Towards Efficient Assembly Part Motion Planning for Robotic Assembly

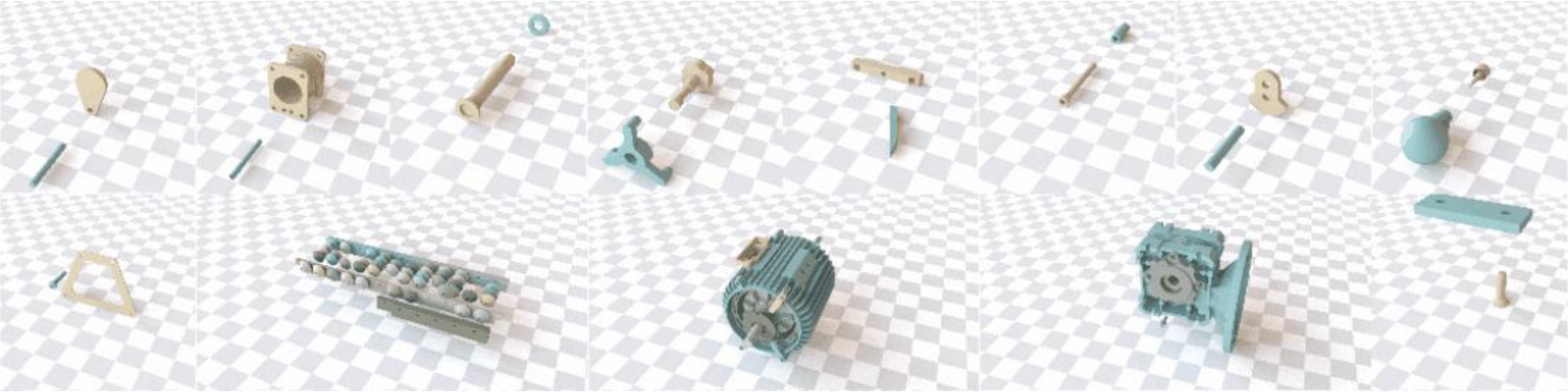
Team 5
Jung Geumyoung
Kim Jaemin

KAIST

Contents

Introduction

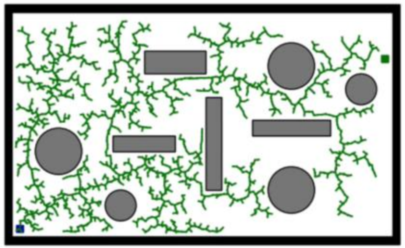
Assembly part motion planning : process of determining collision-free path of a part to move and join individual components into final assembled product



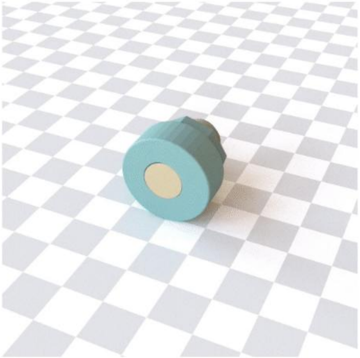
Challenges

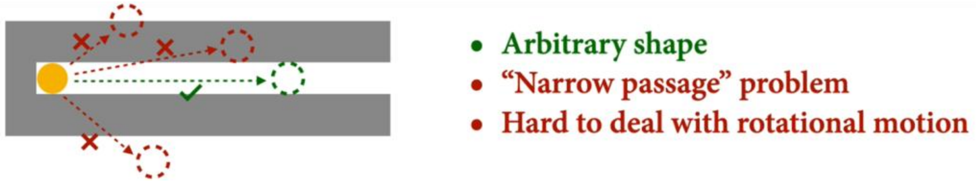
1. Large search space (6 DoF 3D poses), narrow passage problem

2. Assembly must be conducted under physical constraints

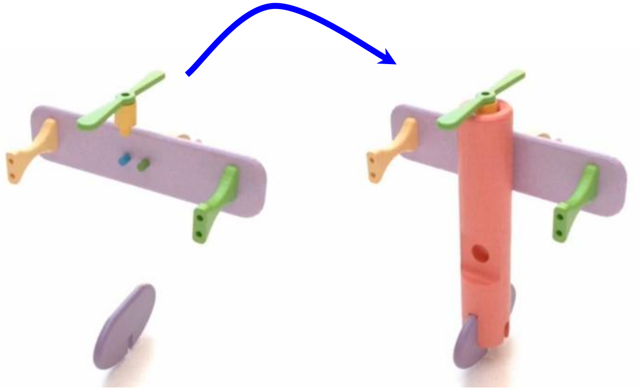


RRT [LaValle 1998]





assembly feasible?

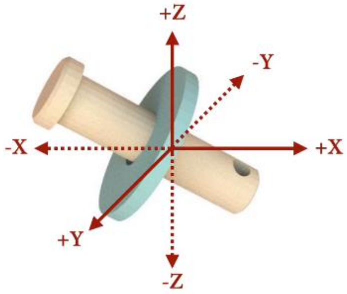


Baseline : ASAP

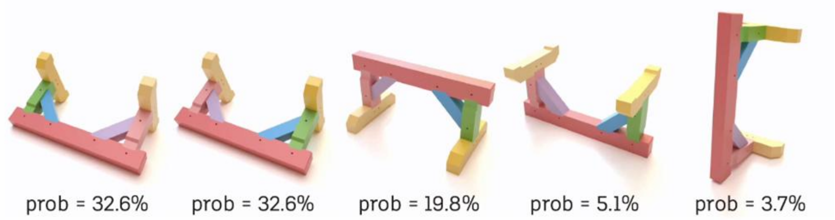
Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility

Offers : robotic assembly planning pipeline

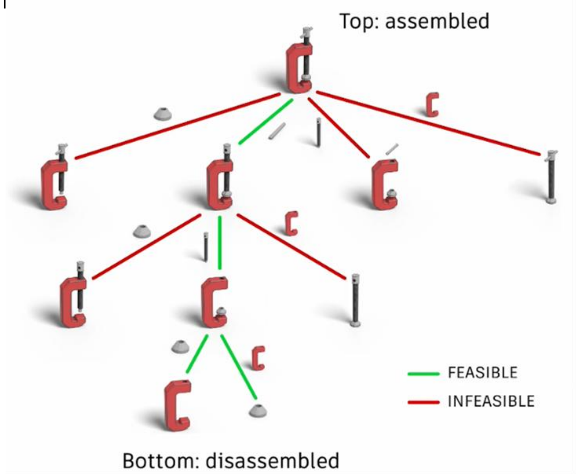
1. Part motion planning : 6 Discrete-action sampling
2. Physical stability check : Physics-based simulation + quasi-static pose
3. Assembly sequence planning : Disassembly tree search



6 actions



quasi-static pose



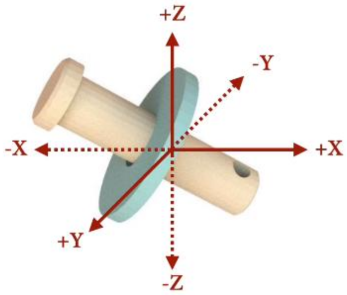
disassembly tree

KAIST

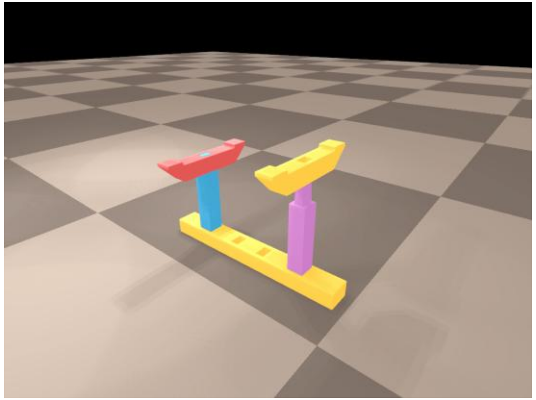
Baseline : ASAP

1. Part motion planning

- Samples discretized actions (forces) in 6 directions
- Apply the actions in physics-based simulation, check disassembly success



6 actions

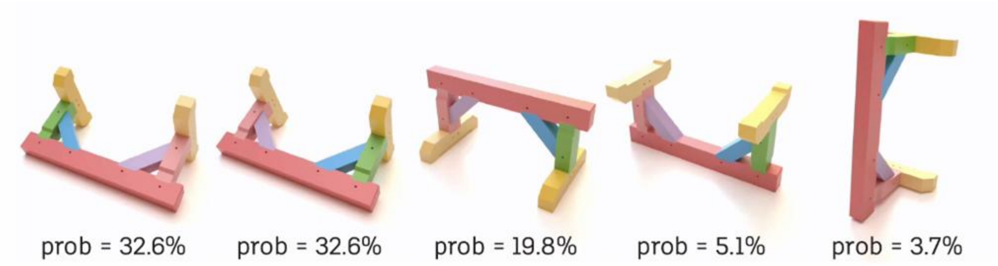


KAIST

Baseline : ASAP

2. Physical stability check

- Quasi-static pose estimation + physics-based simulation

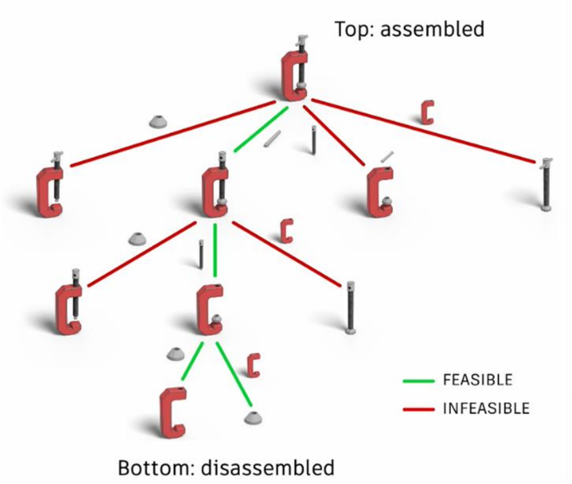


Quasi-static poses

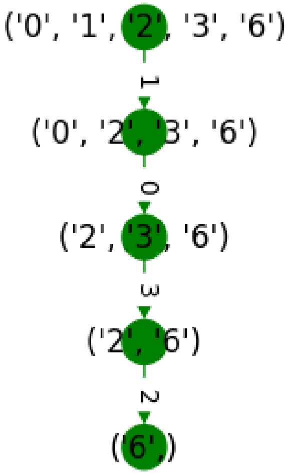
Baseline : ASAP

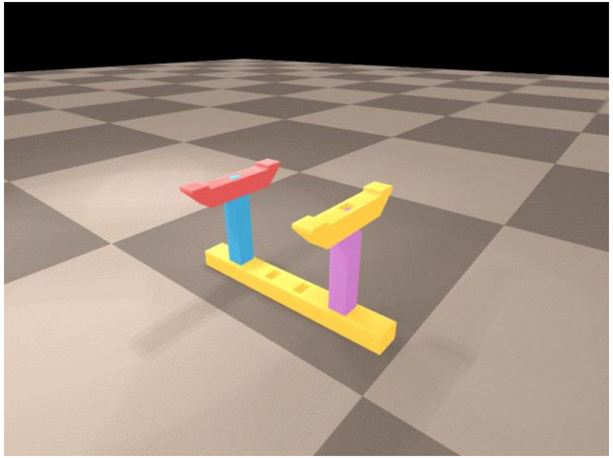
3. Assembly sequence planning

- Disassembly tree search



Disassembly tree

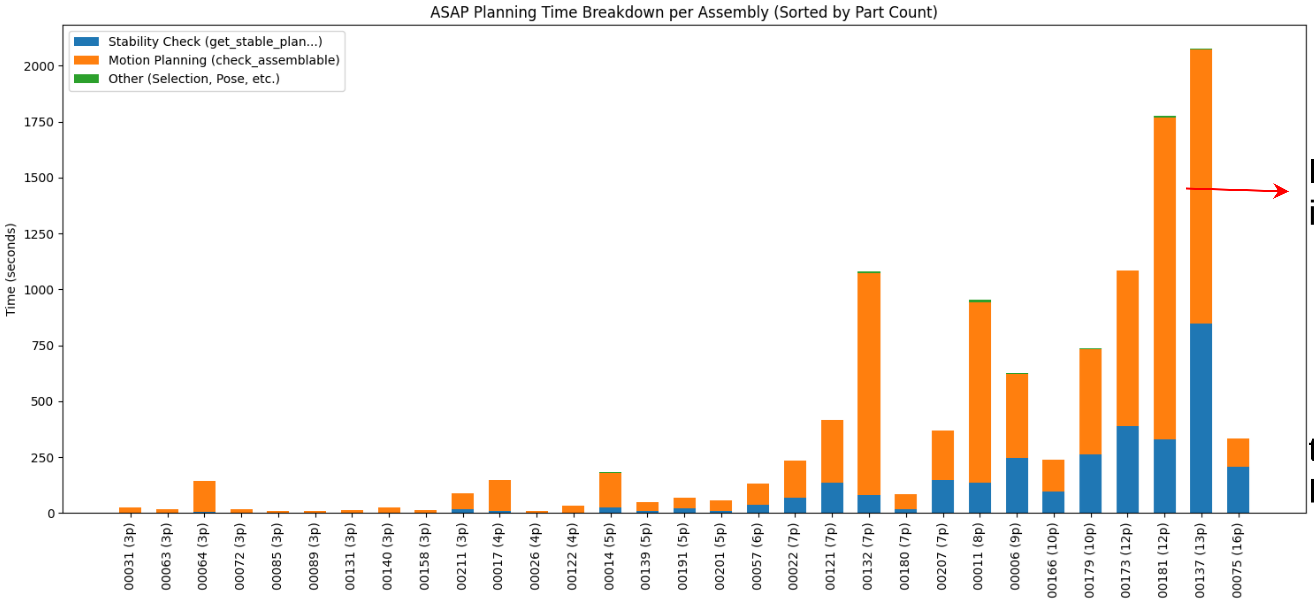




Limitation 1 | Running Time

Physics-based simulation has **high computation costs**

Main functions: check_assemblable (**motion planning**), get_stable_plan (**stability check)**



Part motion planning (in orange) is the most time-consuming task

thought : can't we do better than BFS with 6 discrete actions?

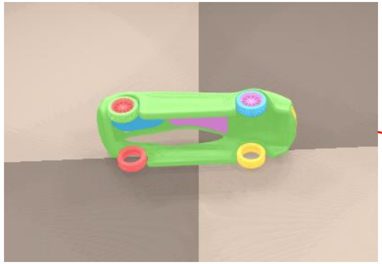
KAIST

Limitation 2 | Planning Accuracy

Disassembly tree fully expanded, but **no solutions found**

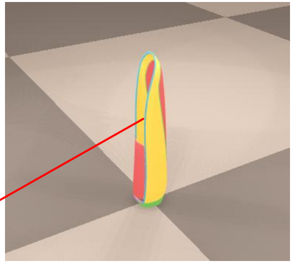
Observed frequent failures on assemblies with **round objects**

Possible factor: **limited simulation accuracy**, needs more inspection



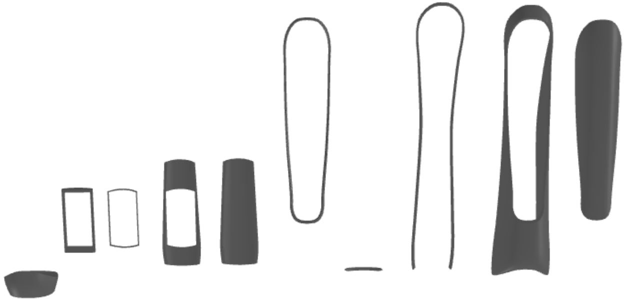
full assembly

arbitrary shape collapses during
physical simulation (not stable)



full-assembly

separate parts



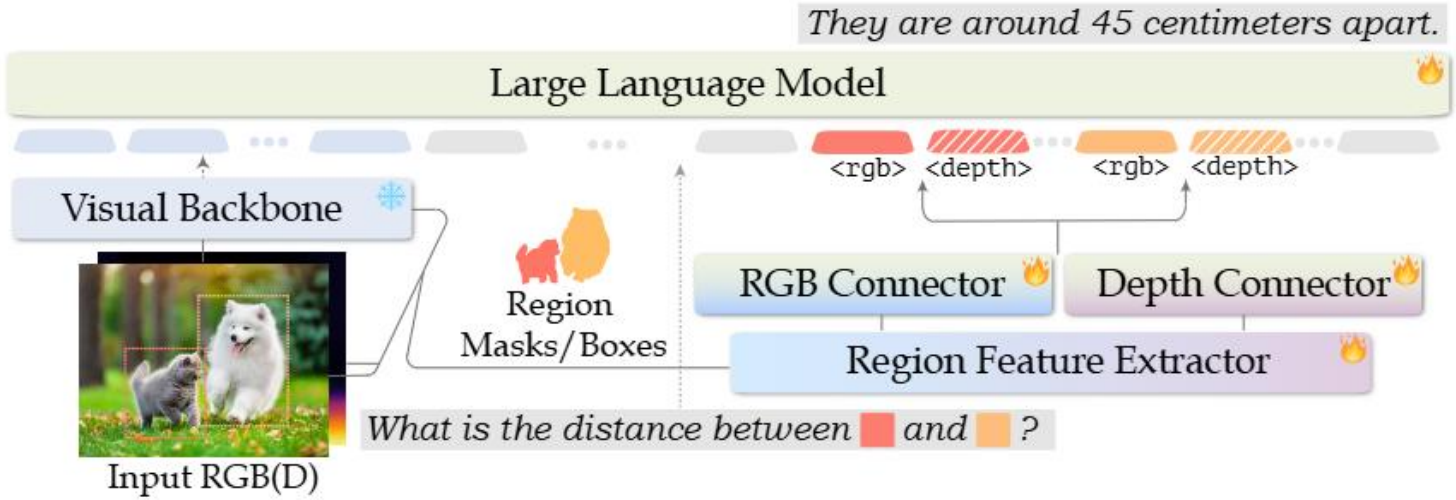
separate parts

KAIST

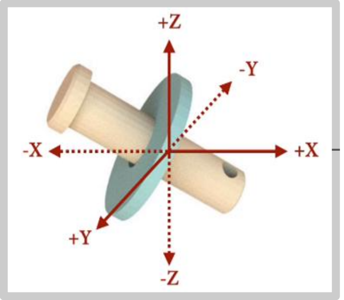
Limitation 1 | Possible Approach

Utilize **Spatial reasoning capable FM** to guess initial disassembly force direction

Reduce the number of physics-based action queries to **speed up the motion planning**



Action Prediction Using FMs | System Overview



[Input]
Picture of an assembly

[Input]
text prompt

In 3D space,
give me **action vector** to
pull **blue part** out of **beige part**

Spatial Reasoning Foundation Model

[Output]
predicted action vector

[x, y, z] = [0.7, 0, -0.7]

Prioritize actions
closer to prediction

physics-based simulator

Try [1, 0, 0], [0, 0, -1]

Action Prediction Using FMs | Example

considering in 2D x, y space, give me the unit vector that pulls the blue part out of the beige one.

To determine the unit vector that pulls the **blue part (disc)** out of the **beige shaft**, we observe the 2D projection of the image you uploaded. The assembly is shown at an **angle**, with the blue disc intersecting and encircling the beige shaft.

From visual inspection:

- The **beige shaft** is oriented diagonally from **top-left** to **bottom-right**.
- The **blue disc** is centered around the shaft, suggesting the removal direction is **perpendicular to the shaft's axis**.

So to **pull the blue part out**, we want the direction **along the shaft axis**, pointing away from the center of the disc and out through either side of the shaft.

Let's define the shaft's 2D orientation in image space:

- Shaft axis: from **top-left to bottom-right** → a rough vector like `[-1, -1]`.

Therefore, the **unit vector that pulls the blue disc out** along this axis is:

$$\vec{v} = \frac{1}{\sqrt{2}}\begin{bmatrix}-1\\-1\end{bmatrix} = \begin{bmatrix}-0.7071\\-0.7071\end{bmatrix}$$

Action Prediction Using FMs | Example

Alternatively, if you want the disc to be pulled in the **opposite direction** (down-right):

$$\vec{v} = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$$

✅ **Final Answer (choose one depending on desired direction):**

- **Up-left removal:** `[-0.7071, -0.7071]`
- **Down-right removal:** `[0.7071, 0.7071]`

Would you like this vector expressed in a specific coordinate system (e.g., world frame vs. local part frame)?

SpatialRGPT: Grounded Spatial Reasoning in VLMs

- VLM trained with explicit spatial information
- 3D scene graphs built using depth estimation, camera calibration, ...

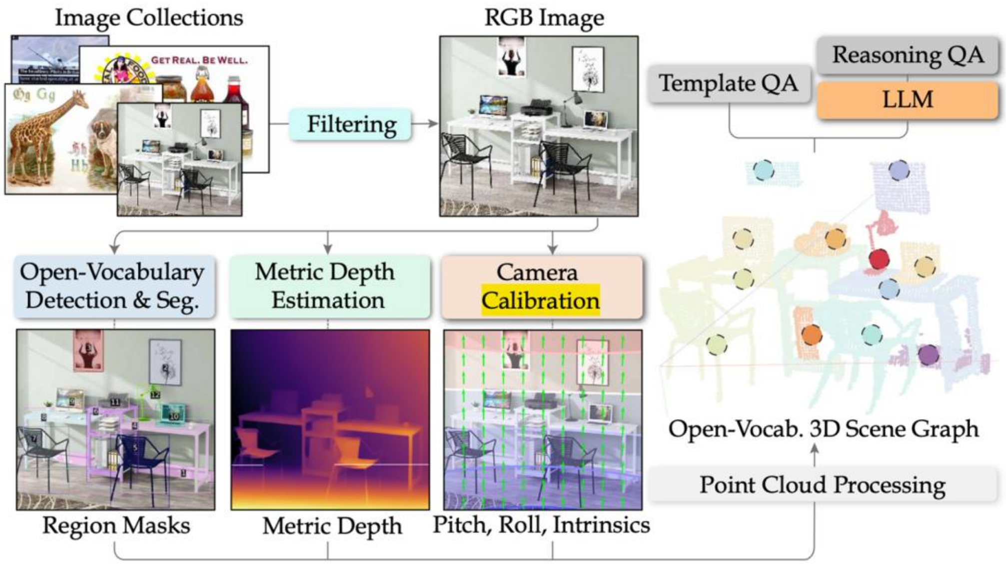
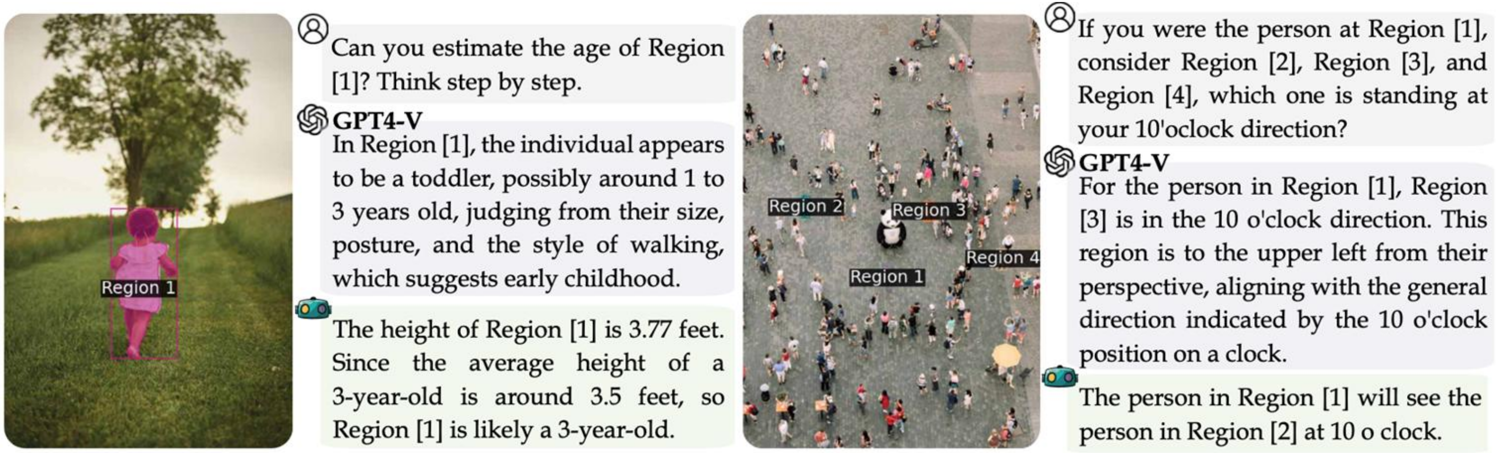


Figure 1: 3D scene graph construction via automatic data curation pipeline.

SpatialRGPT: Grounded Spatial Reasoning in VLMs

- Claims superior performance on spatial reasoning
- Candidate model for our disassembly motion estimation



Summary

- **Baseline**: assembly motion & sequence planning with physics-based simulation
- Naive action sampling for simulation is expensive
- **Idea**: use learning-based methods (e.g. foundation models) to leverage spatial understanding on disassembly motion planning
- **Desired result**: accelerated planning through efficient action sampling

Schedule & Roles

O : main

V : support

		Geumyoung Jung	Jaemin Kim
Have Done	Analyze previous work	O	O
	Test Reference code	O	O
To Do	VLM background research & prompt engineering	O	V
	Integrating VLM into pipeline	V	O
	Testing & collecting results	O	O
	Prepare final presentation	O	O

Thank you