Karlsruhe Institute of Technology

Autonomous Learning Robots Lab

Sim2Real Transfer Contact-Rich Assembly Tasks with Point Clouds

Praktikum Presentation – Weiheng Wang, Sheng Liu, Lucas Thomaz, Petr Novikov

Supervisors: Aleksandar Taranovic, Balázs Gyenes

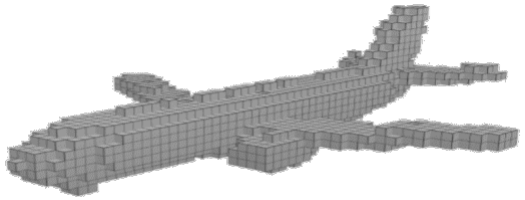
Background

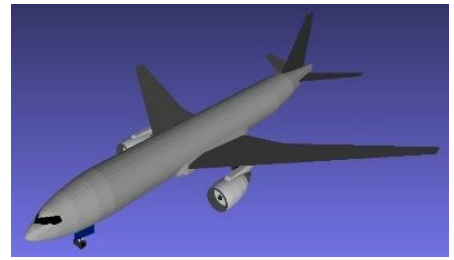
- Generalization Ability is Crucial for Robots to Perform Daily Tasks
- → **Design and train a vision-based policy**

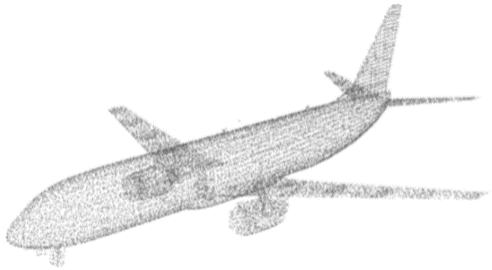
Background

- A good 3D visual representation is important
- → Point Cloud, Voxel Grid, RGB-D, Object Mesh.....

- Why not the others?
  - Voxel Grid: hard to obtain in the real world...
  - RGB-D: perspective bias, camera intrinsics and extrinsics...
  - Object Mesh: high computational cost
  - Final Choice: PC!







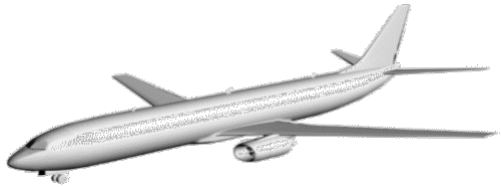


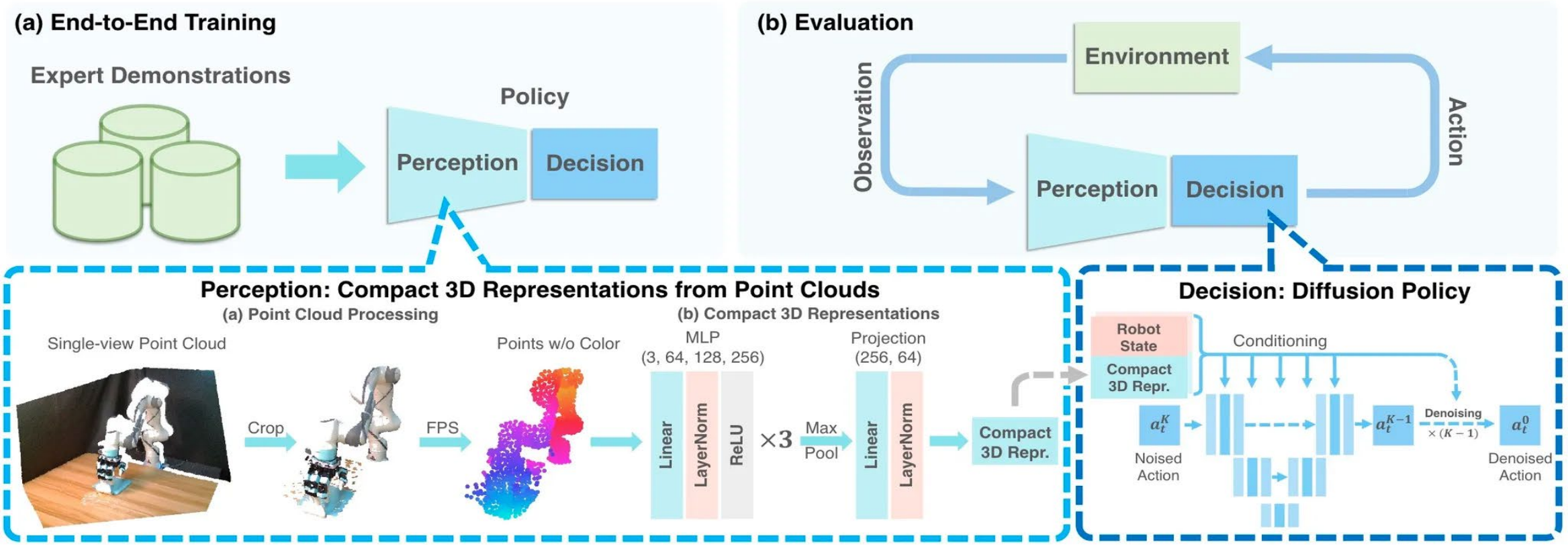
Image from Li, Y., Pirk, S., Su, H., Qi, C. R., & Guibas, L. J. (n.d.). FPNN: Field Probing Neural Networks for 3D Data; https://arxiv.org/abs/1605.06240

Background

- Also, large Amount of High-Quality Demonstration Trajectories
- In the real world, collecting demonstration data is costly, often leading to wear and tear on robotic arms, potentially causing damage.

- What we have done:
  - In a simulation environment
  - reduces operational safety risks for human operators

Method



- Perception
- Decision

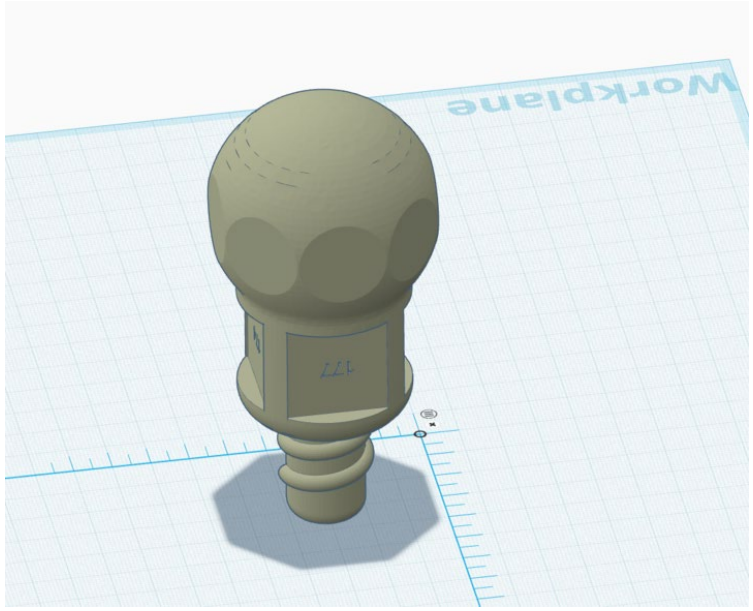
Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In Proceedings of Robotics: Science and Systems (RSS), 2024.

Experiment Setup

- Simulation Platform: IsaacLab
- Robot System: ALOHA
- Scene Setup: FurnitureBench-Lamp/Drawer
- Training: 1× NVIDIA A40 GPU

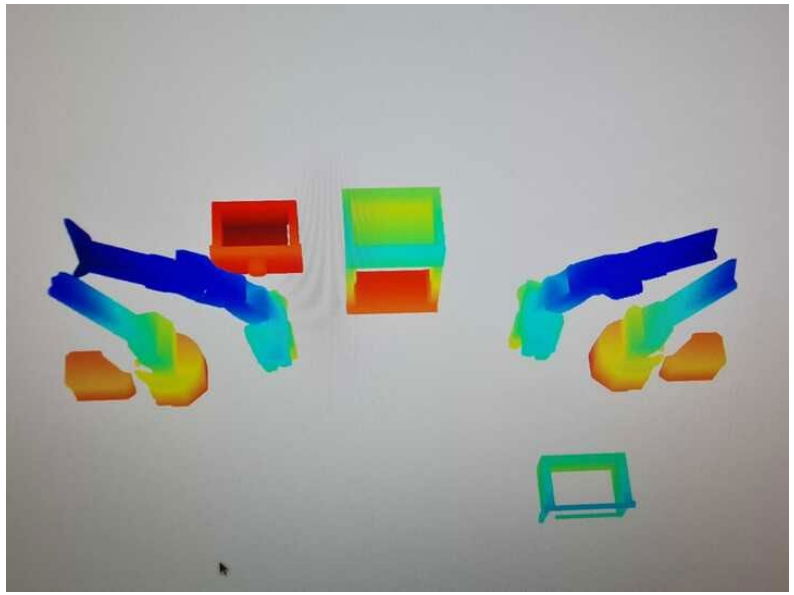
Collecting Demonstarions

- First, we tried the Lamp assemble task
  - Failed, for the drift between screw base and lamp holder
- Successfully placed the drawer box from the desk into the drawer cabinet
  - Totally collected 100 trajectories for the task



Collecting Demonstarions

- Transfer the trajectories into point cloud representation
  - Using RANSAC remove the desk
  - Sampling and removing colors of the point cloud to 1024×3
- Integrating Point Cloud, Action, and State using Zarr



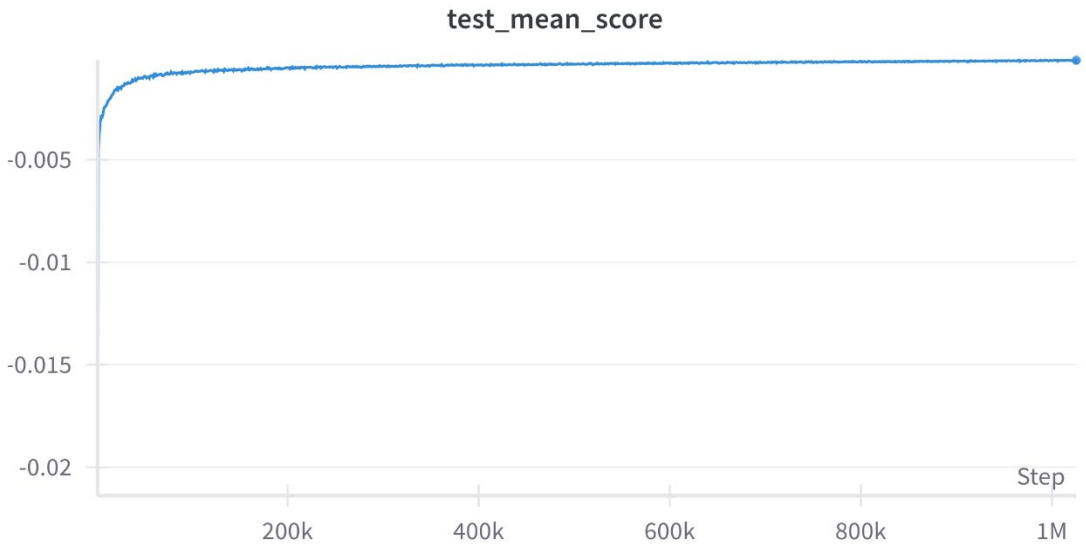
Collecting Demonstarions

- Transfer the trajectories into point cloud representation
  - Using RANSAC remove the desk
  - Sampling and removing colors of the point cloud to 1024×3
- Integrating Point Cloud, Action, and State using Zarr

```
(dp3) root@autodl-container-92e511a9ae-577bc8b4:~/autodl-tmp# python read_zarr.py
(10,)
Episode Ends: [  1363   3049   4841   6464   7729   9020  10285  11907  13478  15012
  16499  17703  18965  20356  22115  23909  25549  27001  28181  29703
  31185  32365  33779  35317  36533  37915  39307  41032  42313  43864
  45249  47030  48285  50692  51727  53269  54434  55871  57394  59353
  60719  61924  63238  65078  66683  67893  69221  70672  72053  74073
  75644  76981  79078  80535  82326  84179  85727  86822  89132  90470
  91771  93387  94710  96096  97729  99046 100595 101761 103035 104697
 106191 107711 109294 110964 112208 113868 115099 116397 117825 119097
 120496 122514 124392 125990 128090 129457]
Shape of episode_ends: (86,)
/
 ├── data
 │   ├── action (129457, 7) float32
 │   ├── point_cloud (129457, 1024, 3) float64
 │   └── state (129457, 8) float32
 └── meta
     └── episode_ends (86,) int64
```

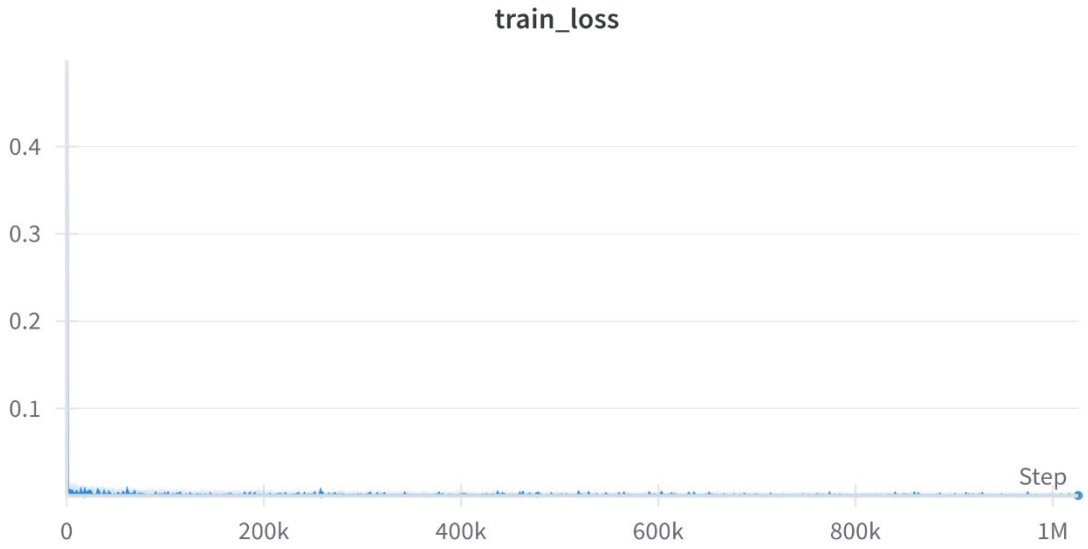
Training and Evaluation

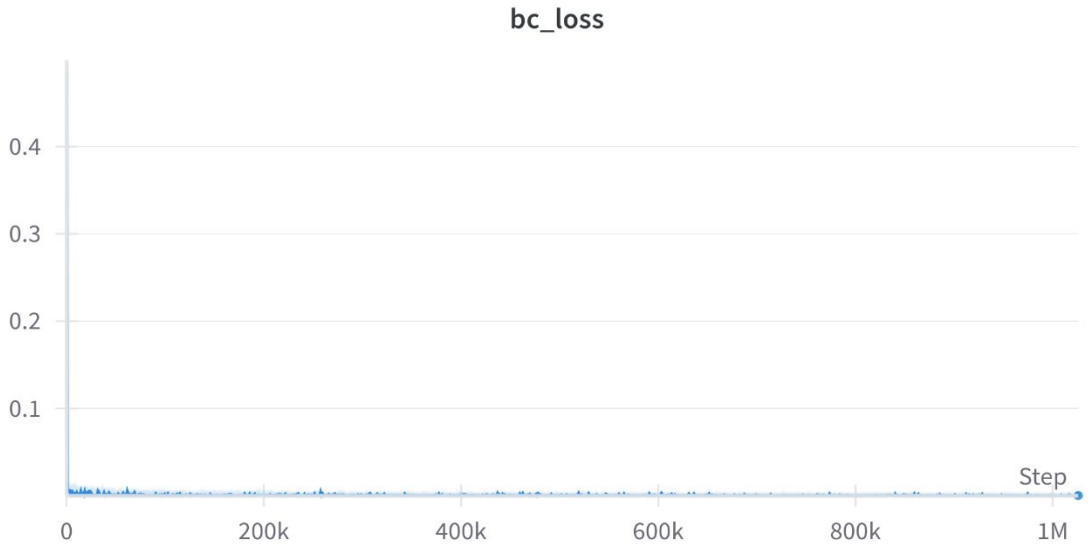
- Writting wrapper for Aloha_Drawer task
- Training 1000 epoch, 24 A40 GPU-hours
- Get the checkpoint for policy



Training and Evaluation

- Writting wrapper for Aloha_Drawer task
- Training 1000 epoch, 24 A40 GPU-hours
- Get the checkpoint for policy





Failed Visualization

- We tried to use Isaaclab play the new policy
- But it occurs the mismatched dimension of matrix problem
  - Need to check the structure of encodes and checkpoints

```
  File "/home/shengliu/.local/share/ov/pkg/isaac-sim-4.2.0/exts/omni.isaac.ml_archive/pip_prebundle/torch/nn/modules/module.py", line 1562, in _c
all_impl
    return forward_call(*args, **kwargs)
  File "/home/shengliu/.local/share/ov/pkg/isaac-sim-4.2.0/exts/omni.isaac.ml_archive/pip_prebundle/torch/nn/modules/container.py", line 219, in
forward
    input = module(input)
  File "/home/shengliu/.local/share/ov/pkg/isaac-sim-4.2.0/exts/omni.isaac.ml_archive/pip_prebundle/torch/nn/modules/module.py", line 1553, in _w
rapped_call_impl
    return self._call_impl(*args, **kwargs)
  File "/home/shengliu/.local/share/ov/pkg/isaac-sim-4.2.0/exts/omni.isaac.ml_archive/pip_prebundle/torch/nn/modules/module.py", line 1562, in _c
all_impl
    return forward_call(*args, **kwargs)
  File "/home/shengliu/.local/share/ov/pkg/isaac-sim-4.2.0/exts/omni.isaac.ml_archive/pip_prebundle/torch/nn/modules/linear.py", line 117, in for
ward
    return F.linear(input, self.weight, self.bias)
RuntimeError: mat1 and mat2 shapes cannot be multiplied (1x384 and 512x1024)

Set the environment variable HYDRA_FULL_ERROR=1 for a complete stack trace.
2025-03-18 20:06:23 [16,376ms] [Warning] [omni.usd] Unexpected reference count of 4 for UsdStage 'anon:0x26da7400:World0.usd' while being closed
in UsdContext (this may indicate it is still resident in memory).
[INFO]: Simulation is stopped. Shutting down the app.
```