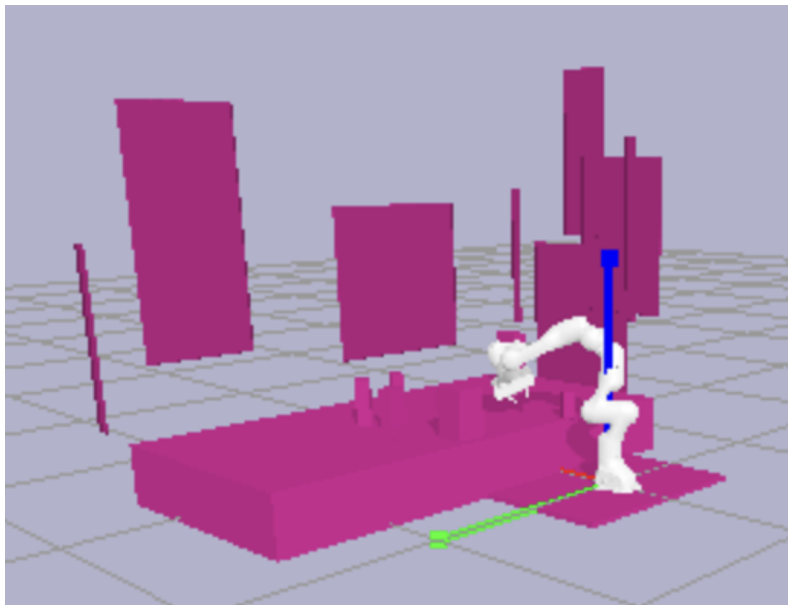
Perception Aware Planning

A. Buynitsky

Dec 10, 2024

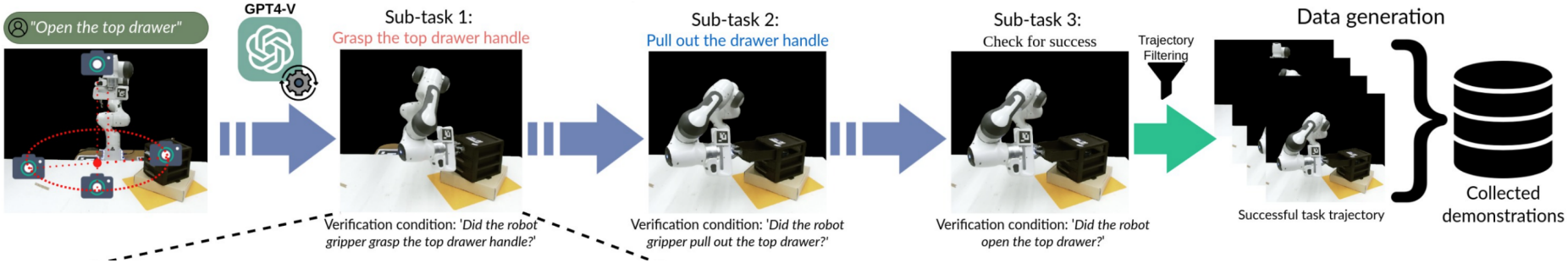


Outline

Outline

Manipulate-Anything

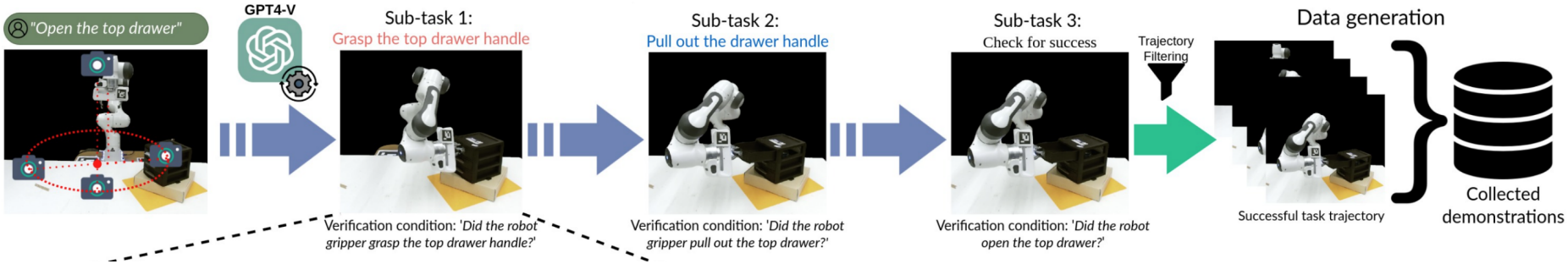
Many approaches use either a fixed viewpoint or a collection of viewpoints.



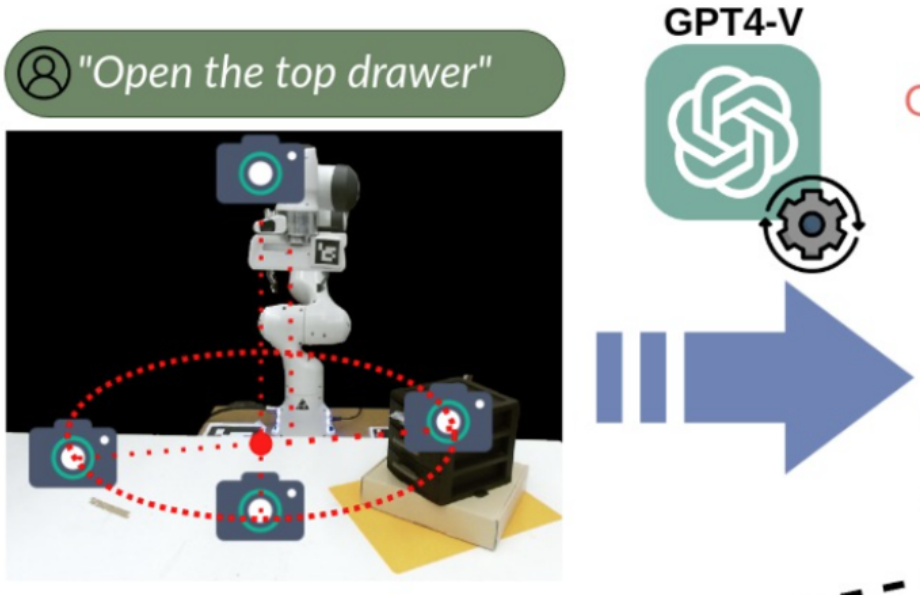
Approach: Decomposes a task into smaller subtasks with VLM

Manipulate-Anything

Many approaches use either a fixed viewpoint or a collection of viewpoints.



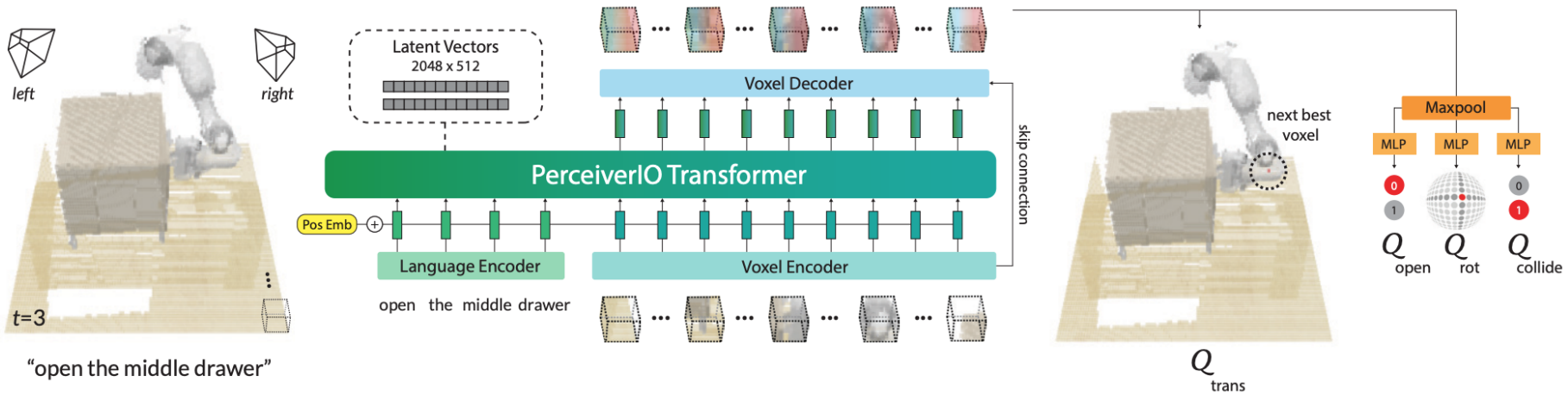
Approach: Decomposes a task into smaller subtasks with VLM



Problem: Uses 4 RGB-D cameras placed around the scene. Then Select a viewpoint or a combination of viewpoints for each subtask.

Perceiver-Actor

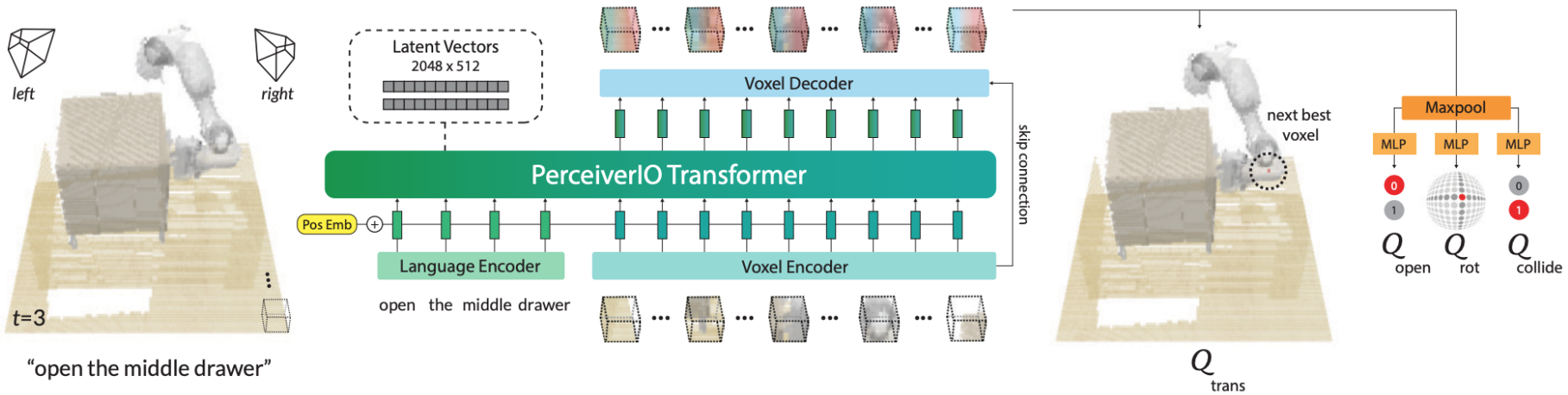
Many approaches use either a fixed viewpoint or a collection of viewpoints.



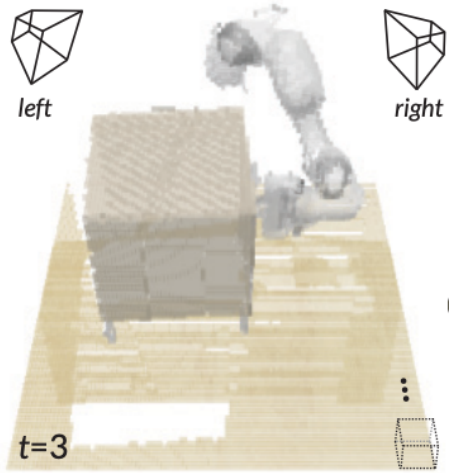
Approach: Takes language goal and RGB-D voxel as input, predicts discrete actions

Perceiver-Actor

Many approaches use either a fixed viewpoint or a collection of viewpoints.



Approach: Takes language goal and RGB-D voxel as input, predicts discrete actions



Problem: Uses 4 RGB-D cameras to construct an occlusion-free voxel representation of the scene.

Problem: Manipulation under Occlusion

The majority of approaches assume either:

- an occlusion-free view from a fixed camera
- the ability to construct an occlusion-free representation of a scene from multiple viewpoints

Most motion planners would fail to generate successful trajectories in occluded, crowded environments

Problem: Manipulation under Occlusion

The majority of approaches assume either:

- an occlusion-free view from a fixed camera
- the ability to construct an occlusion-free representation of a scene from multiple viewpoints

Most motion planners would fail to generate successful trajectories in occluded, crowded environments

Project Goal: Active Vision / Sensing in bi-manupulator systems

Outline

Observe Then Act

Use a dual-agent structure consisting of two policies:

Next-Best-View (NBV): infer optimal viewpoints

Next-Best-Pose (NBP): predict next discrete 6-DOF gripper pose

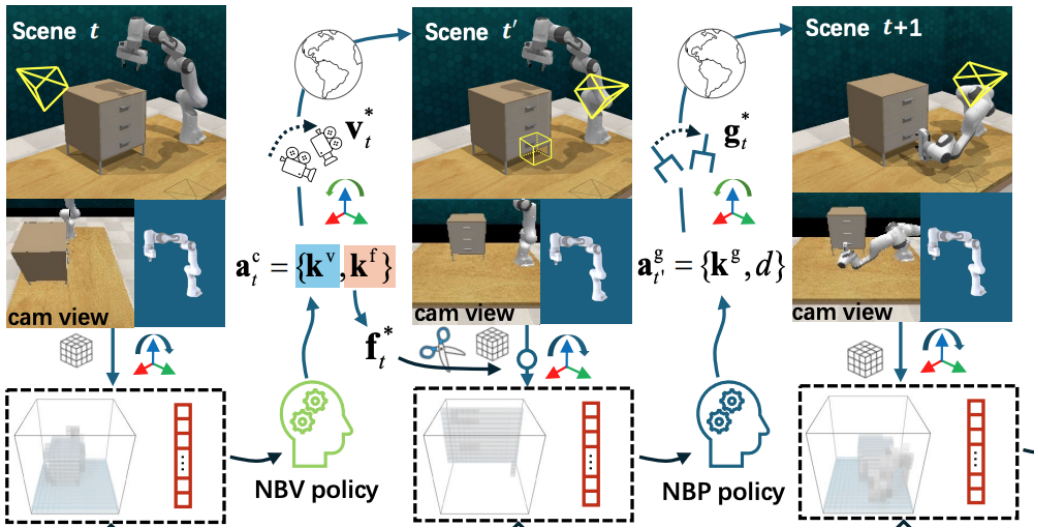
Observe Then Act

Use a dual-agent structure consisting of two policies:

Next-Best-View (NBV): infer optimal viewpoints

Next-Best-Pose (NBP): predict next discrete 6-DOF gripper pose

Formulate as POMDP, dividing one interaction into a NBV action and NBP action



Reward: Sum of whether the training episode was completed, (reached ROI), reduction of entropy in voxel occupancy, and reachible ROI.

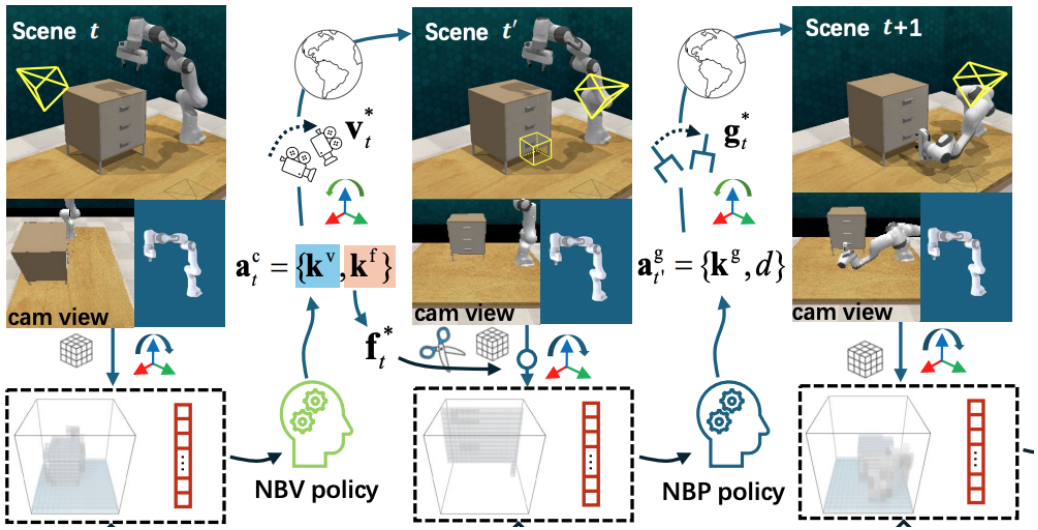
Observe Then Act

Use a dual-agent structure consisting of two policies:

Next-Best-View (NBV): infer optimal viewpoints

Next-Best-Pose (NBP): predict next discrete 6-DOF gripper pose

Formulate as POMDP, dividing one interaction into a NBV action and NBP action



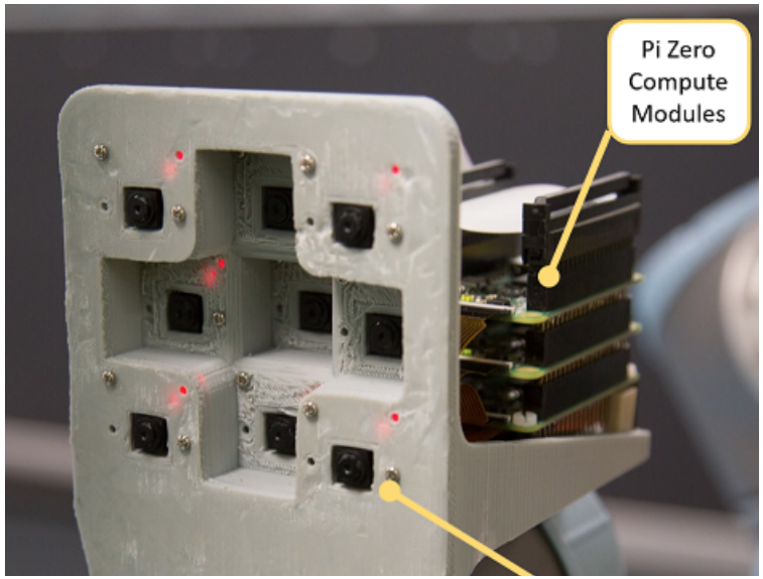
Reward: Sum of whether the training episode was completed, (reached ROI), reduction of entropy in voxel occupancy, and reachible ROI.

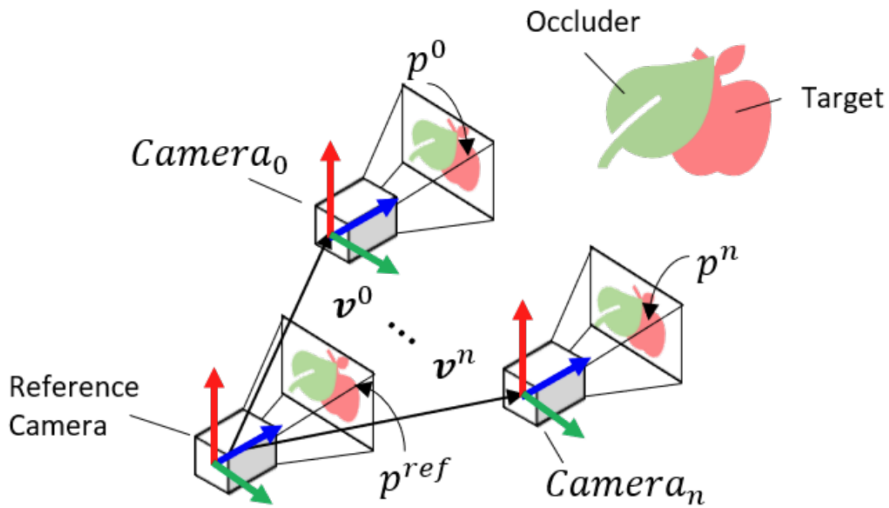
Limitation: Camera position is limited to the upper half-plane at a fixed radius

3D Move To See

Solve for gradient of the objective function representing the goal of the system
Objective Function consists of two parts:

- occlusion of the target object
- robot manipulability score





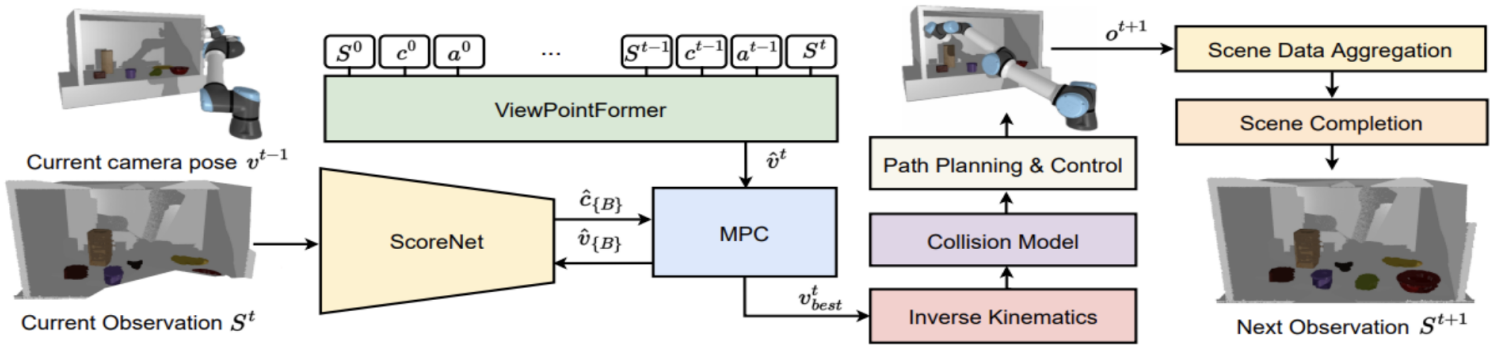
Limitation: Camera can get caught in a local minimum

Active Sensing and Planning in Cluttered Envs

Goal: maximize scene convergence in the least time steps

$$\max_{v^t \sim \pi} \min_{T} \sum_{t=0}^{T-1} \phi(S_o^{t+1}(v^t) \setminus S_o^t, S)$$

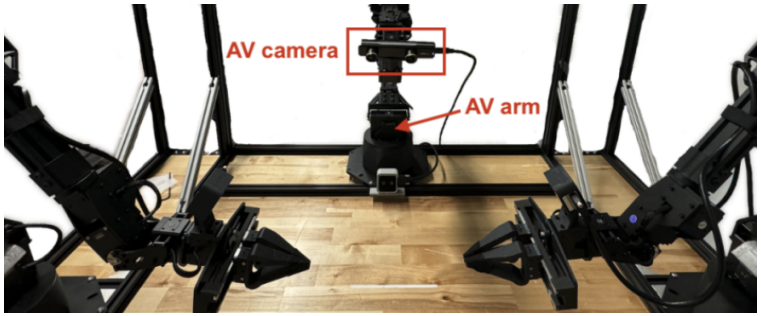
Score Net: NN approach forecast scene convergence at various viewpoints
Trajectory Generation: Generate the next viewpoint that maximizes scorenet score via MPC or transformer



Limitation: Single manipulator exploring an unknown environment

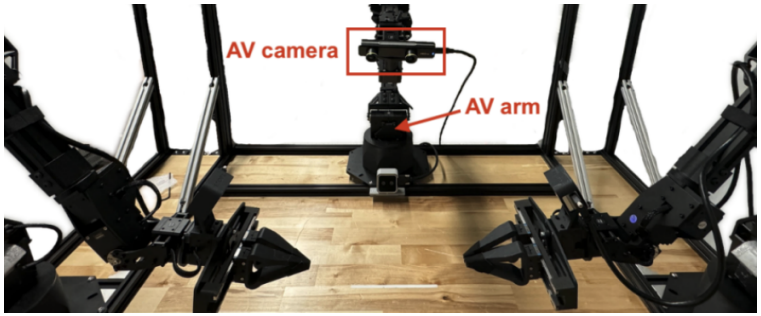
ACT + AV-Aloha

Robotic System: 3 total robotic arms (2 manipulators + one camera)



ACT + AV-Aloha

Robotic System: 3 total robotic arms (2 manipulators + one camera)

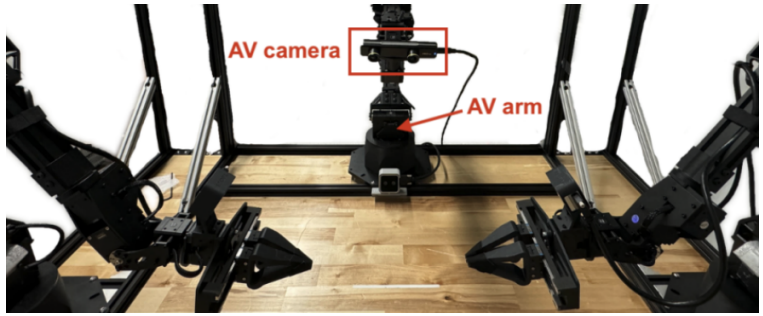


Data Generation: Data for manipulator arms collected from human grippers

- Data for manipulator arms collected from human grippers
- Data for AV arm collected from Oculus headset

ACT + AV-Aloha

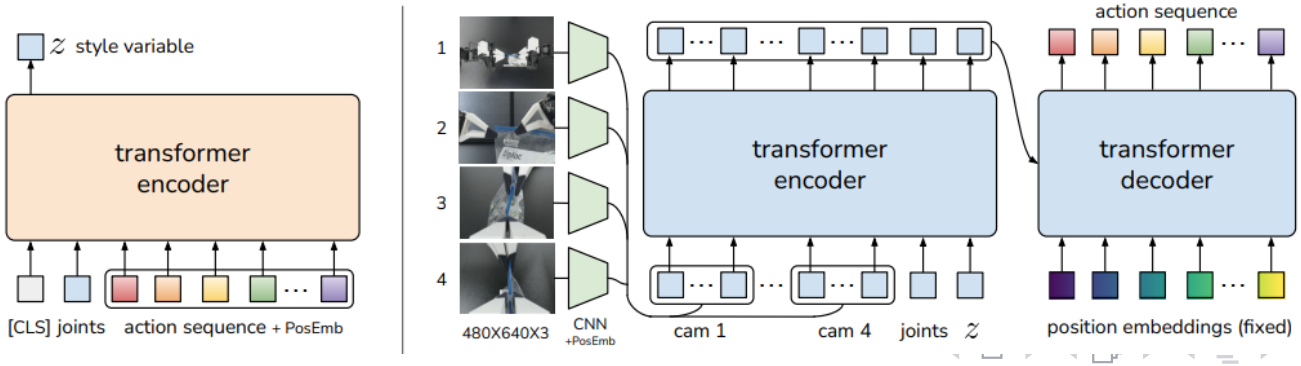
Robotic System: 3 total robotic arms (2 manipulators + one camera)



Data Generation: Data for manipulator arms collected from human grippers

- Data for manipulator arms collected from human grippers
- Data for AV arm collected from Oculus headset

Training: Train with ACT (Action-Chunking Transformer) - predict the next 50 actions at each time step via Imitation Learning



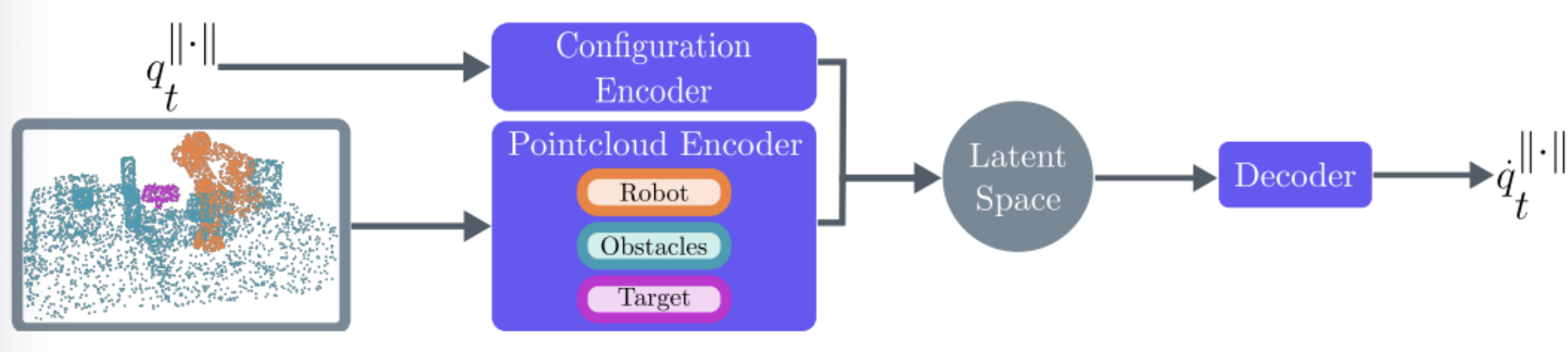
Outline

Mπnets

Generate collision-free smooth motion plans from a single-depth camera observation.

Training: Trained on 3 million trajectories in 500,000 environments with behavioral cloning to predict c-space waypoints.

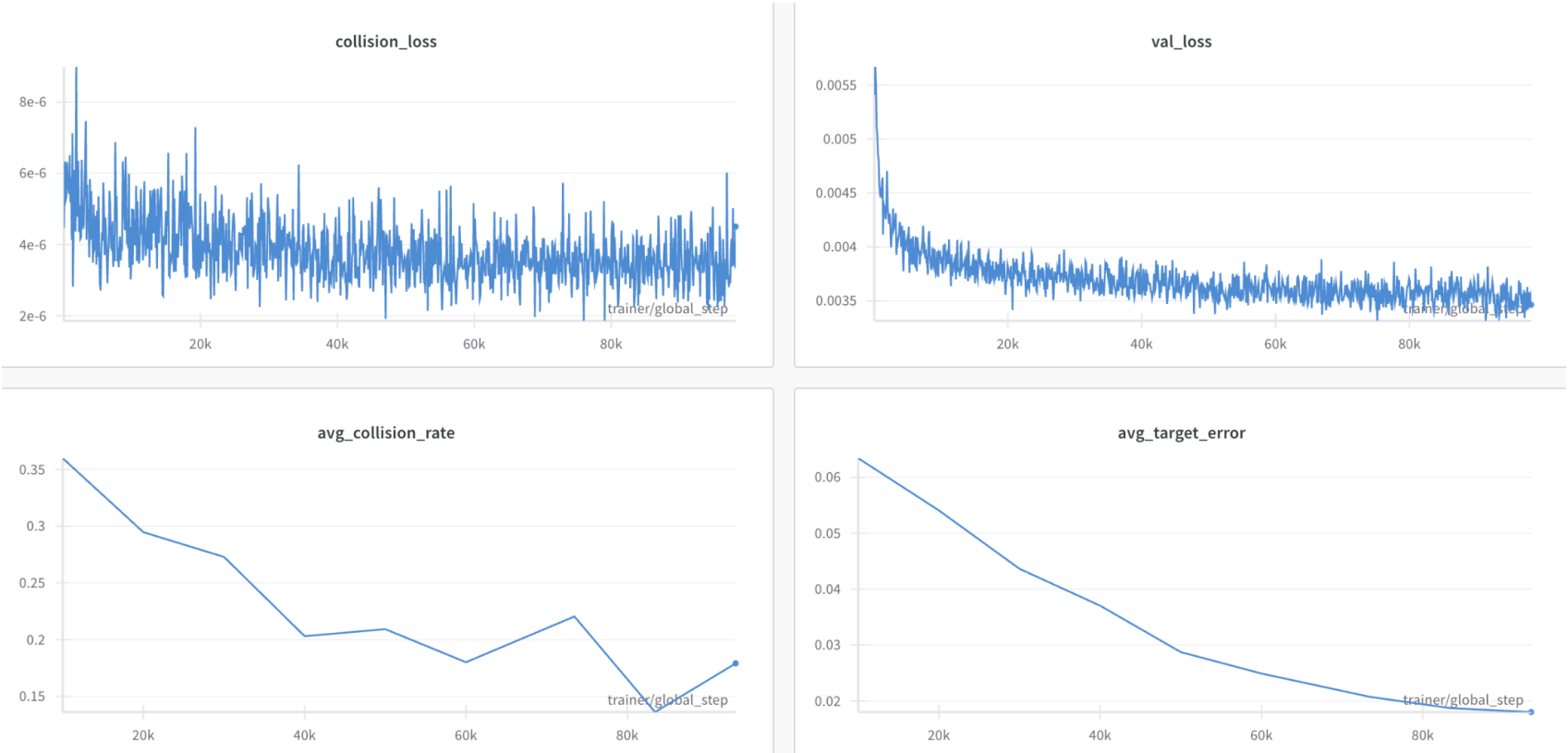
Architecture:



Outline

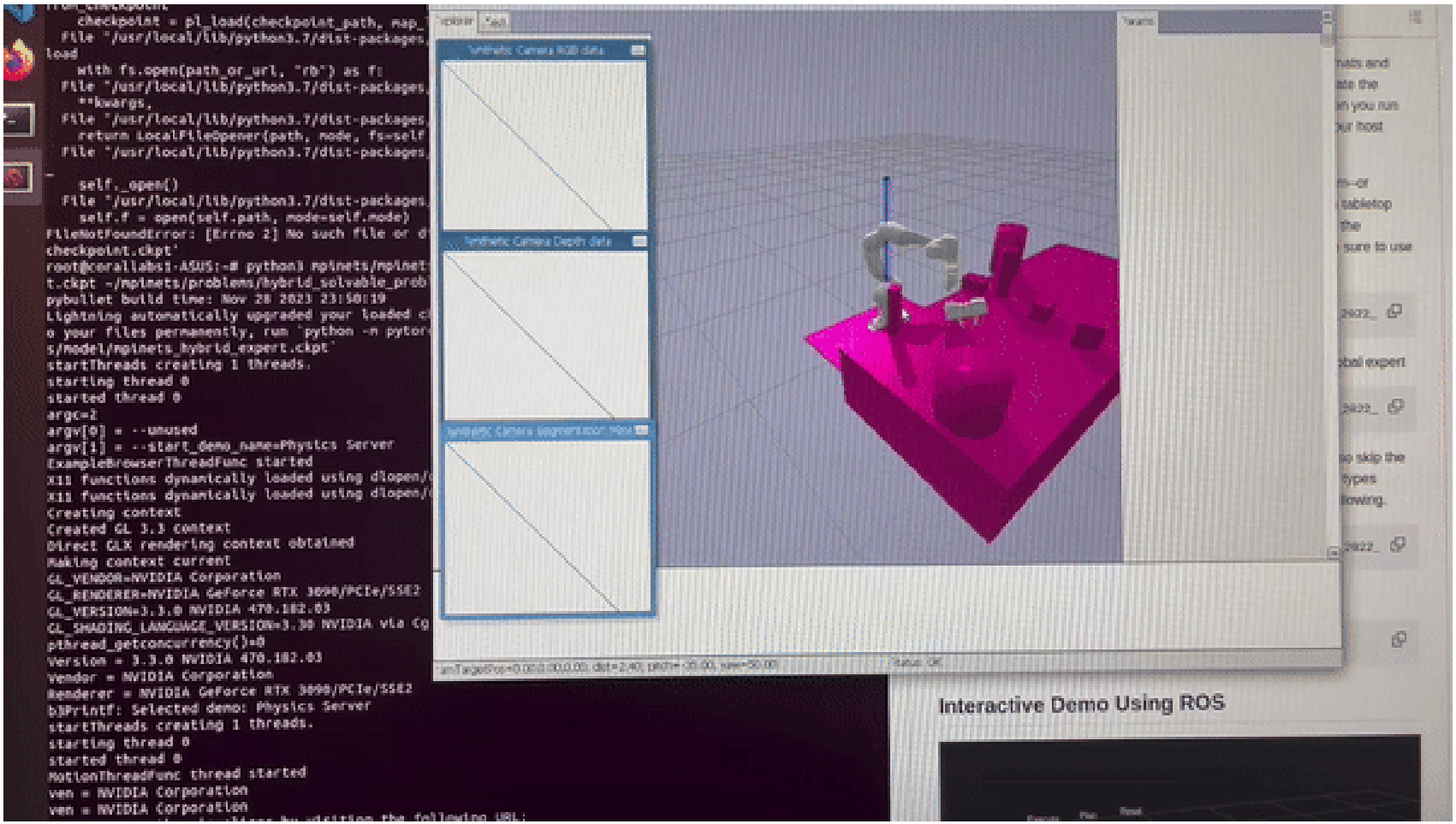
Reproducing Results

Faced problems with setting up the repo (dell xps + vm failed) and CoRAL server lost power multiple times...



Reproducing Results

Faced problems with setting up the repo (dell xps + vm failed) and CoRAL server lost power multiple times...

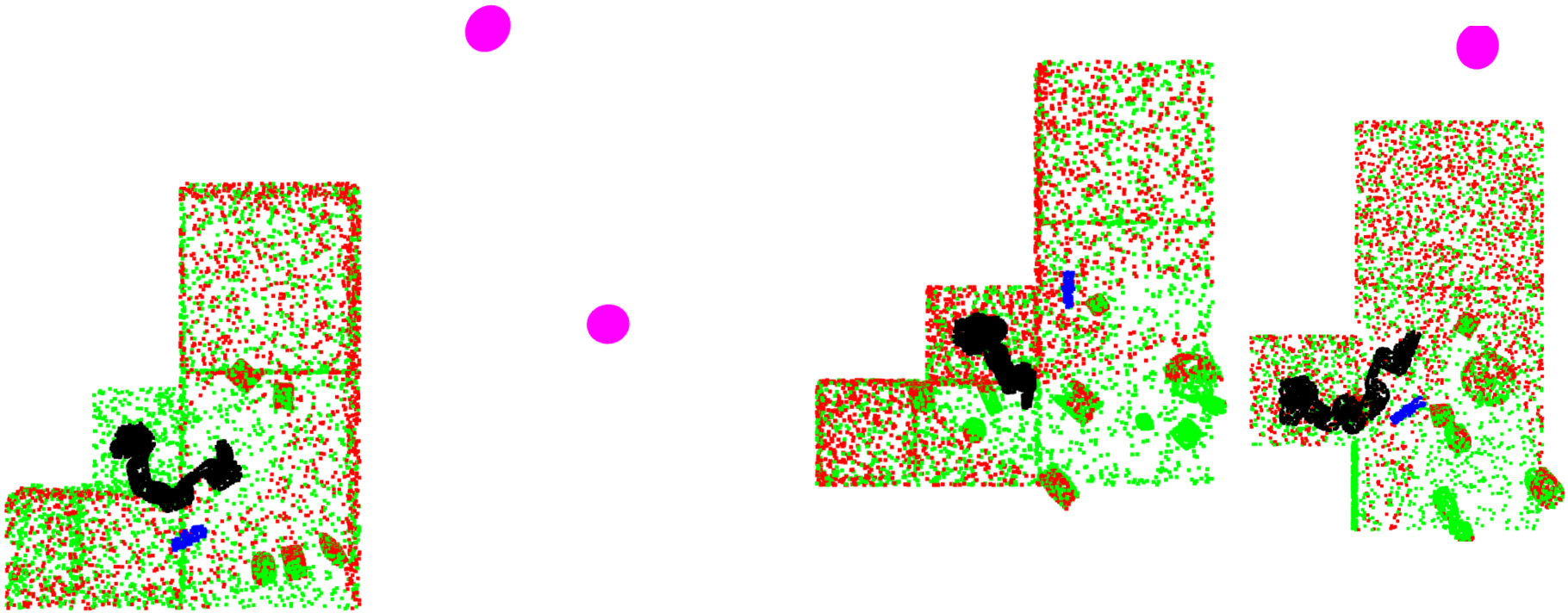


Creating Custom dataset (Part 1)

Mπnets input consists of 3 point clouds: obstacles, target location, robot point cloud

A dataset with occlusions will project the obstacle pointcloud on some camera locations.

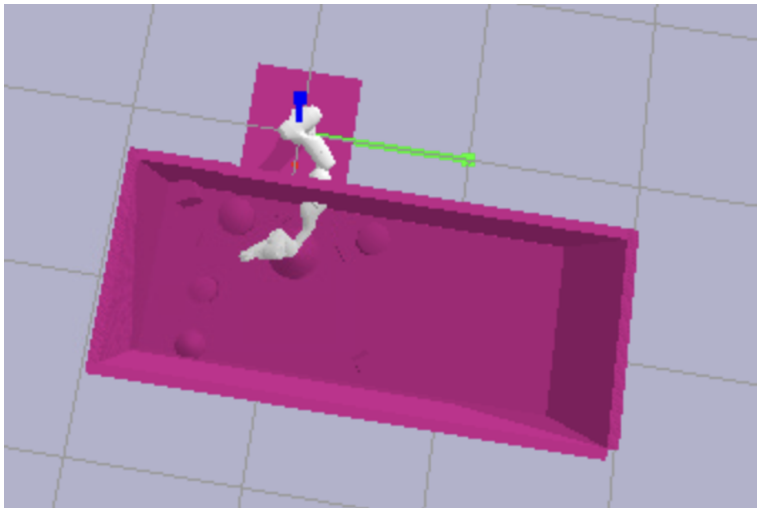
Dataset Samples:

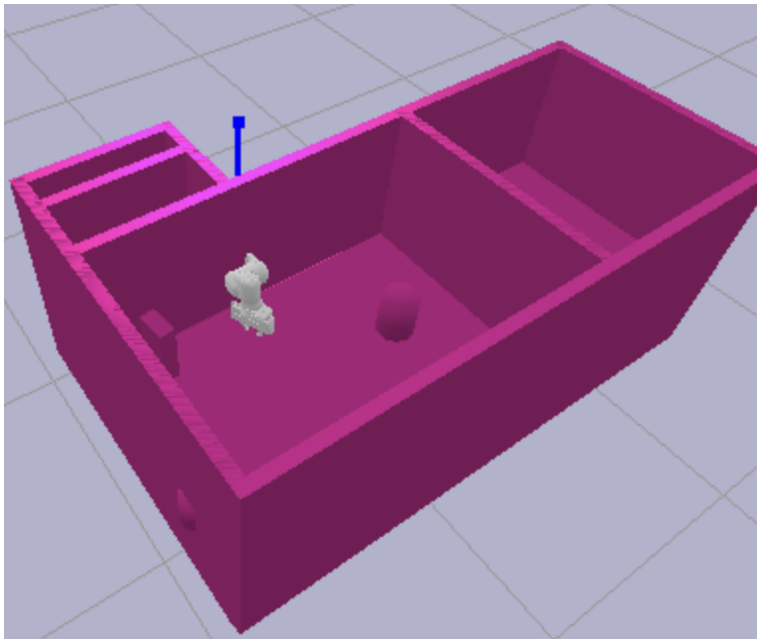


Creating Custom dataset (Part 2)

Generated sensible occlusions around the perimeter of the robot's workspace and within it:

Dataset Samples:

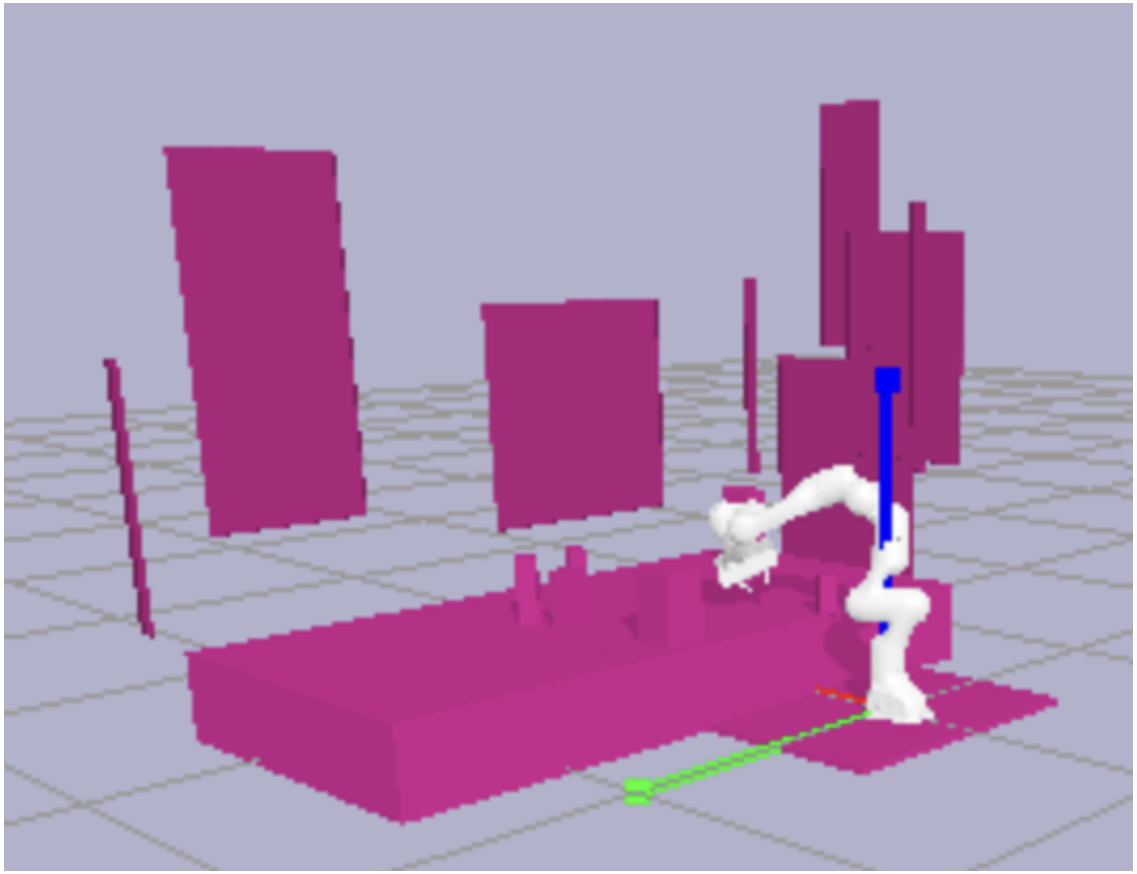




Creating Custom dataset (Part 3)

Generate occlusions around perimeters/borders, representing obstacles (i.e. a human leaning over)

Dataset Samples:



Training Comparison

Receive identical training curves with occlusion vs no occlusions

Training Comparison

Receive identical training curves with occlusion vs no occlusions

Problem: Collision loss computes the pointcloud with occlusions

Outline

Future Steps

Near Future ($<$ 1 Week)

- Regenerate the full dataset using VAMP
- Modify loss function to compute collision loss with a full, occlusion-free scene

Far Future ($>$ 4 Week)

- Add constraints for the camera-holding arm by treating it as additional obstacle points.
- Continuously resample camera positions by synchronizing one manipulator's movement with the other's to maintain an optimal scene view.

Thank you!

Have a great rest of your Day + GL with finals!!!