Imitation Control of a Robotic Arm Using the Panasonic 3D LiDAR

October 28th, 2020

Max Davidowitz
Lucas Watson

Problem

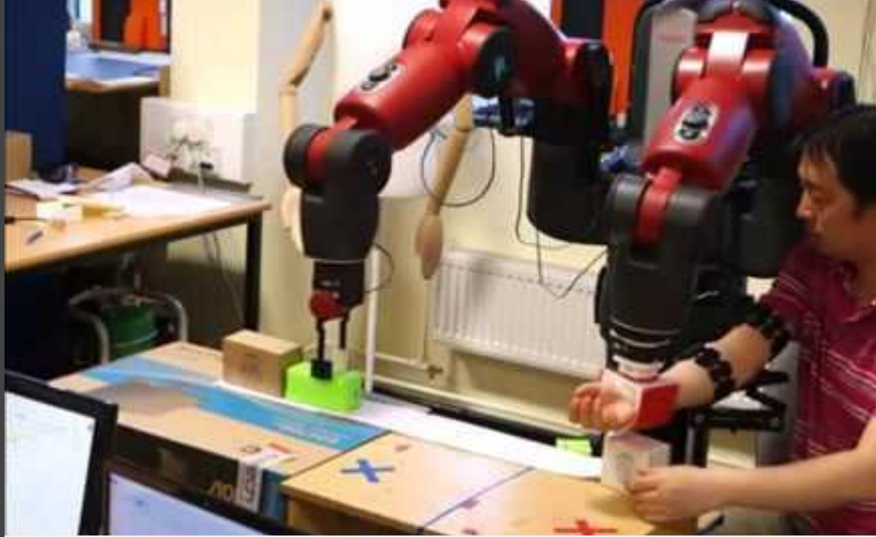
- Programming robot arms is cumbersome and requires expert knowledge.
  - Cobots can be hand-guided, but require setting waypoints which is slow.

- Instructing a robot arm to optimally accomplish a task requires the task be programmed explicitly.

- Industrial and collaborative robot arm market:
  - $190B Industrial automation market (2018)
  - ~$20B Industrial robot arms
  - $2B collaborative robotics market value in 2020

Solution

- A robot arm which imitates a human arm.
  - Direct, real-time control of the arm with a visual feedback loop.
  - To approach a task, a human demonstration instructs the motion of the robot arm.
  - Human-controlled demonstrations of completing a task train the arm to later optimally solve for the task.



Similar solutions

- Shadow Robot
  - Focused on dextrous teleoperation of robot arms.
  - Requires expensive sensors which attach to the arms and hand, coupled with complex end-effectors.
  - Appear to be targeting remote teleoperation.
  - The user directly controls the robot arms.

- SE4
  - Uses virtual reality headset to provide overlay on what the robot sees.
  - The user uses 6 DOF controllers to manipulate the scene, and the robot arm then operates on human instructions.
  - The user commands tasks.

Similar solutions

- Toyota Research Institute
  - Autonomous home robot with teleoperation capabilities.
  - Similar to SE4, uses a virtual reality headset and controllers to operate the robot. However in this case, the user directly controls the robot.
  - The user directly controls the robot.

- Surgical robots
  - A model of the robot kinematics is directly manipulated by the user, which in then controls the robot.
  - The user directly controls the robot.

Use Case: Home Robotics

I need a system that
- puts my dishes away
- can set the dinner table
- do a diverse range of cleaning
- takes care of my grandmother

I don’t have:
- time to fine tune each task for the robot
- expertise to set up a complex teach system

This solution will allow for:
- human-in-the-loop exception handling
- learning in an unstructured environment
- safe and intuitive human-robot interaction

Home Robotics

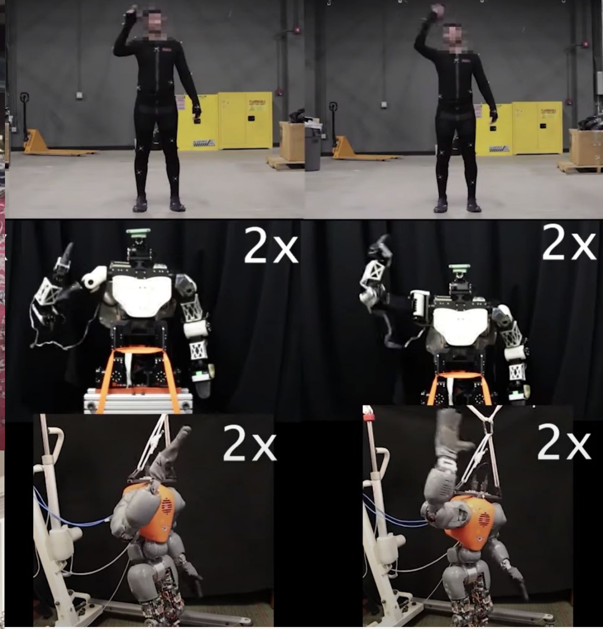
Hello Robot
Startup building a robot platform

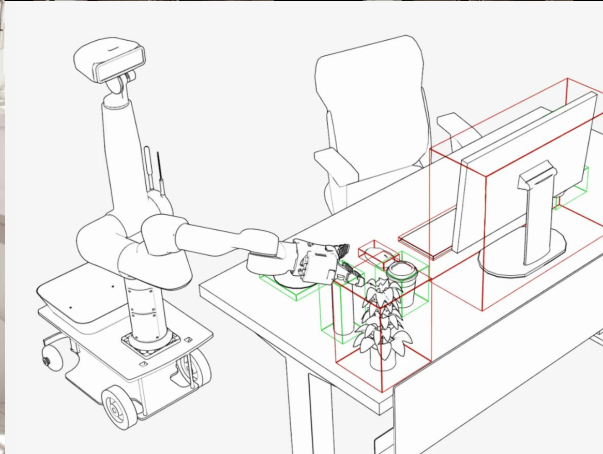
Disney Research
Using human training to control humanoid robots

Toyota Research Institute
Building a home robot for chores

Google X – Everyday robot project
Building a home robot, first tasked with sorting waste

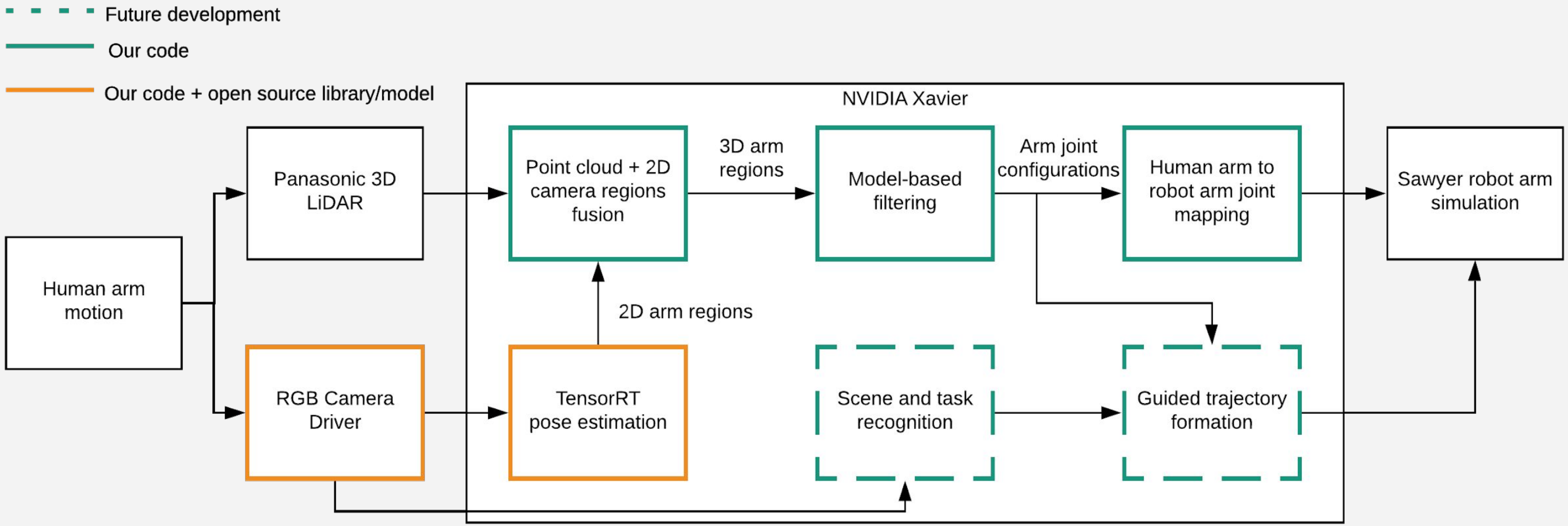
Amazon
Rumors about a home robot in development





Demo Video

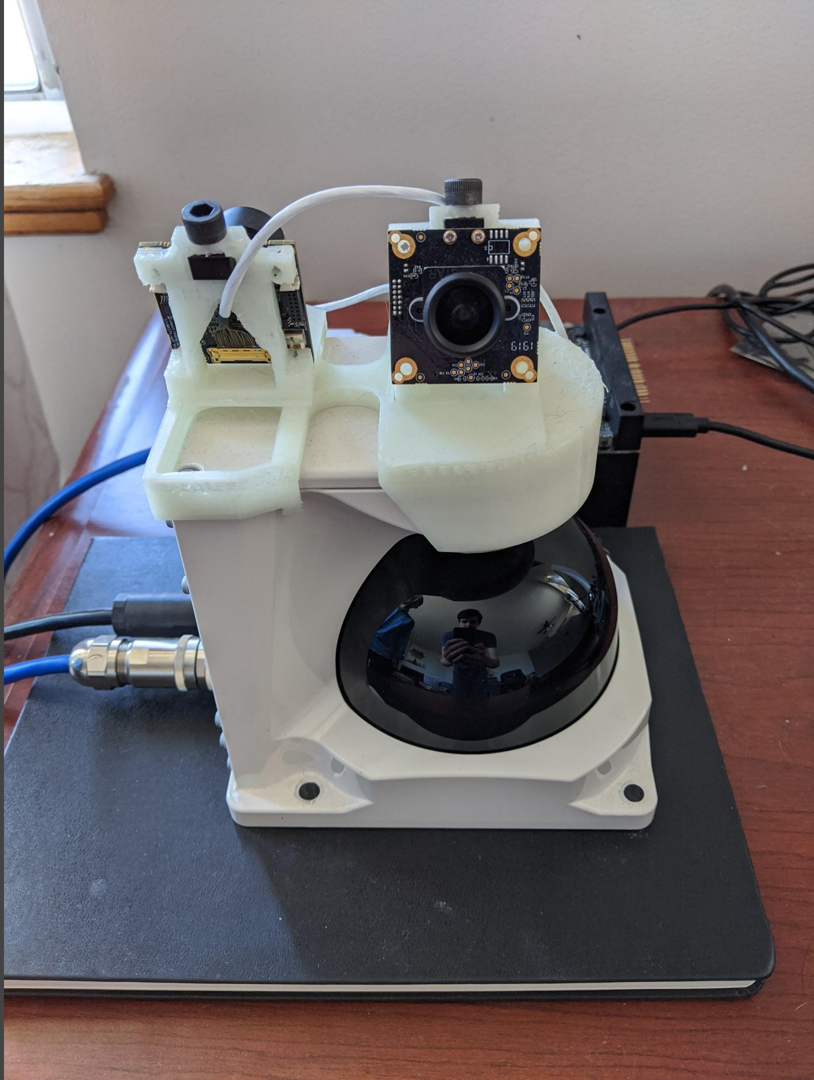
SYSTEM OVERVIEW



Sensor Configuration

- LiDAR
    - ~7.5 Hz, ~3.75 Hz if multi_frame = true
    - 15 scan lines, best compromise between refresh rate and VFOV
    - Scan mode 3
    - **multi_echo = false,** this fixed our frame dropout issue

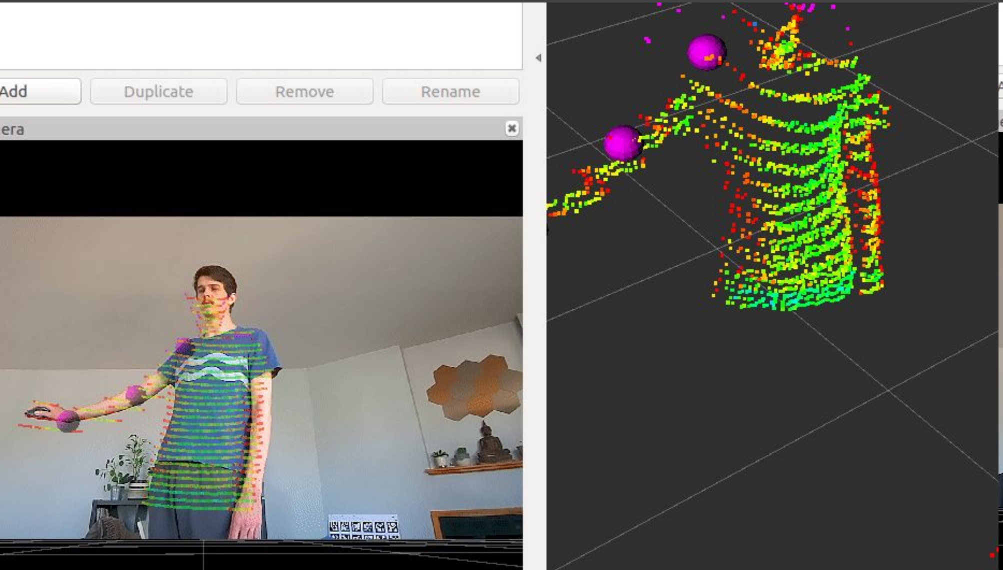
- Camera
    - Leopard imaging IMX274
    - 1280x720 at 60 fps when utilizing NVIDIA's Argus camera library
    - Denoising included in Argus



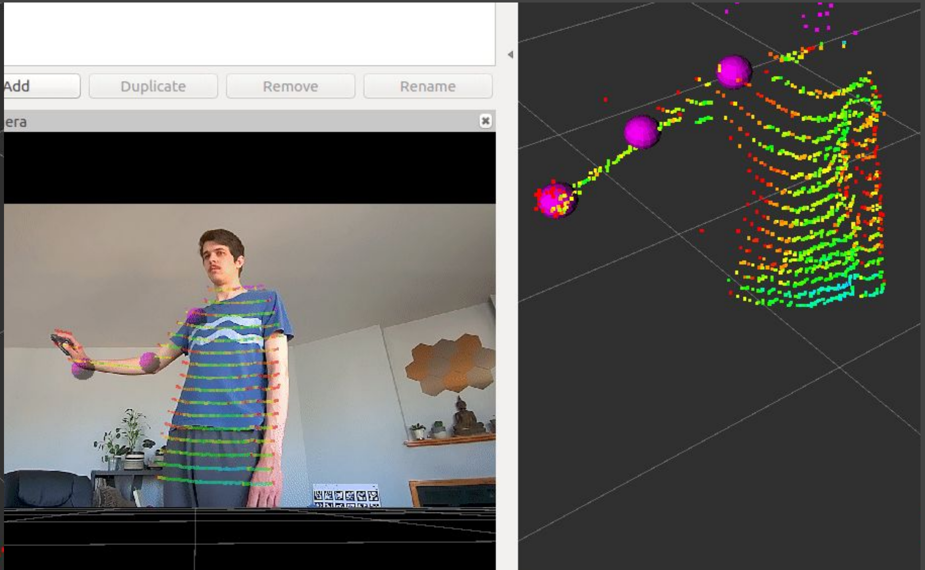
Sensor Configuration: multi_frame

- Much better fusion with 2D keypoints from camera with multi_frame turned on.
- However, slower update rate.

multi_frame = true

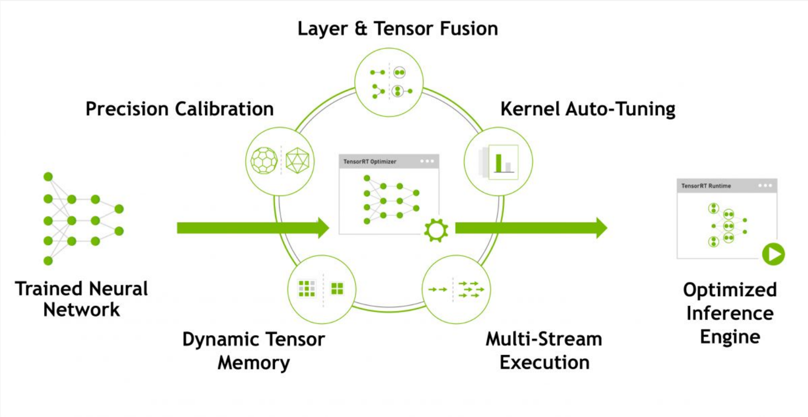


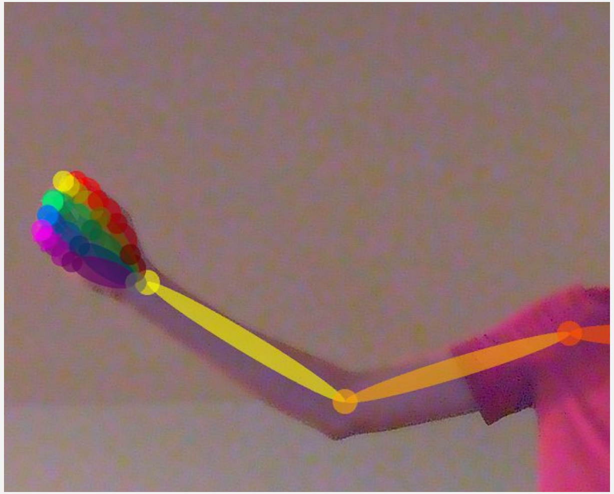
multi_frame = false



Keypoint Detection

- Started by using OpenPose, a popular human pose estimation library.
- Optimized a more recently release Densenet trained with the same COCO keypoint dataset with TensorRT.
    - Prior to TensorRT optimization: 5-8 FPS
    - With TensorRT: 16-20 FPS
    - Model and net came pre-trained
- Densenet model doesn’t output hand data. Experiments with OpenPose showed getting hand data is very slow.
- Next step: custom train a network with limited information about the hand, and optimize with TensorRT.
    - Additionally, remove parts of network estimating leg and body keypoints.



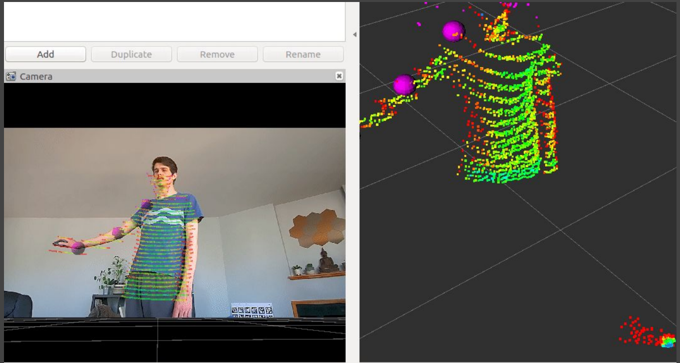


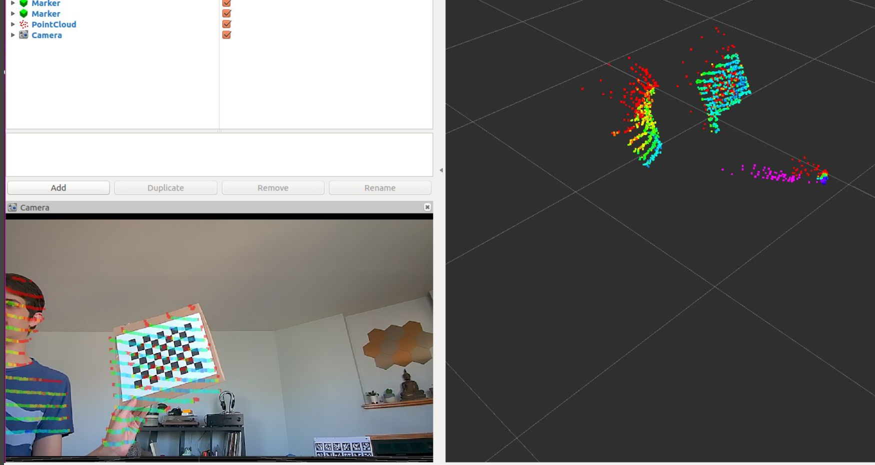
Fusion

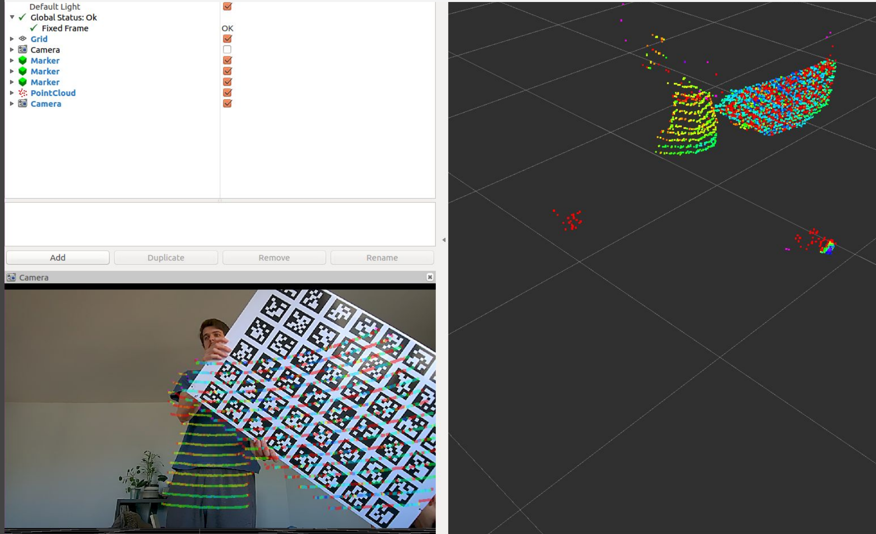
- Calibration of camera intrinsics.
- Calibration of rigid transform between camera and LiDAR (extrinsics).

- Take pose keypoints from camera frame, and retrieve depth information from point cloud.

- Tried region segmentation of point cloud, but found this didn't improve estimation of arm.

- Result: 

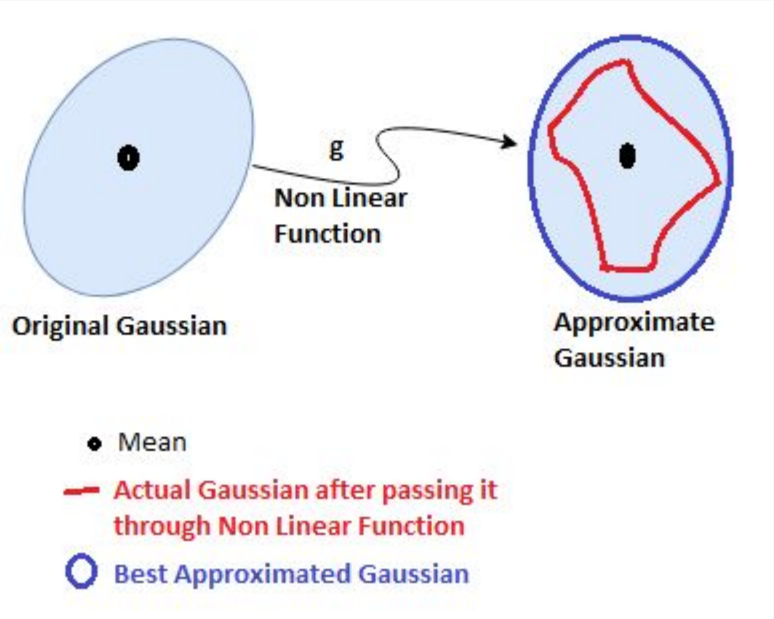


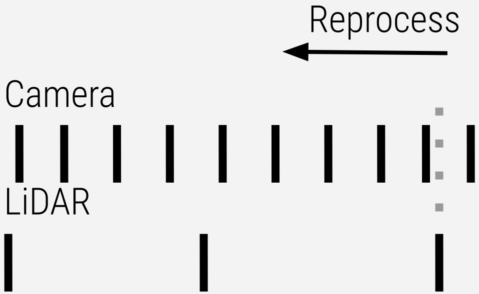


Filtering

- Created a kinematic model for estimating the pose and velocity of a human arm.

- Goal: use an Extended Kalman Filter to estimate the state of the human arm from both image keypoint and point-cloud measurements.
  - Uncertainty (covariance) is different for image keypoint measurements, large covariance in depth dimension, using previous point-cloud measurement as estimate.
  - Variable time update filter is difficult, requires reprocessing on sensor history on each new lidar scan.
  - Issues with estimation drifting in-between LiDAR scans, still in progress.

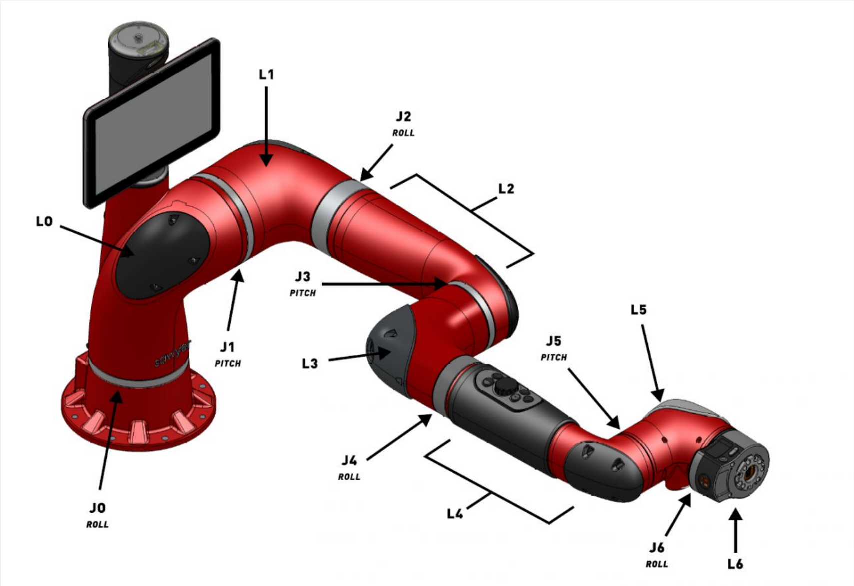




Kinematic Mapping

- Mapping created between human arm joint angles and robot arm joint angles.
  - Difficulties in different joint abilities between human and robot

- Future development:
  - Trajectory generation with kinematic model of sawyer arm to reach task goal.

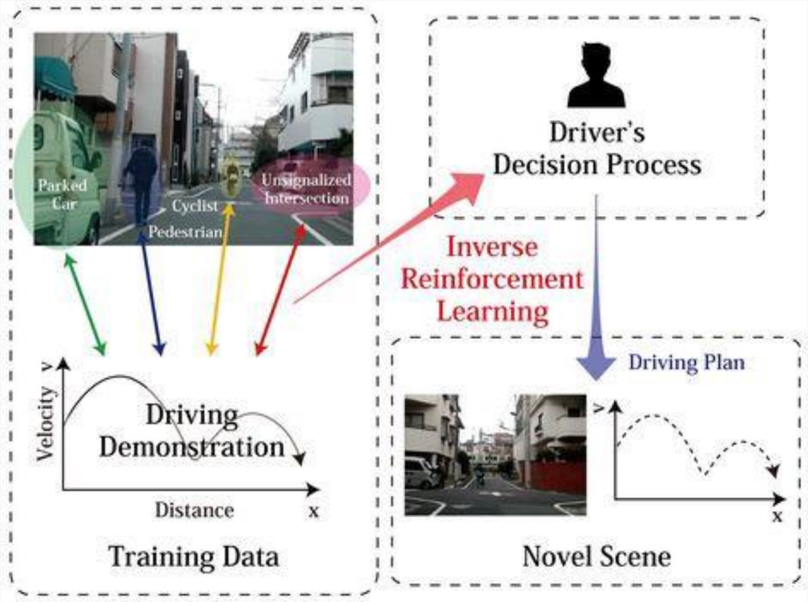


Current Control: J0, J1, J2

Inverse Reinforcement Learning

- Given human demonstrations of accomplishing a task, we may train a model to learn the optimal approach to this task.

- More simple, and easier to generalize, than trying to directly learn everything.
  - Other reinforcement learning methods try to engineer reward functions for specific tasks. **This is learning from the ground-up.**
  - With IRL, the task can be inferred from human demonstration. We're interested in **learning the best approach to the task**, and start with more data.

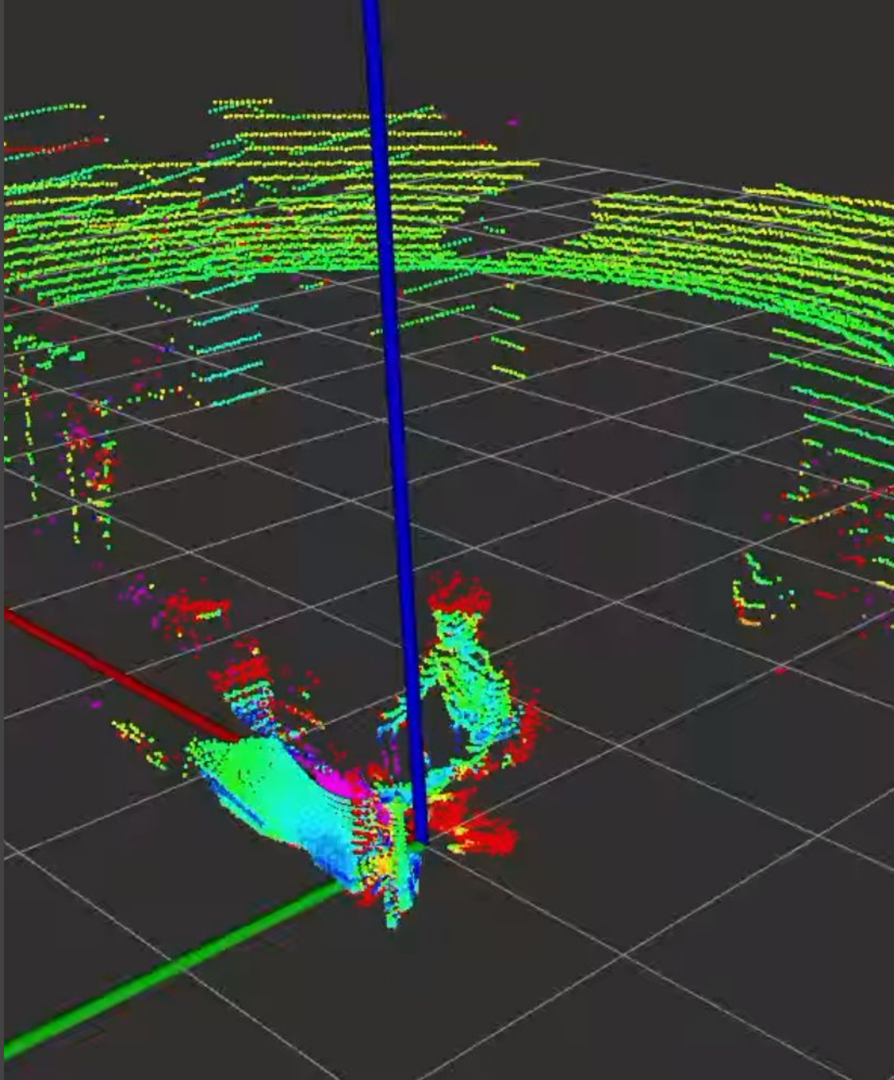


FUTURE DEVELOPMENT

- Integration with physical robot arm.
    - Gripper functionality.
    - Scene recognition to aid task completion.

- Create an example home robotics problem to study human-guided control.
    - A task that may be repeated, but have significant variation between different occurrences.

- Inverse reinforcement learning.
    - Use human demonstrations of a task, allow the arm to learn a generalized, optimal approach to the task.



QUESTIONS

MEDIA

- FANUC Industrial Robots at AUDI: https://www.youtube.com/watch?v=rbki4HR41-4
- UR cobot arms are tested in Denmark: https://www.roboticsbusinessreview.com/manufacturing/cobot-arms-grippers-offer-value-imts/
- UR5 using Soft Robotics Gripper: https://www.youtube.com/watch?v=Z3TC-PLqGP4
- TensorRT: https://developer.nvidia.com/tensorrt
- EKF: https://towardsdatascience.com/the-unscented-kalman-filter-anything-ekf-can-do-i-can-do-it-better-ce7c773cf88d

Continuation Timeline

LUCAS WATSON
MAX DAVIDOWITZ

NOVEMBER

Refine system in simulation

Continue to develop integration with the 7 DoF Sawyer arm in simulation. Improve kinematic mapping and model-based filtering.

DECEMBER

JANUARY

Optimization of a single task

Construct a 'home robotics' task workspace to train the system with - mount Panasonic LiDAR, RGB camera(s).

FEBRUARY

Integrate with physical robot arm

Using a robot arm from MassRobotics, integrate and tune system to mimic motions of a human arm. Improve kinematic mapping.

MARCH

Scene recognition / guided trajectory generation

Begin building in scene recognition, trajectory optimization functionality and inverse reinforcement learning on a set of tasks.

APRIL

Demonstrate system capabilities to Panasonic