y_i\|_2^2$$

$$H_{k+1} = H_k - s[\nabla f(H_k; X, Y)]^\top \text{: the gradient descent}$$

$$H_{k+1} = H_k - s\left[\nabla^2 f(H_k; X, Y)\right]^{-1}[\nabla f(H_k; X, Y)]^\top \text{ : the Hessian}$$

ROBOTICS

BIPED ROBOTICS LAB@MICHIGAN

LIVE DEMO

1. GitHub
2. Download dataset
3. Put them under ROB101_data folder
4. Run rob101_optimization.m

BACKUP SLIDES

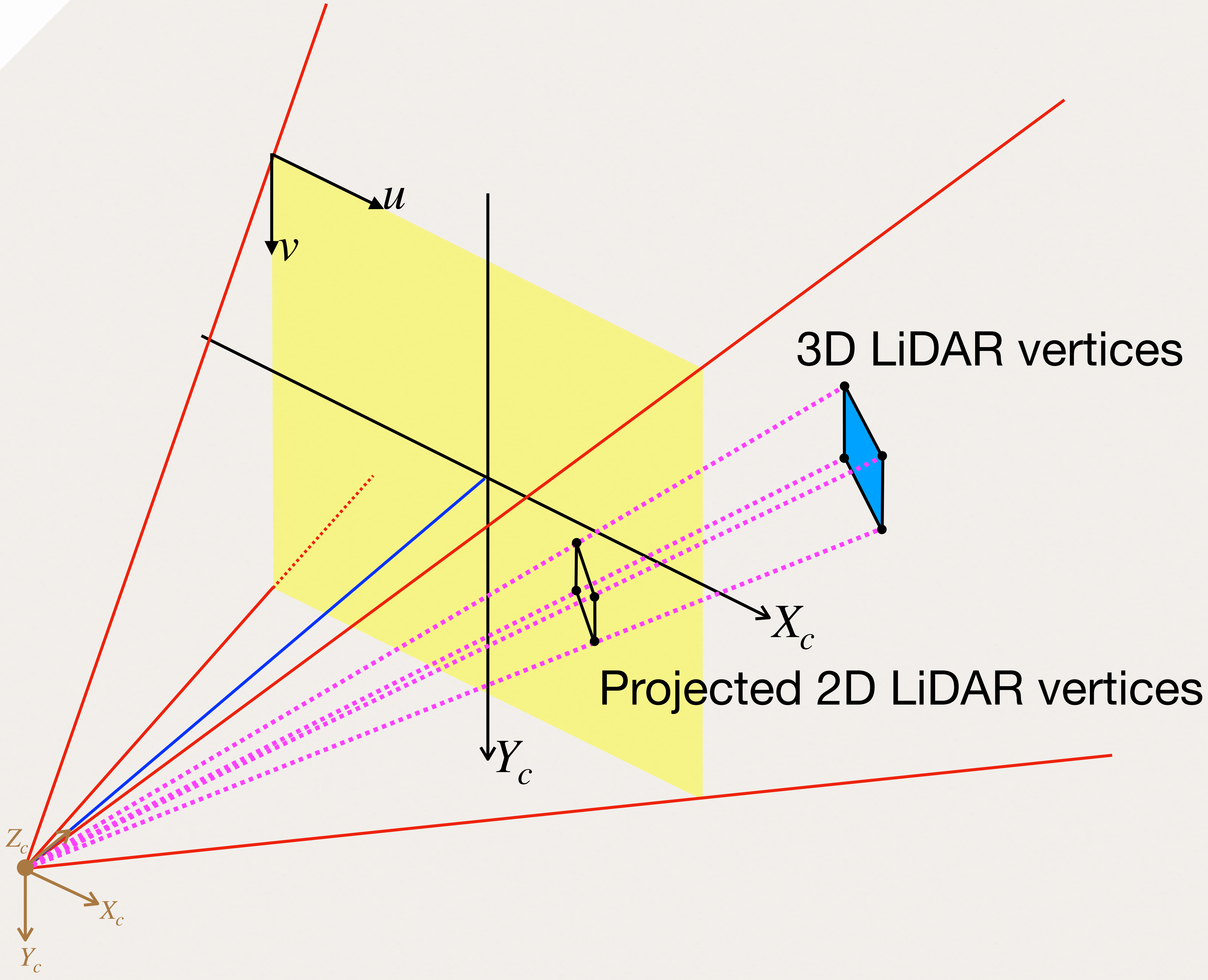
HOW TO REPRESENT R?

If you dig in more…

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \iff R = \exp(\mathrm{skew}([\omega_1, \omega_2, \omega_3]))$$

Power of Lie group!

PROJECTION MAP: 3D POINTS TO 2D POINTS



$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}}_{\text{intrinsic parameters}} \begin{bmatrix} 1_{3\times3} \\ 0_{1\times3} \end{bmatrix}^{\top} \underbrace{\begin{bmatrix} R_L^C & t_L^C \\ 0_{1\times3} & 1 \end{bmatrix}}_{\text{extrinsic parameters}} \underbrace{\begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}}_{\text{LiDAR points}}$$

$$Y_i = [u \quad v \quad 1]^{\top} = \left[\frac{u'}{w'} \quad \frac{v'}{w'} \quad 1\right]^{\top},$$

$$\Pi\left(X_i; R, t\right) := Y_i$$

KNOW MORE!

Paper: Improvements to Target-Based 3D LiDAR to Camera Calibration

GitHub: https://github.com/UMich-BipedLab/extrinsic_lidar_camera_calibration

BIPEDLAB@MICHIGAN

Lab Website!

YouTube Channel!