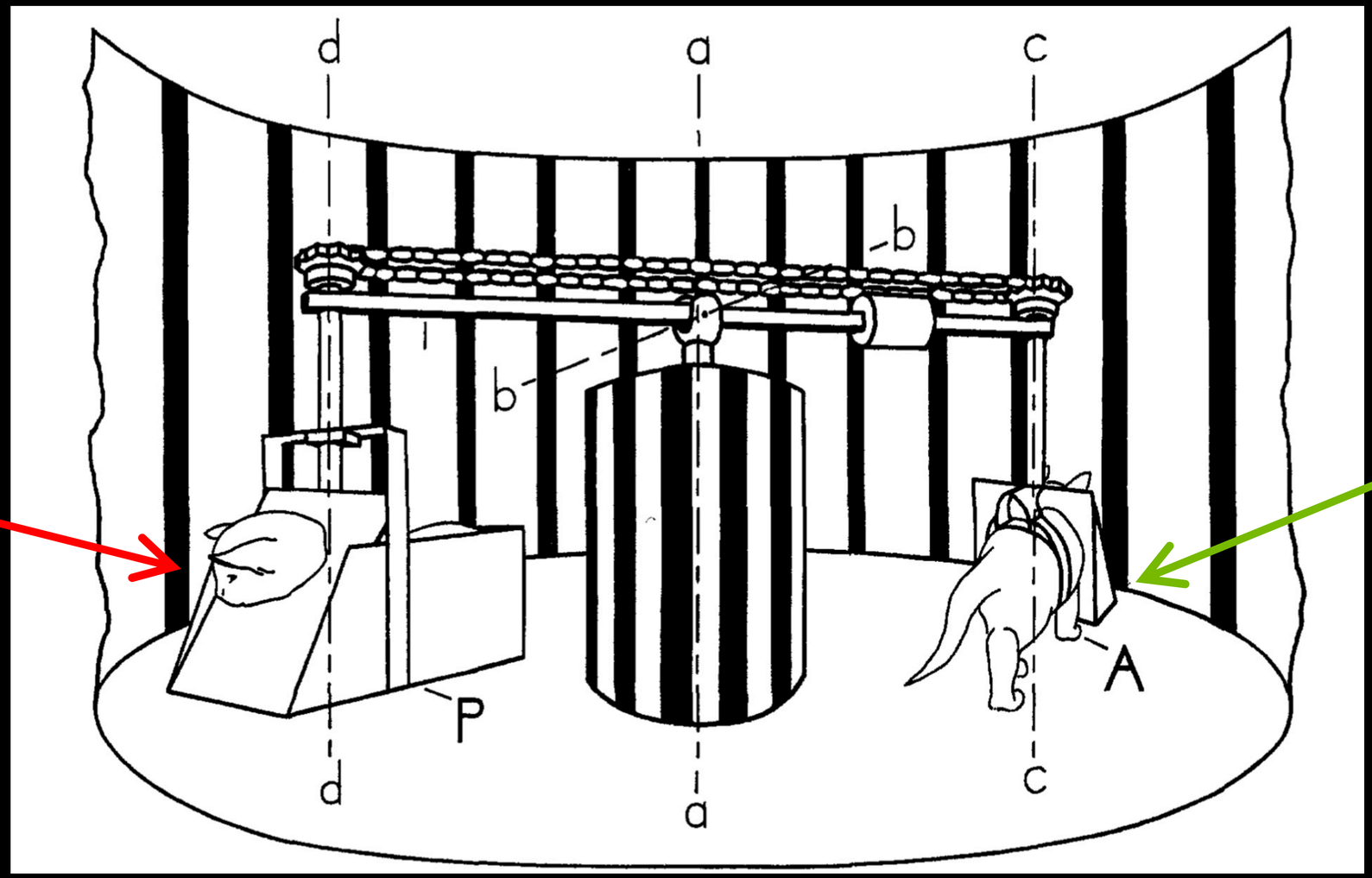
A Tale of Two Kittens

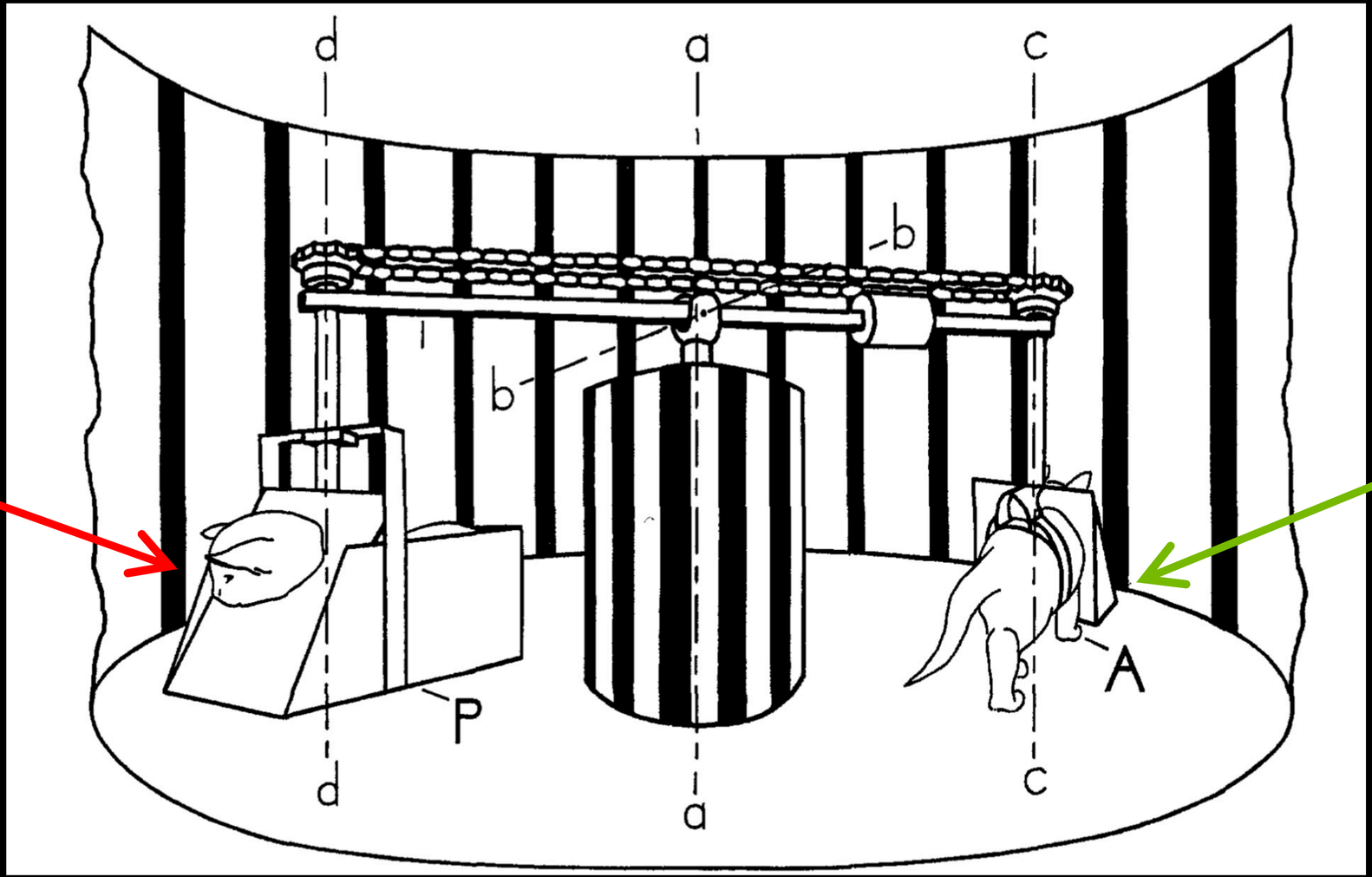
Passive Kitten



Active Kitten

Held & Hein, 1963.

ChatGPT

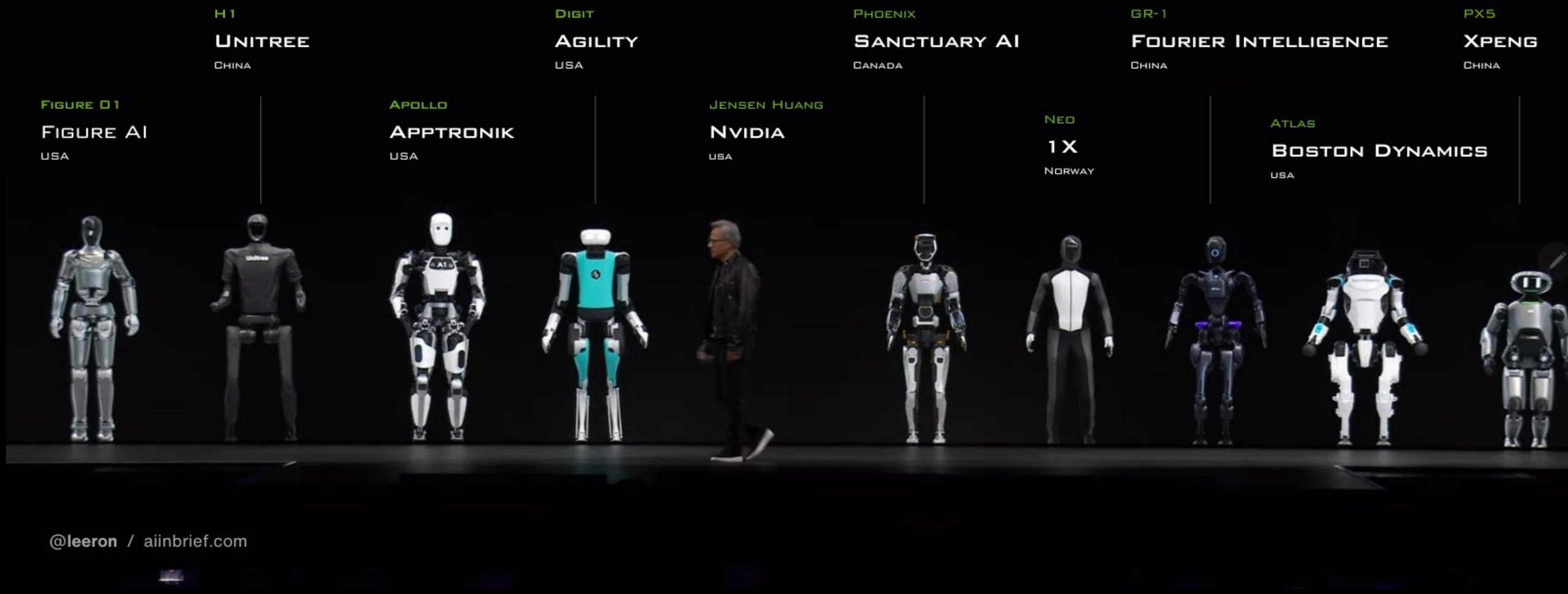


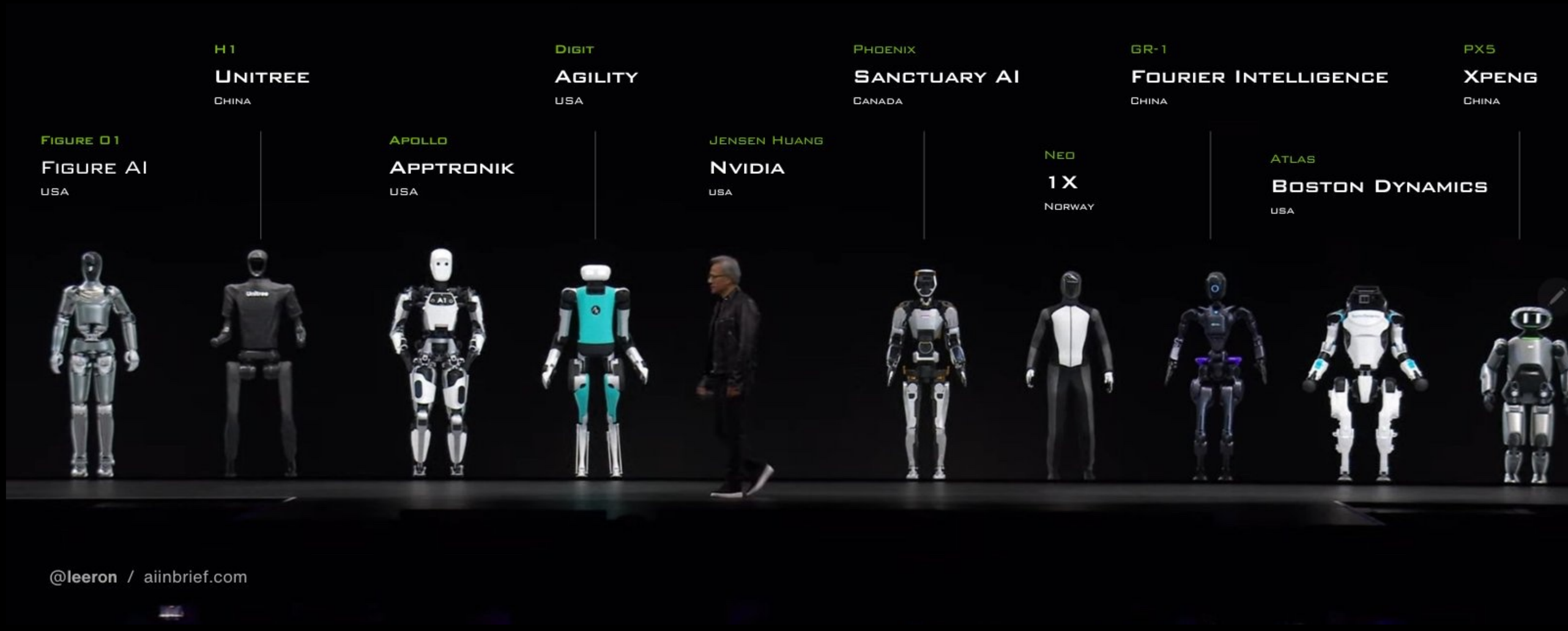
Embodied AI

Held & Hein, 1963.

Project GR00T

Dr. Jim Fan, NVIDIA Research


H1
Unitree
China
Digit
Agility
USA
Phoenix
Sanctuary AI
Canada
GR-1
Fourier Intelligence
China
PX5
Xpeng
China
Figure 01
Figure AI
USA
Apollo
Apptronik
USA
Jensen Huang
Nvidia
USA
Neo
1X
Norway
Atlas
Boston Dynamics
USA
@leeron / aiinbrief.com

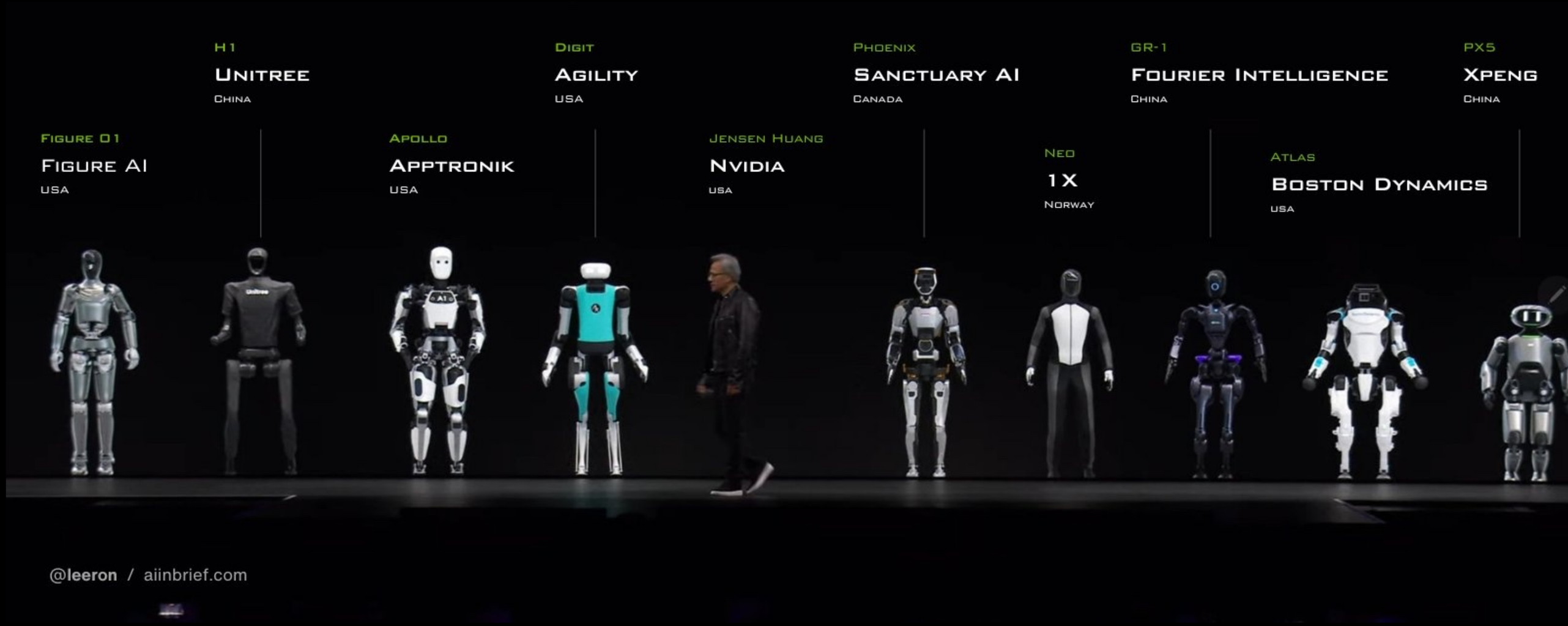


Why Humanoid?

BREAD

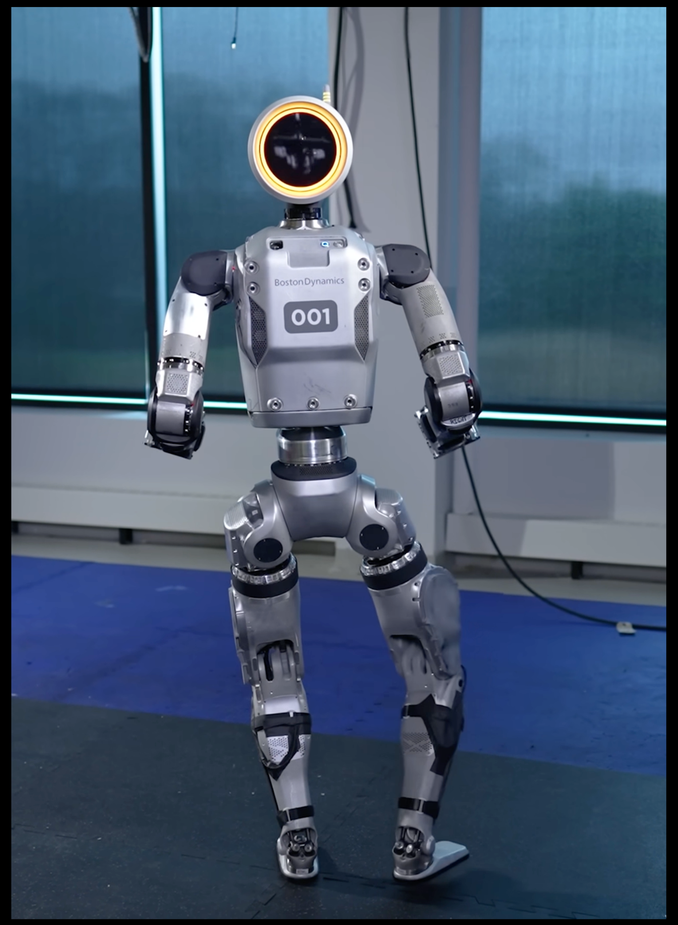
ADULT
stryker

BOAR'S HEAD
EST. 1905
PREMIUM MEATS & CHEESES
Mustard
WE ARE
OPEN
Order your Sandwiches or by the pound Meats & Cheeses for pickup
Boar's Head


H1
Unitree
China
Digit
Agility
USA
Phoenix
Sanctuary AI
Canada
GR-1
Fourier Intelligence
China
PX5
Xpeng
China
Figure 01
Figure AI
USA
Apollo
Apptronik
USA
Jensen Huang
Nvidia
USA
Neo
1X
Norway
Atlas
Boston Dynamics
USA
@leeron / aiinbrief.com

Why Now?

Tesla Optimus



Boston Dynamics e-Atlas

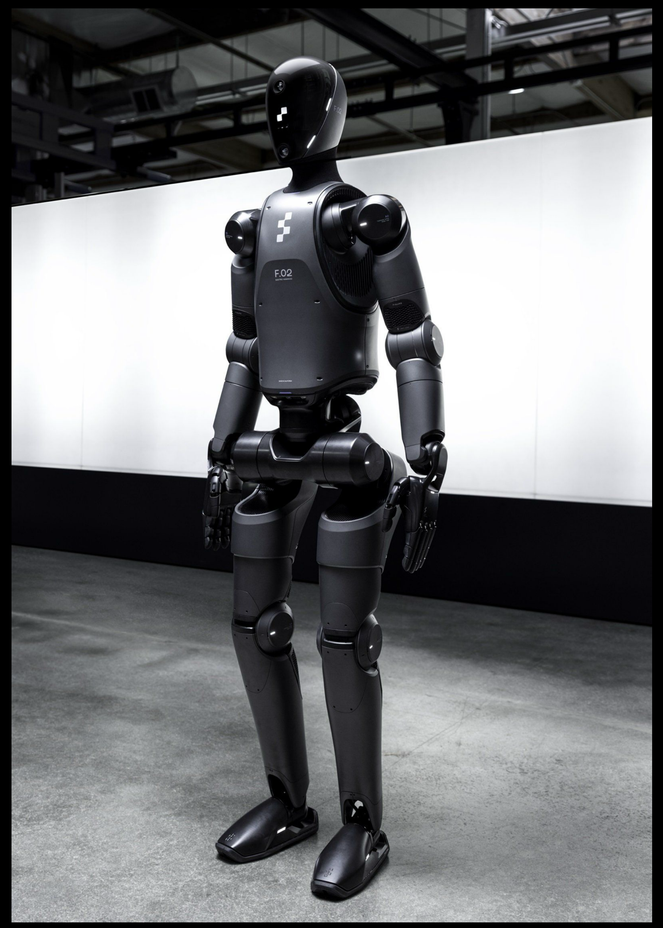
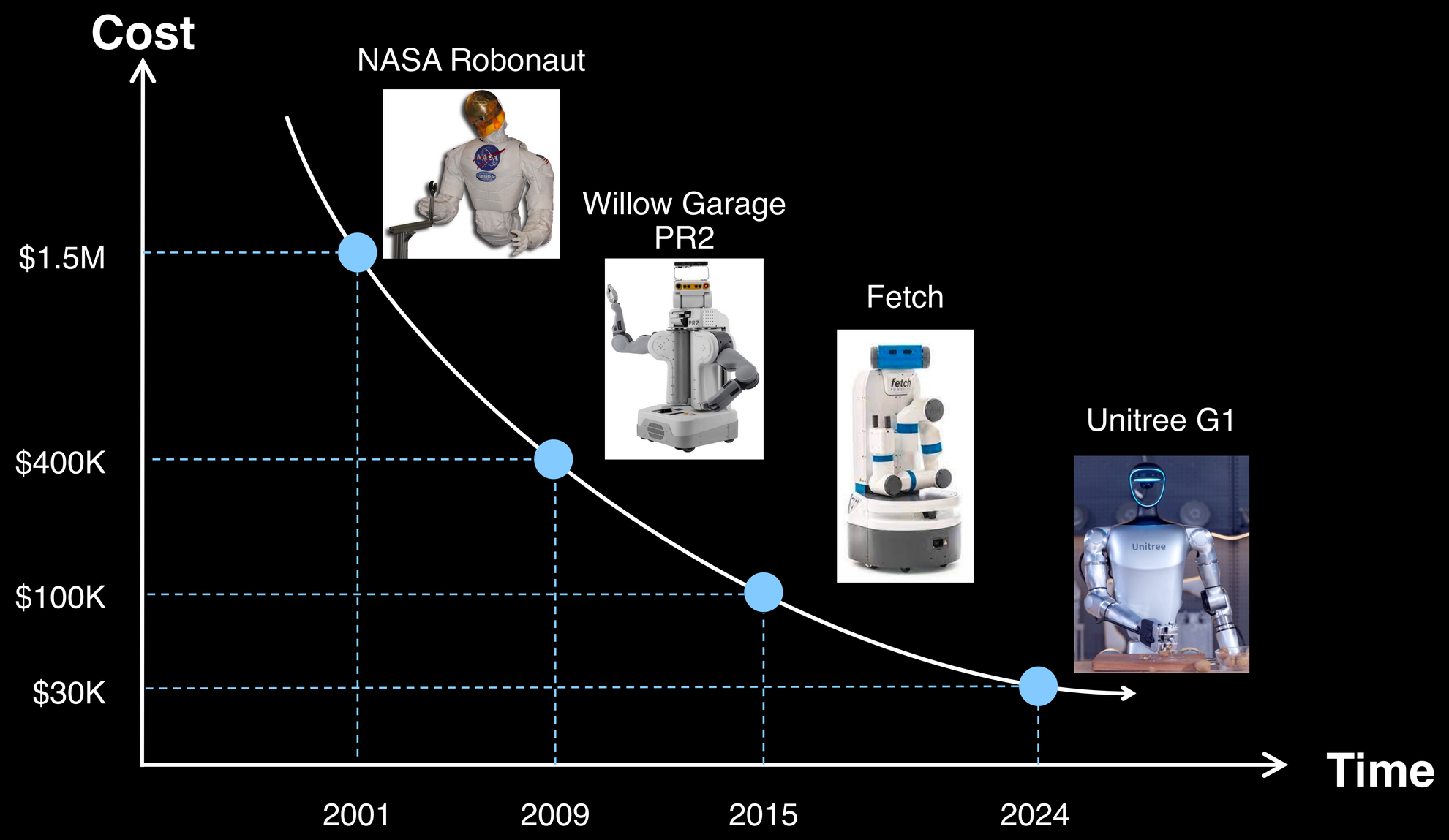
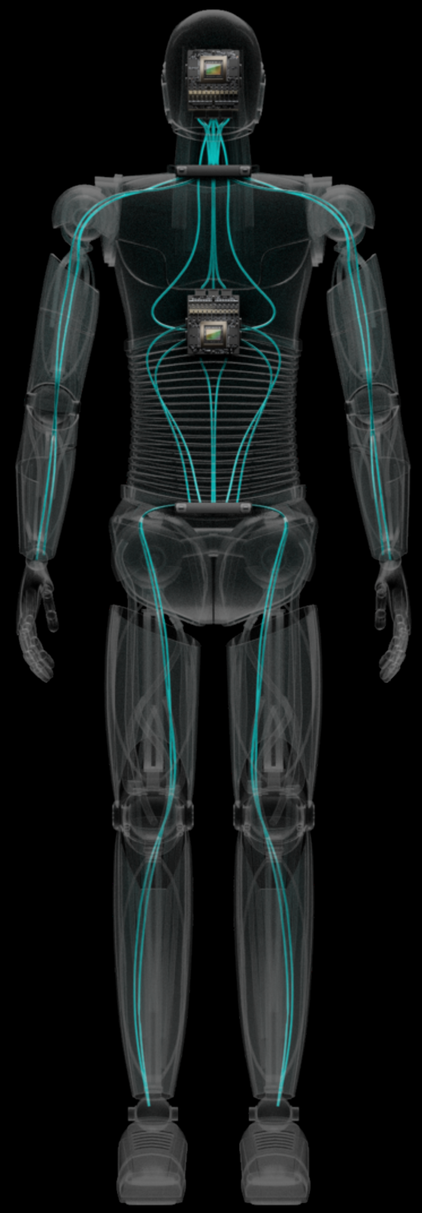


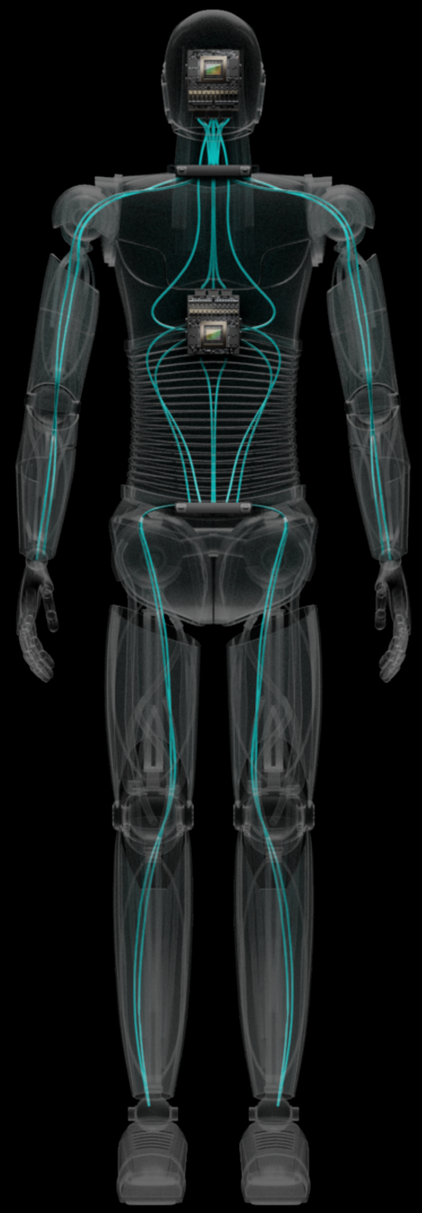
Figure F.02


Cost
NASA Robonaut
Willow Garage
PR2
Fetch
Unitree G1
$1.5M
$400K
$100K
$30K
Time
2001
2009
2015
2024



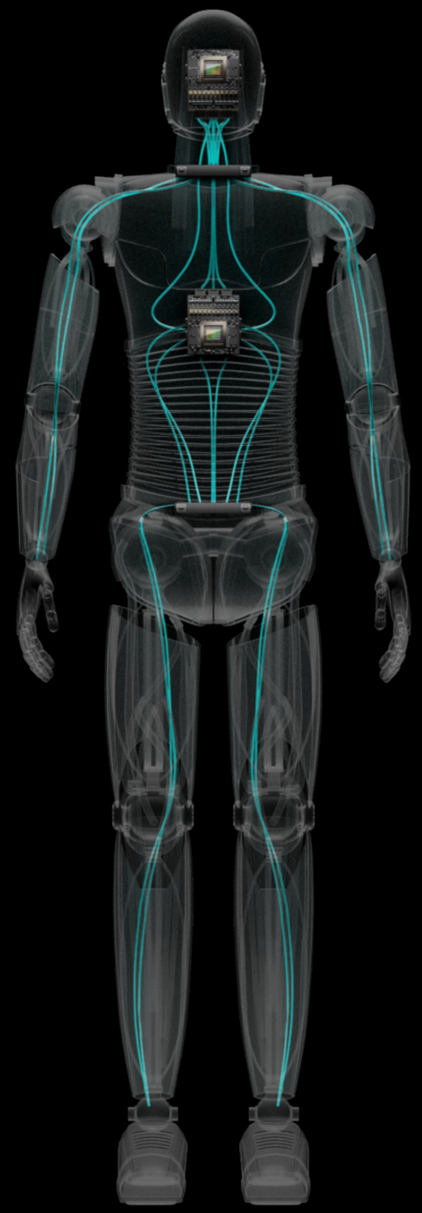
Project **GR00T**

AI Brain for Humanoid Robots



Project **GR00T**

Generalist Robot 0 0 Technology



Project **GR00T**

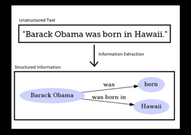
Generalist Robot 0 0 Technology

Specialist

Semantic Parsing

Sentiment Analysis

Summarization



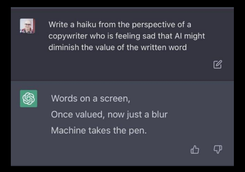
Information Retrieval

Generalist

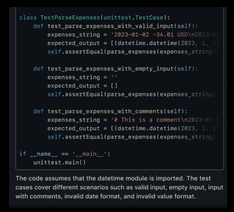
PaLM 2

ANTHROPIC CLAUDE 2

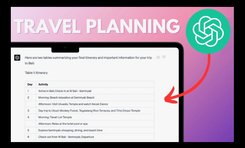
Specialized Generalist



Creative Writing



Coding

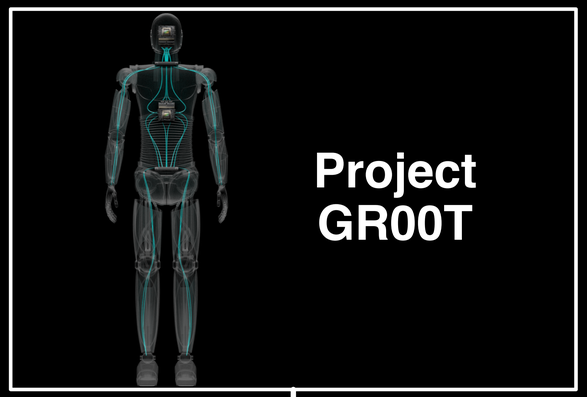


Travel Planning

Apple Intelligence

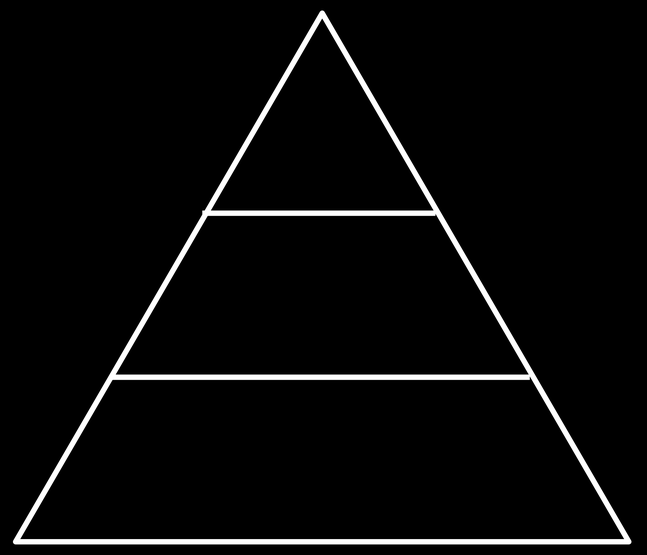
Specialized Generalist

Generalist

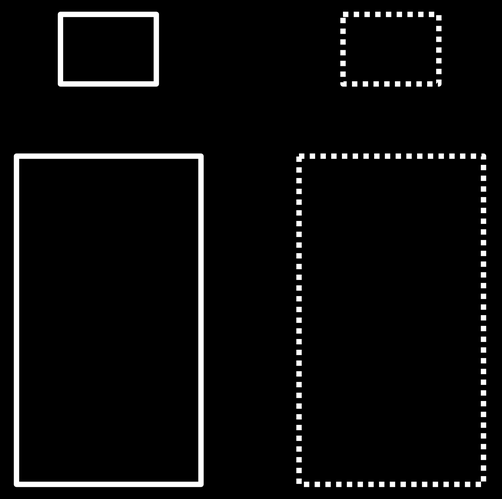


Specialist

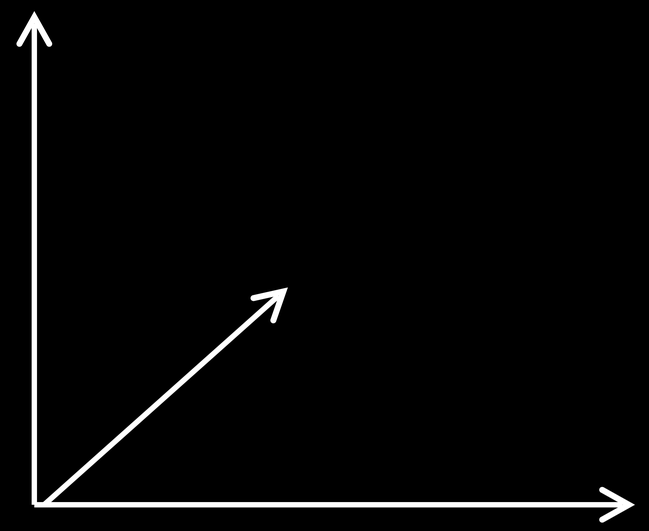
3 Principles



Data Pyramid

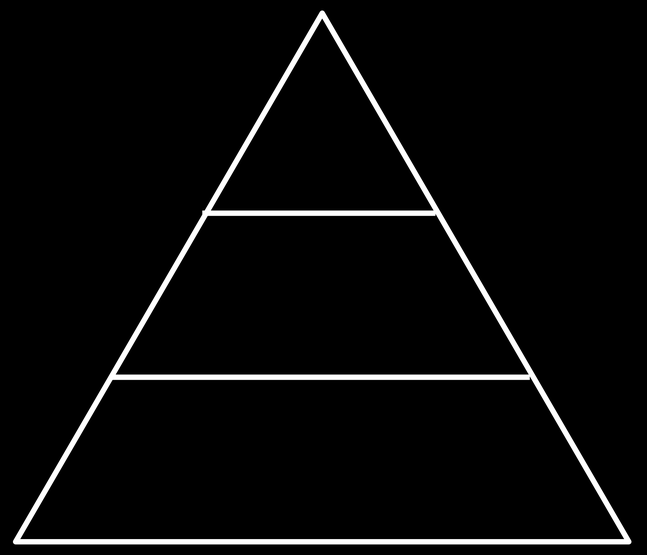


“The Matrix”

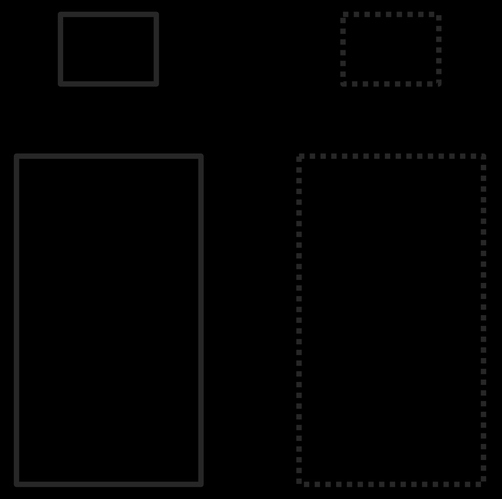


Foundation Agent

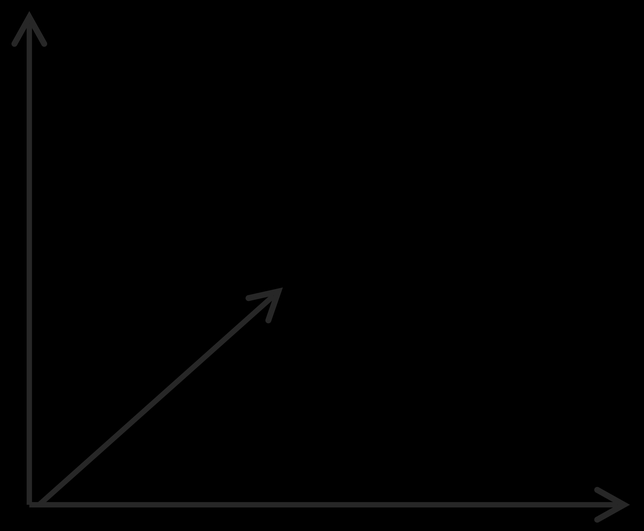
3 Principles



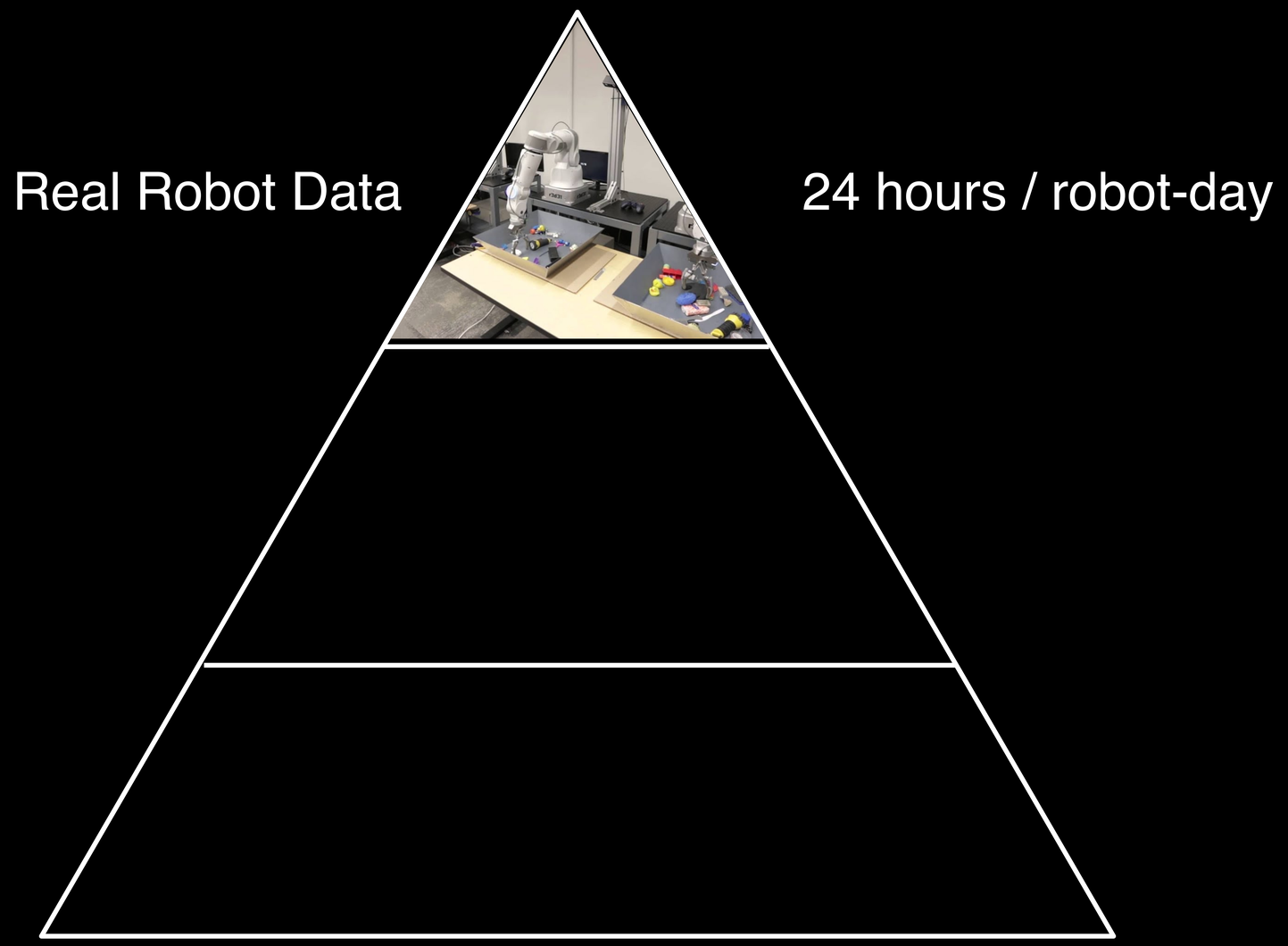
Data Pyramid



“The Matrix”



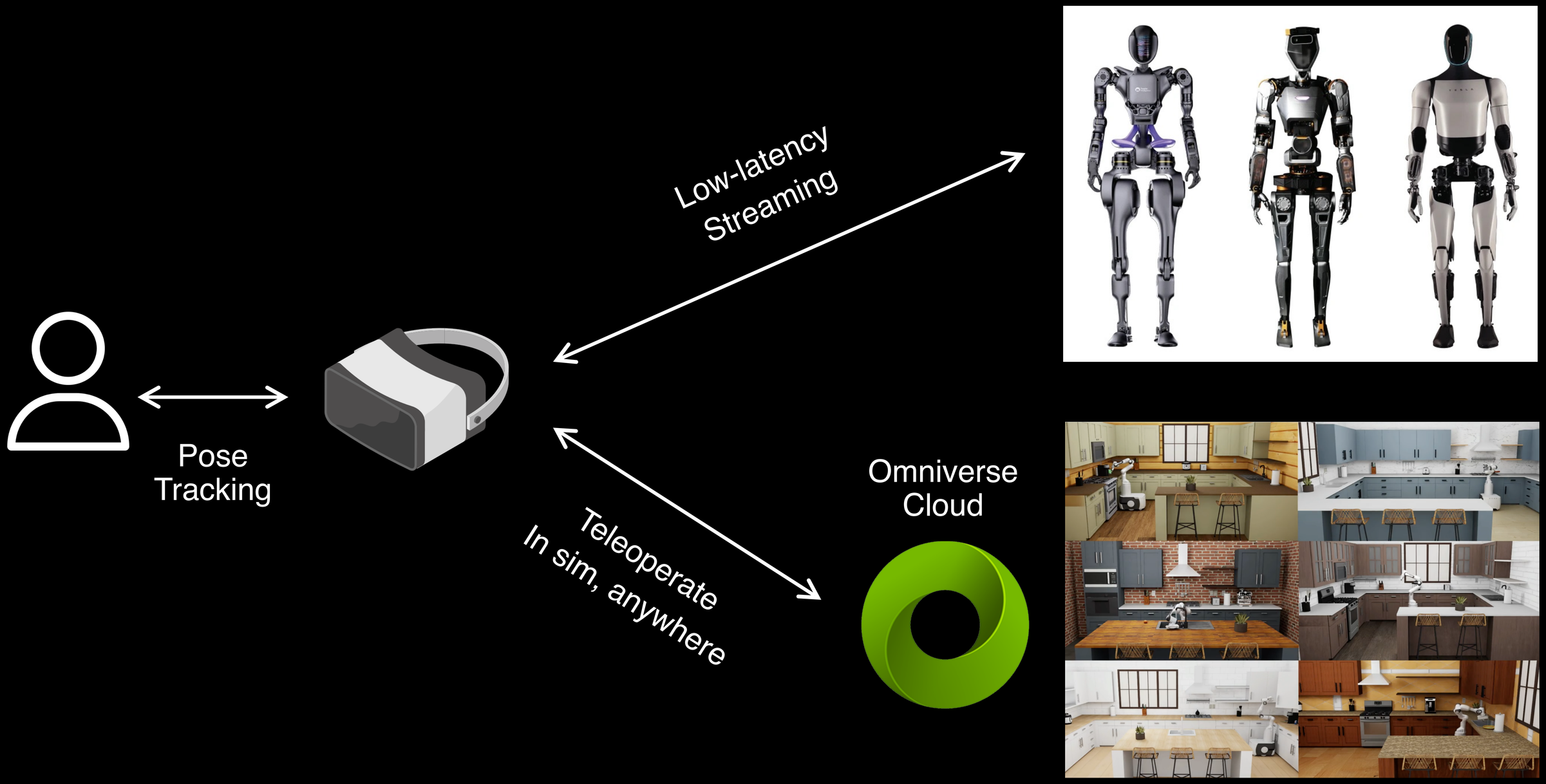
Foundation Agent

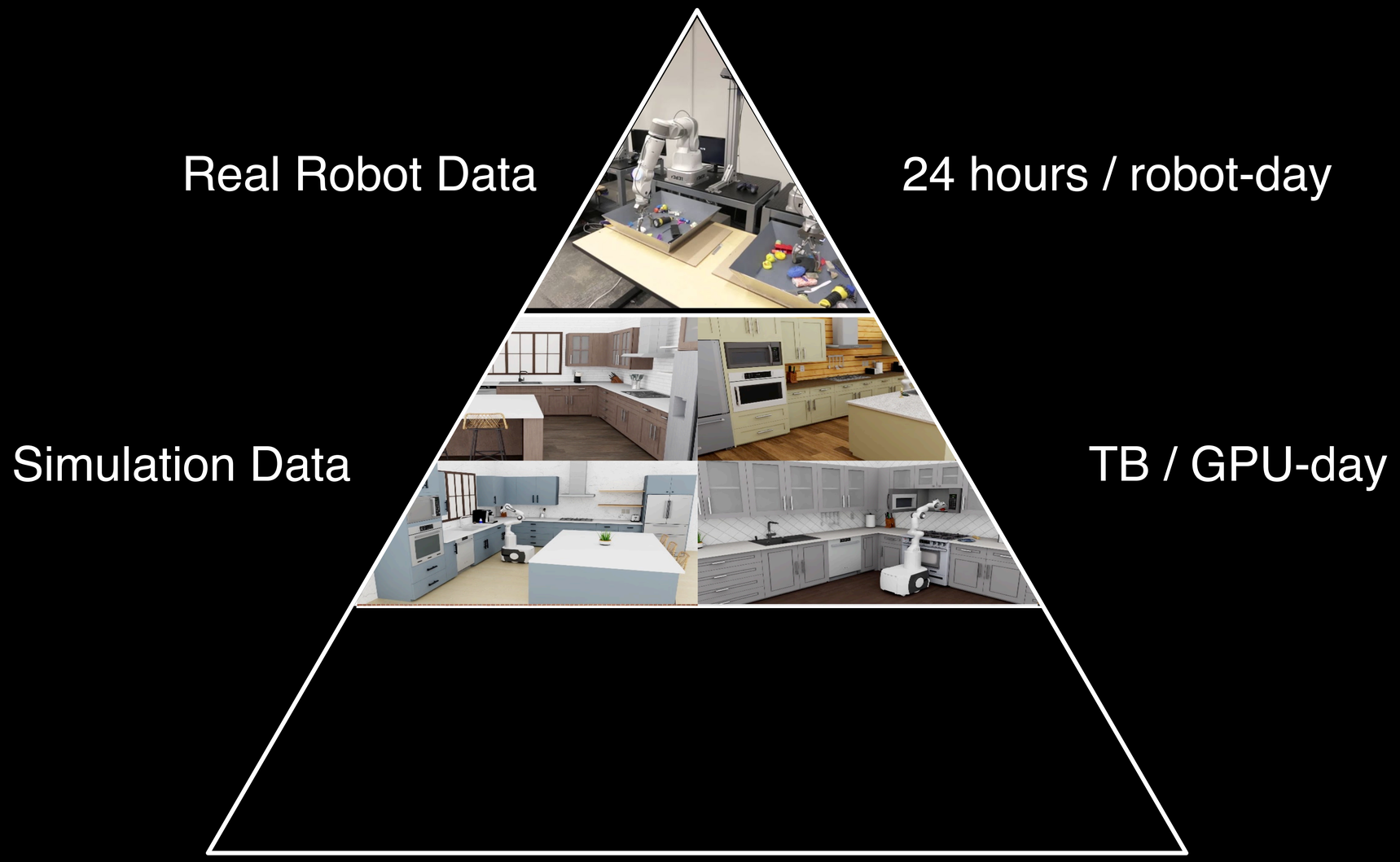


Data Pyramid

keep on

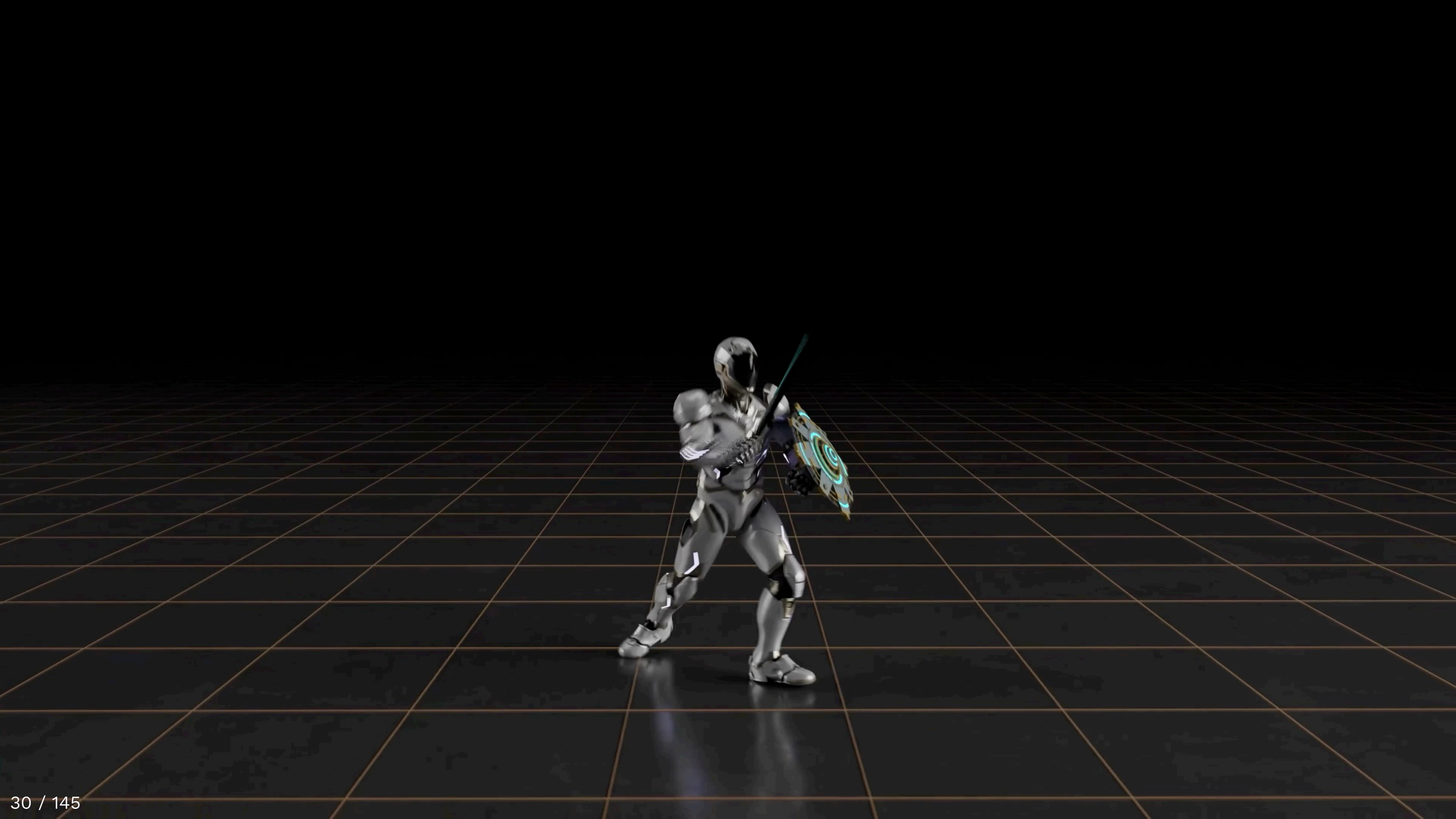
DELL
Visual Studio Code


Pose
Tracking
Low-latency
Streaming
Teleoperate
In sim, anywhere
Omniverse
Cloud



Data Pyramid

Isaac Lab



Omniverse

Real Robot Data

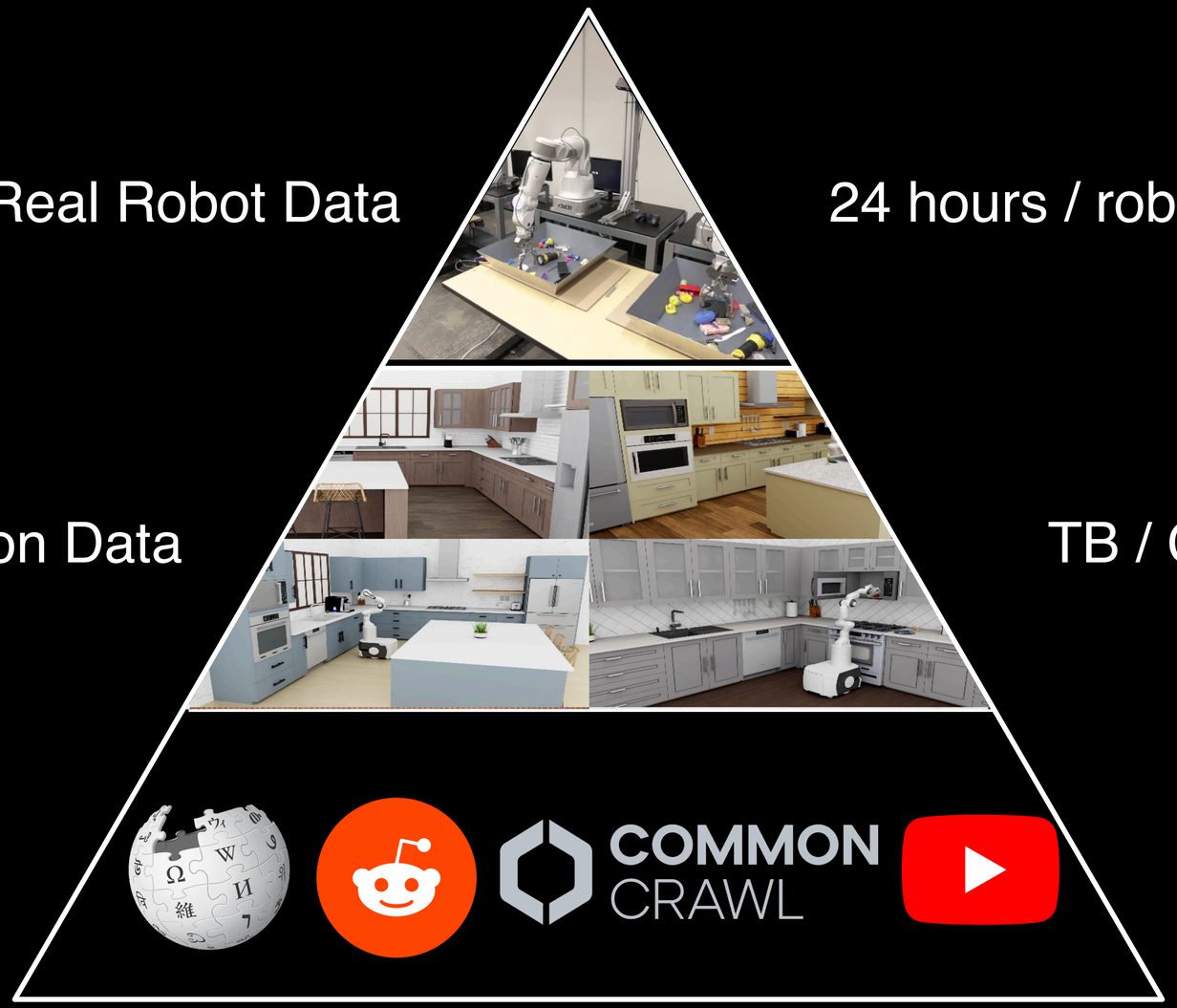
24 hours / robot-day

Simulation Data

TB / GPU-day

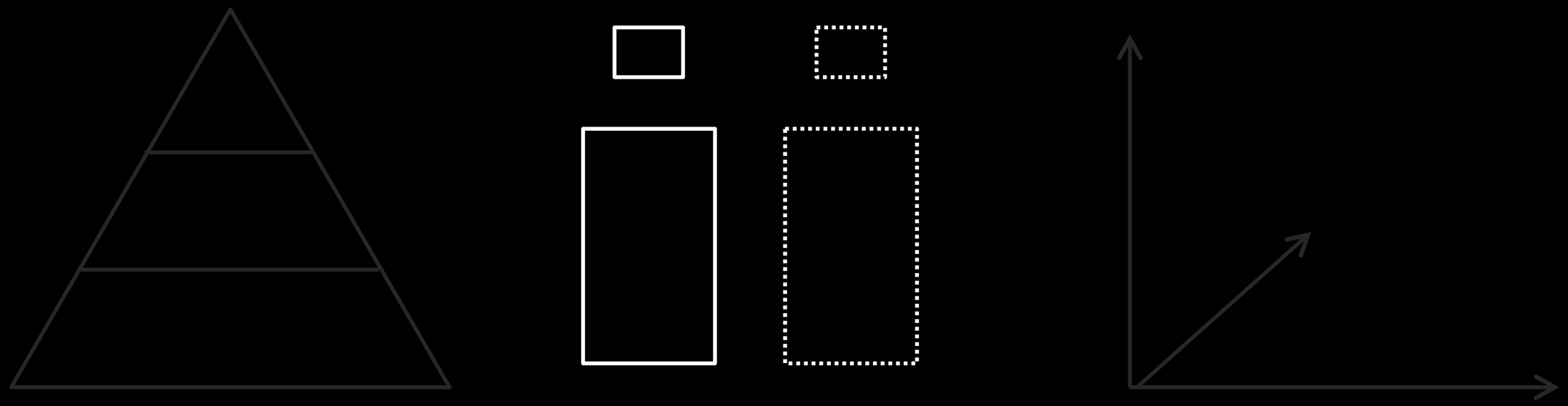
Internet Data

EB / day



Data Pyramid

Principle No. 2



Data Pyramid

“The Matrix”

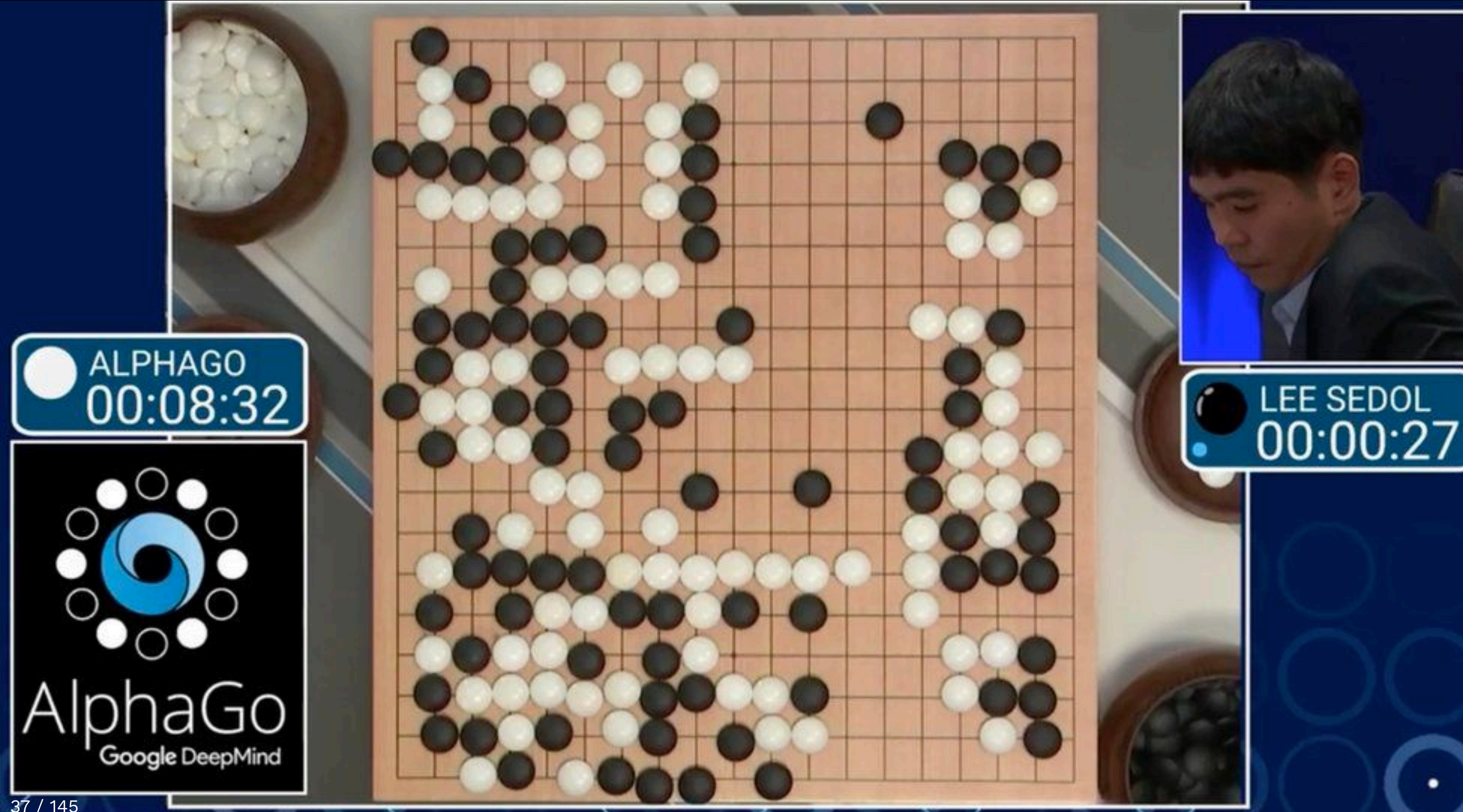
Foundation Agent

Do we live in a simulation?

Do robots live in a simulation?

Why does simulation work for training robots?

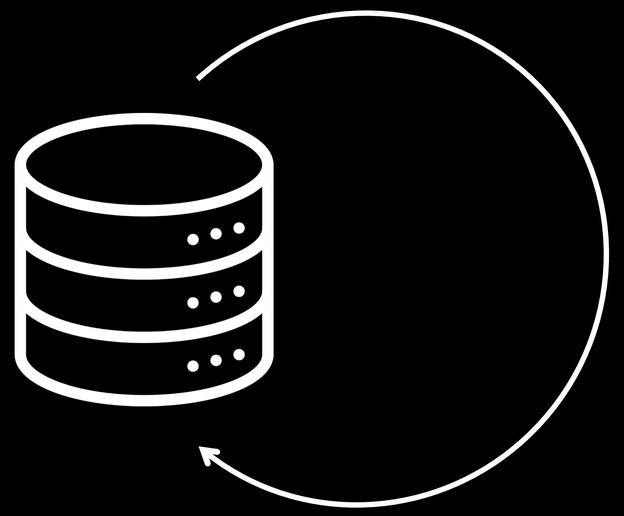
It’s easier to simulate a problem than to solve it.


ALPHAGO
00:08:32
AlphaGo
Google DeepMind
LEE SEDOL
00:00:27

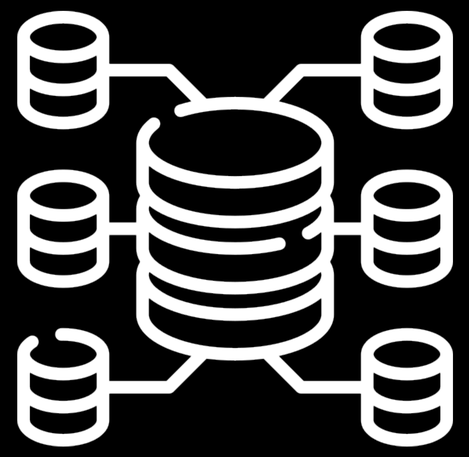
MINECRAFT

MINEDOJO

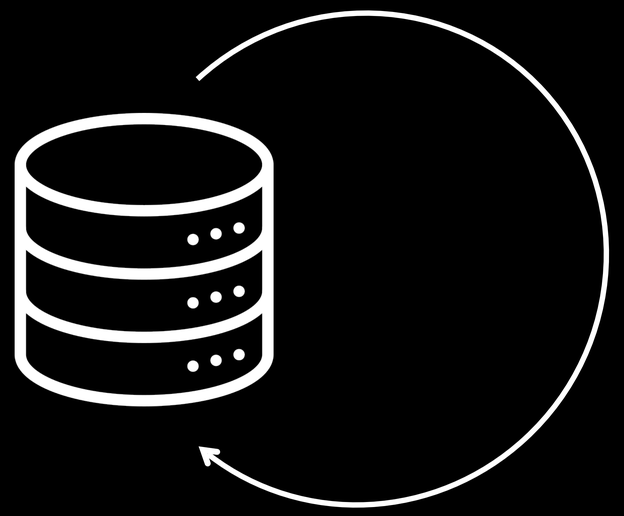
Framework for Generally Capable Agents



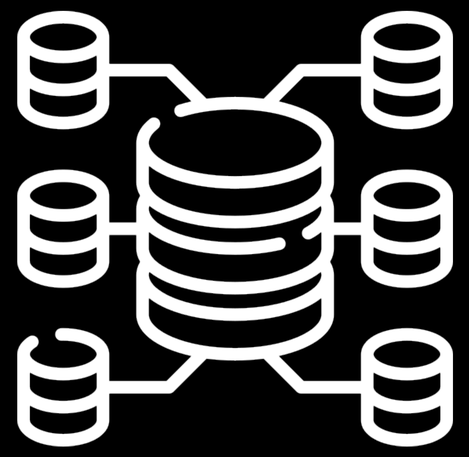
Self-Generated Data



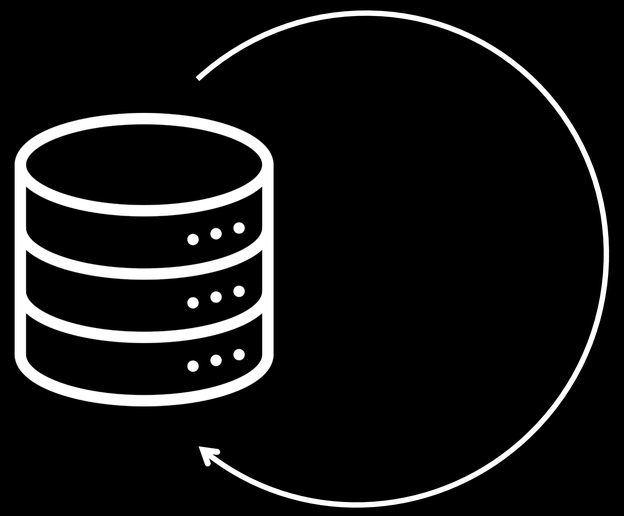
Multiply Data



Reinforcement Learning

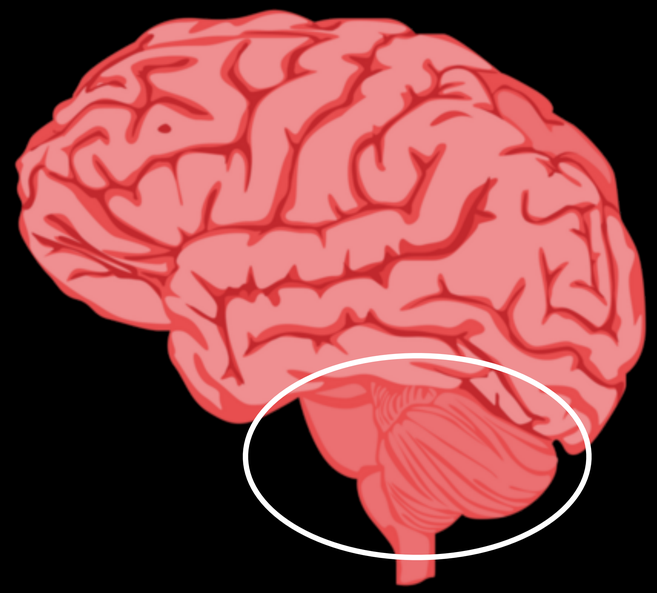


Imitation Learning



Reinforcement Learning

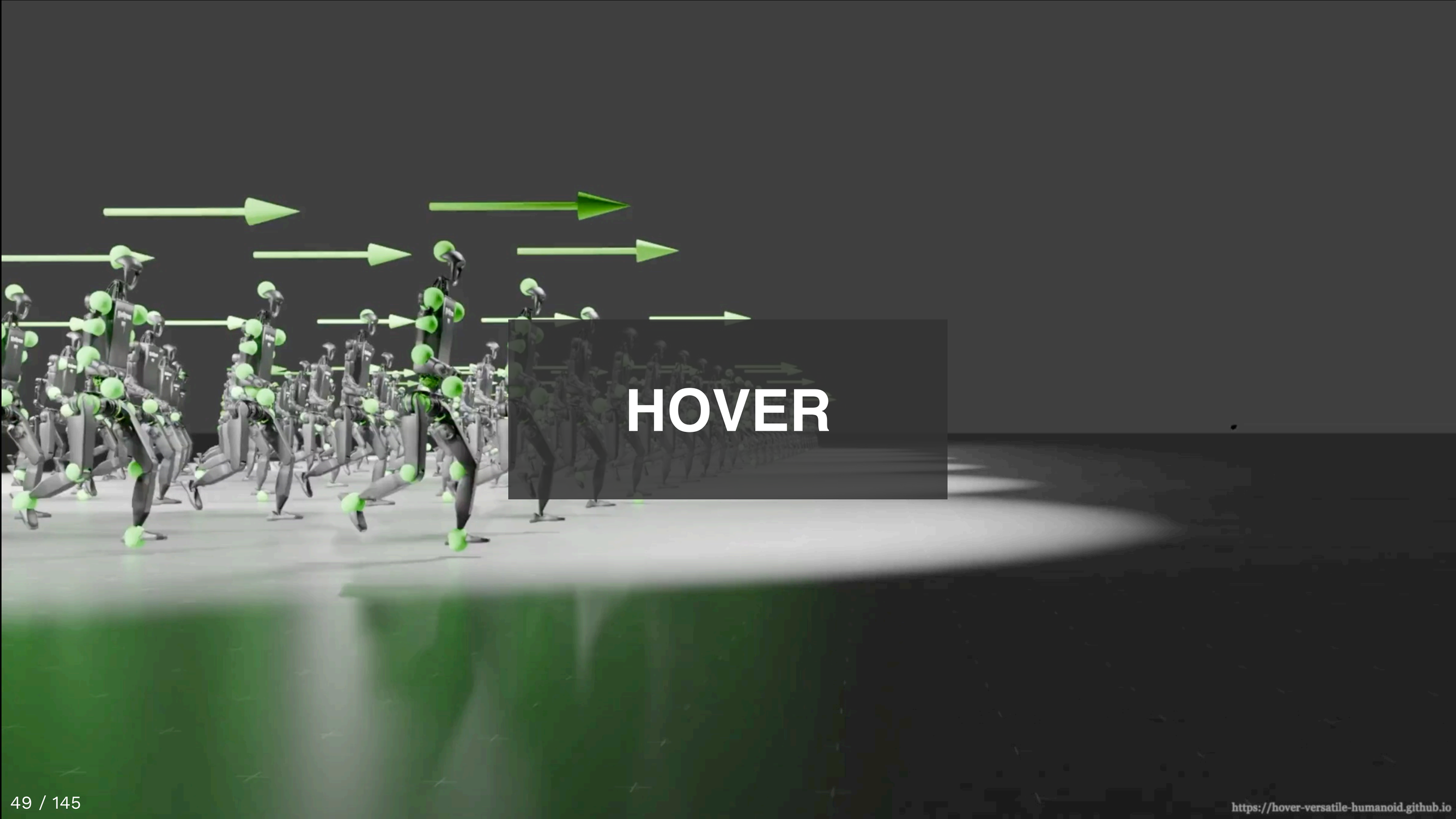
How many parameters
needed to build a Cerebellum?

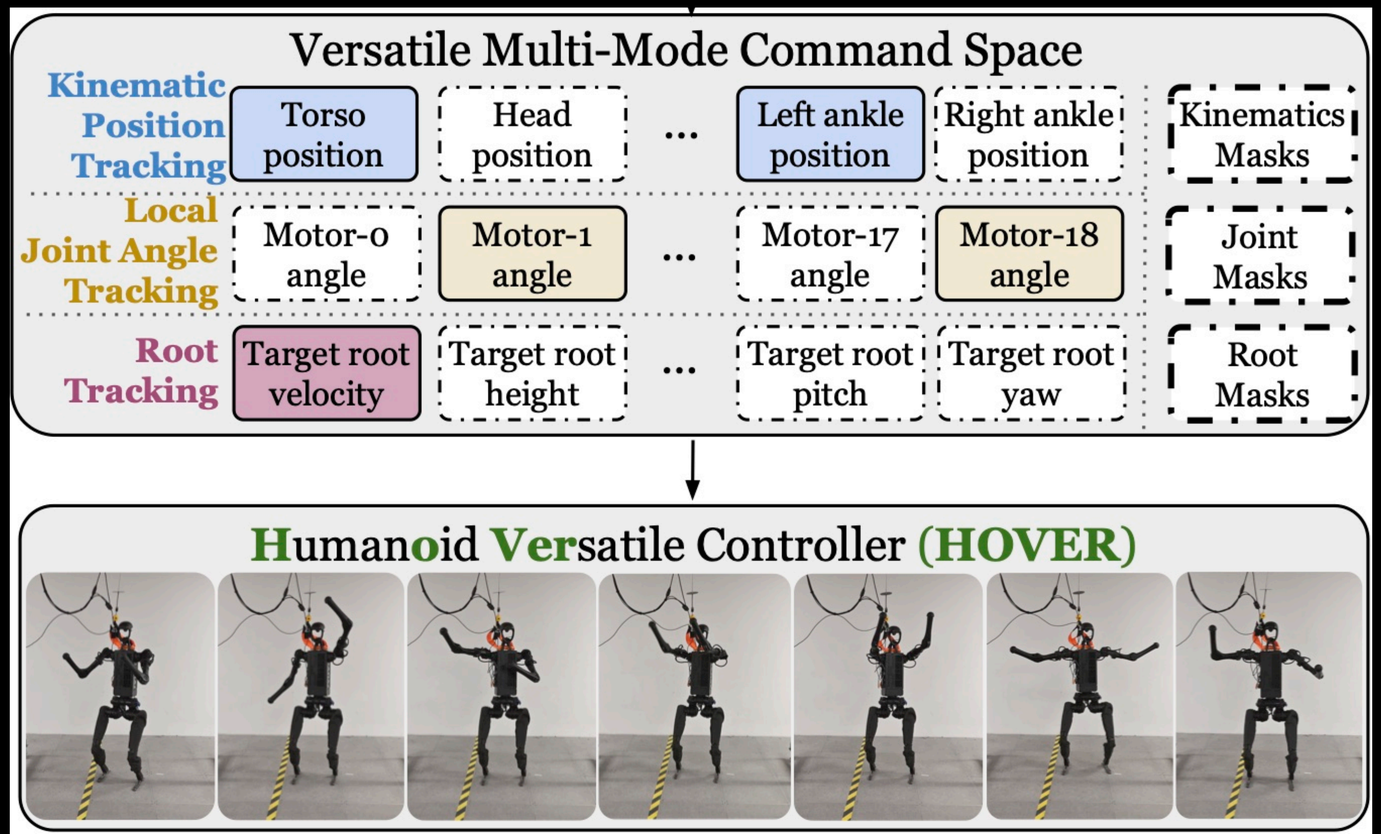


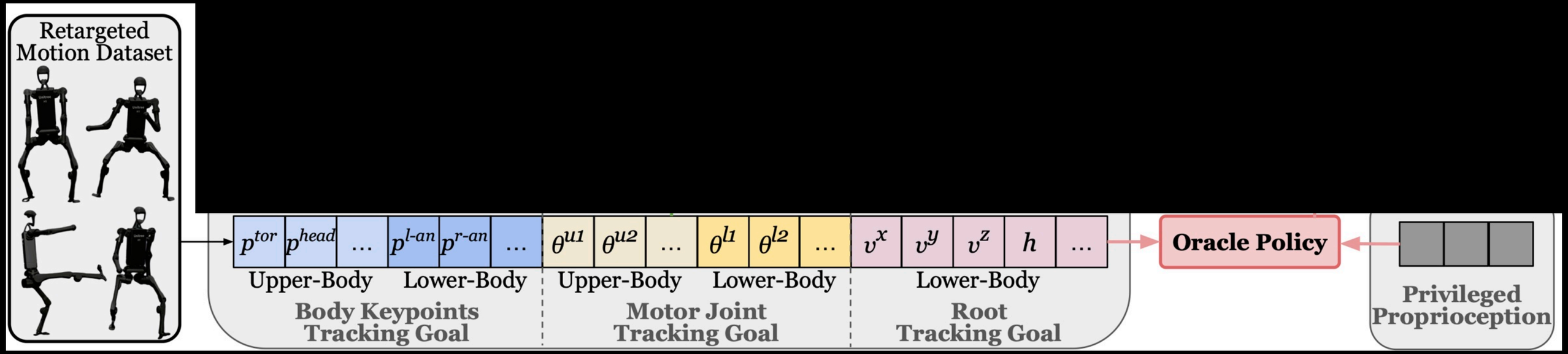
DANGER

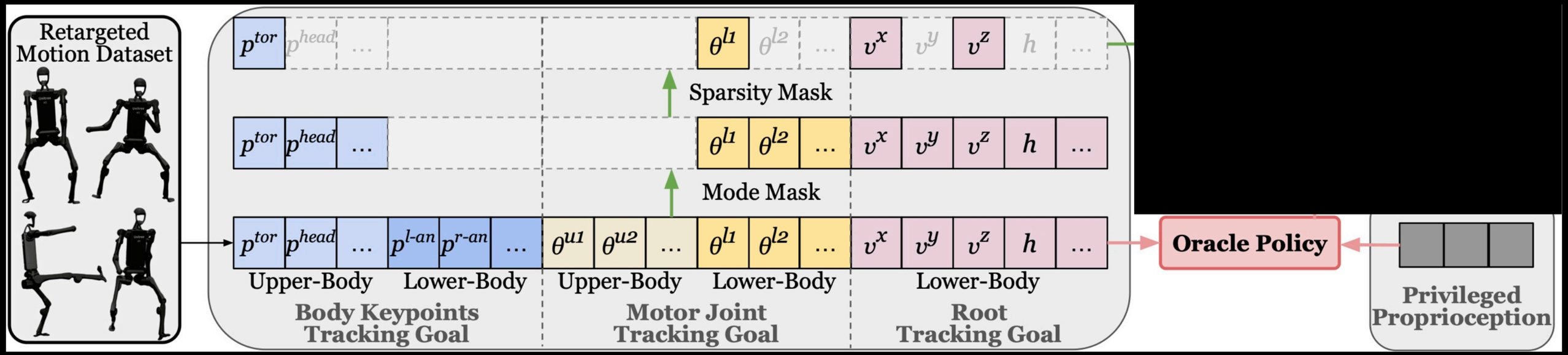
GAME CHANGE

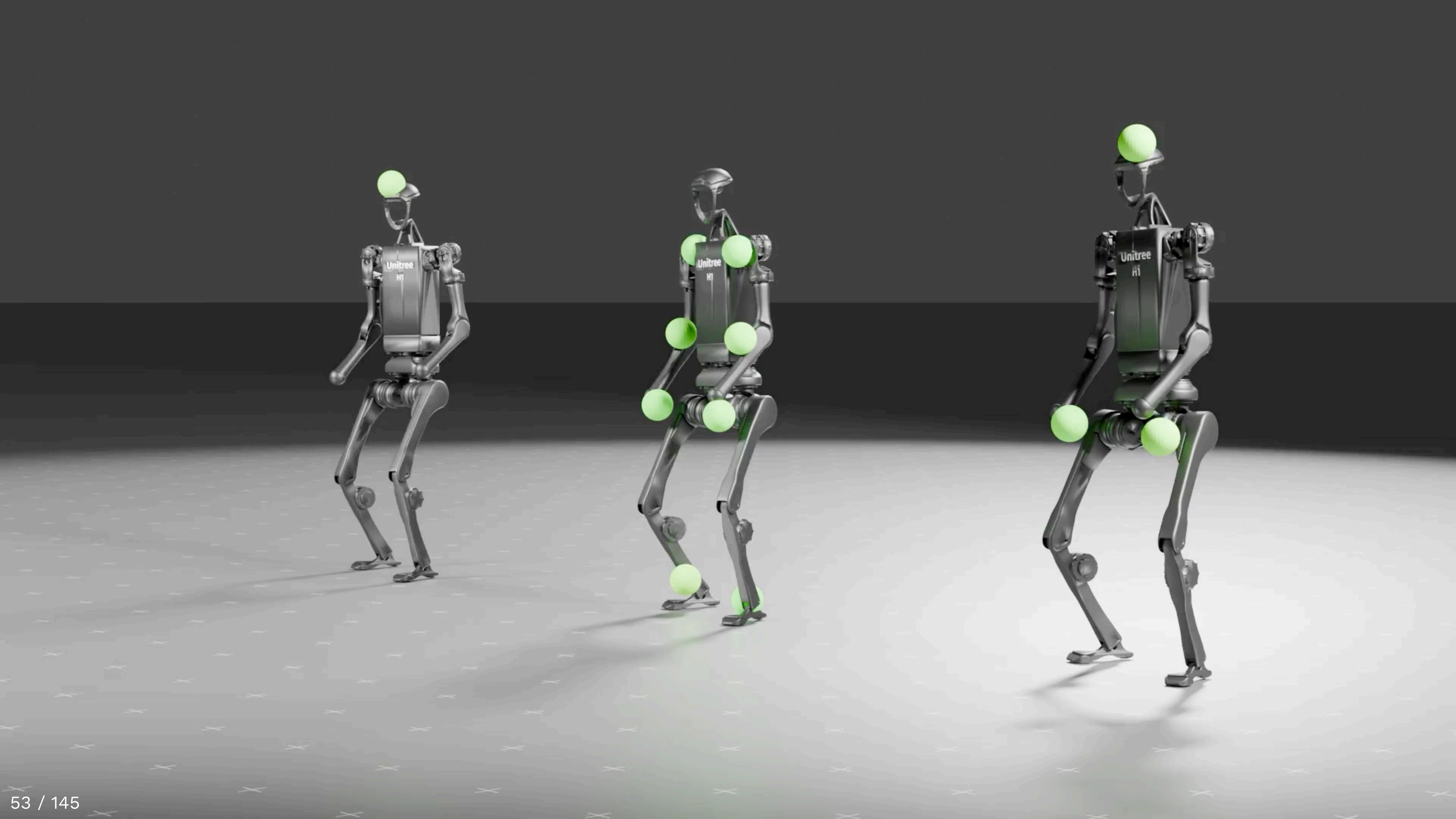
Fully Autonomous

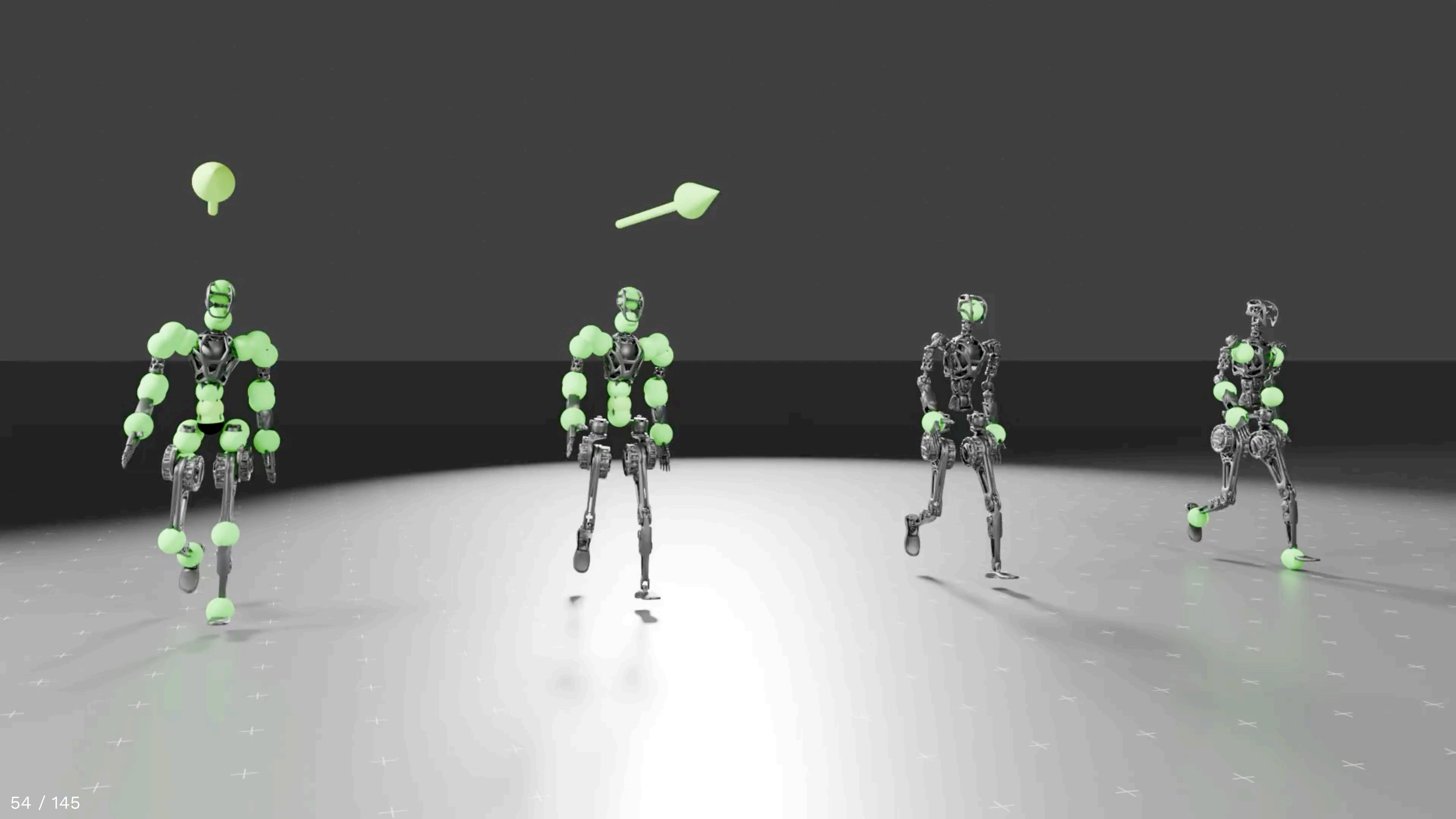

HOVER
https://hover-versatile-humanoid.github.io


Versatile Multi-Mode Command Space
Kinematic Position Tracking
Torso position
Head position
...
Left ankle position
Right ankle position
Kinematics Masks
Local Joint Angle Tracking
Motor-0 angle
Motor-1 angle
...
Motor-17 angle
Motor-18 angle
Joint Masks
Root Tracking
Target root velocity
Target root height
...
Target root pitch
Target root yaw
Root Masks
Humanoid Versatile Controller (HOVER)


Retargeted Motion Dataset
p^{tor} p^{head} … $p^{l\text{-}an}$ $p^{r\text{-}an}$ …
Upper-Body
Lower-Body
Body Keypoints Tracking Goal
θ^{u1} θ^{u2} … θ^{l1} θ^{l2} …
Upper-Body
Lower-Body
Motor Joint Tracking Goal
v^x v^y v^z h …
Lower-Body
Root Tracking Goal
Oracle Policy
Privileged Proprioception

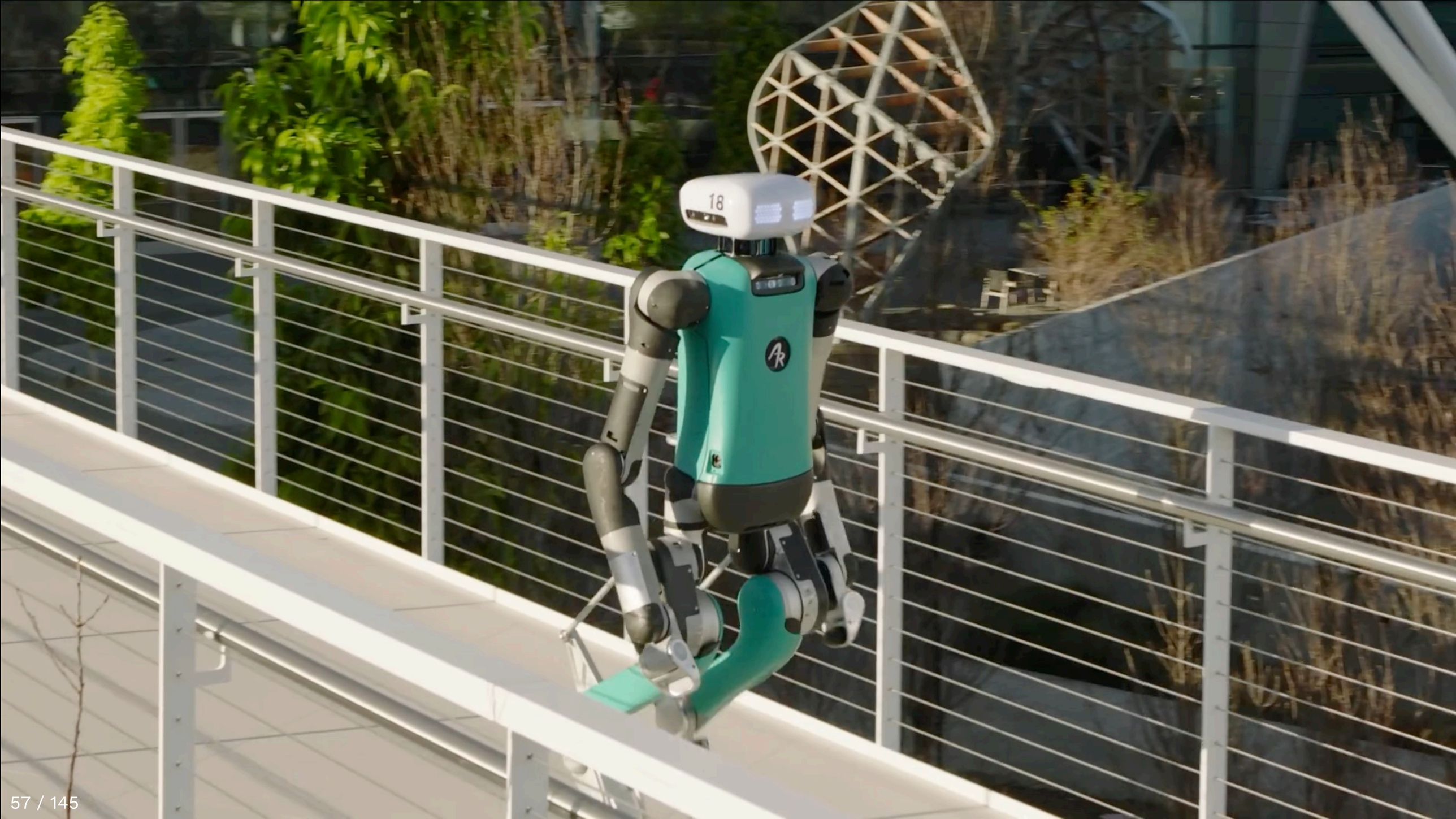

Retargeted Motion Dataset
p^{tor} p^{head} … θ^{l1} θ^{l2} … v^x v^y v^z h …
Sparsity Mask
p^{tor} p^{head} … θ^{l1} θ^{l2} … v^x v^y v^z h …
Mode Mask
p^{tor} p^{head} … $p^{l\text{-}an}$ $p^{r\text{-}an}$ … θ^{u1} θ^{u2} … θ^{l1} θ^{l2} … v^x v^y v^z h …
Upper-Body
Lower-Body
Upper-Body
Lower-Body
Lower-Body
Body Keypoints Tracking Goal
Motor Joint Tracking Goal
Root Tracking Goal
Oracle Policy
Privileged Proprioception

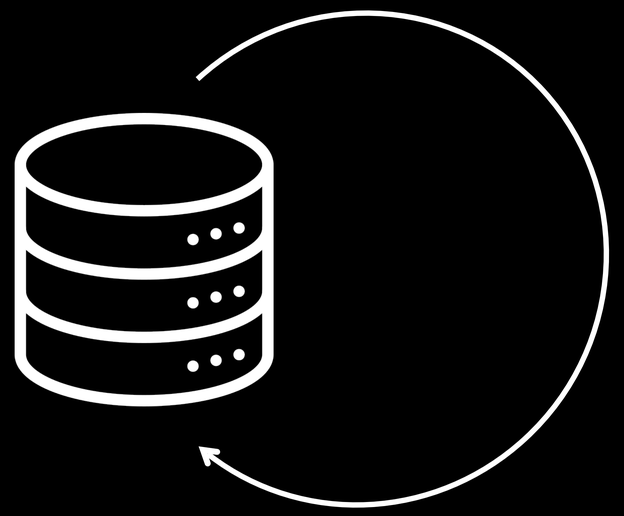

Unitree
H1
Unitree
H1
Unitree
H1



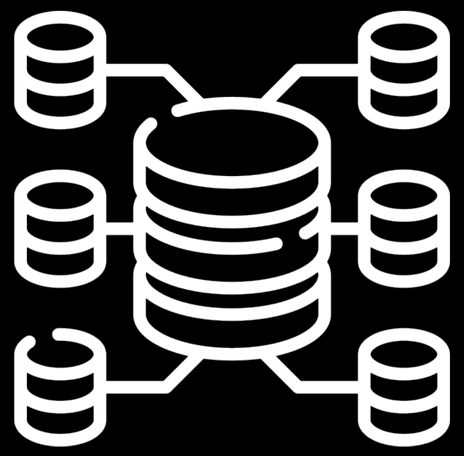
Training Iteration 500

Motion Description: *Walk forward and give a high-five.*

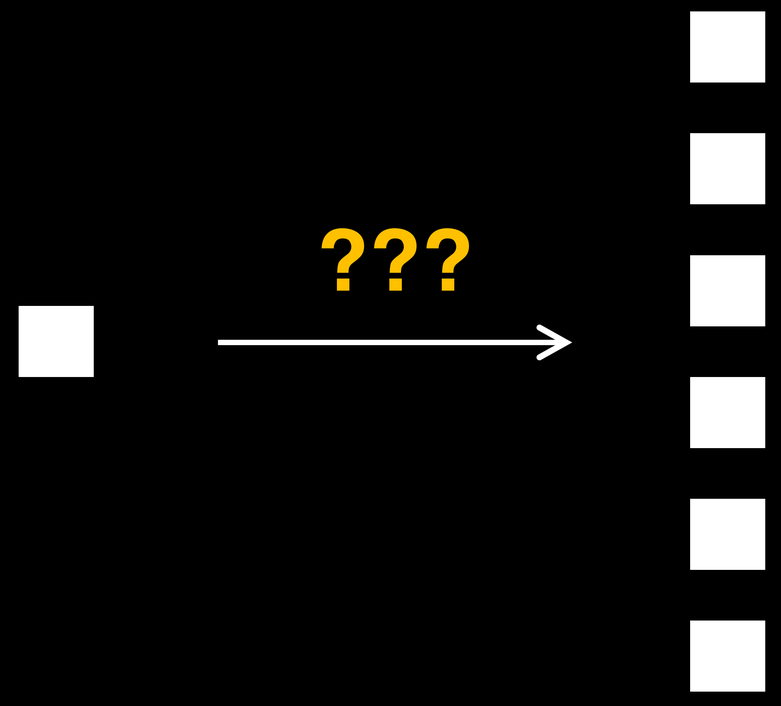

18



Reinforcement Learning



Imitation Learning



1 human demo

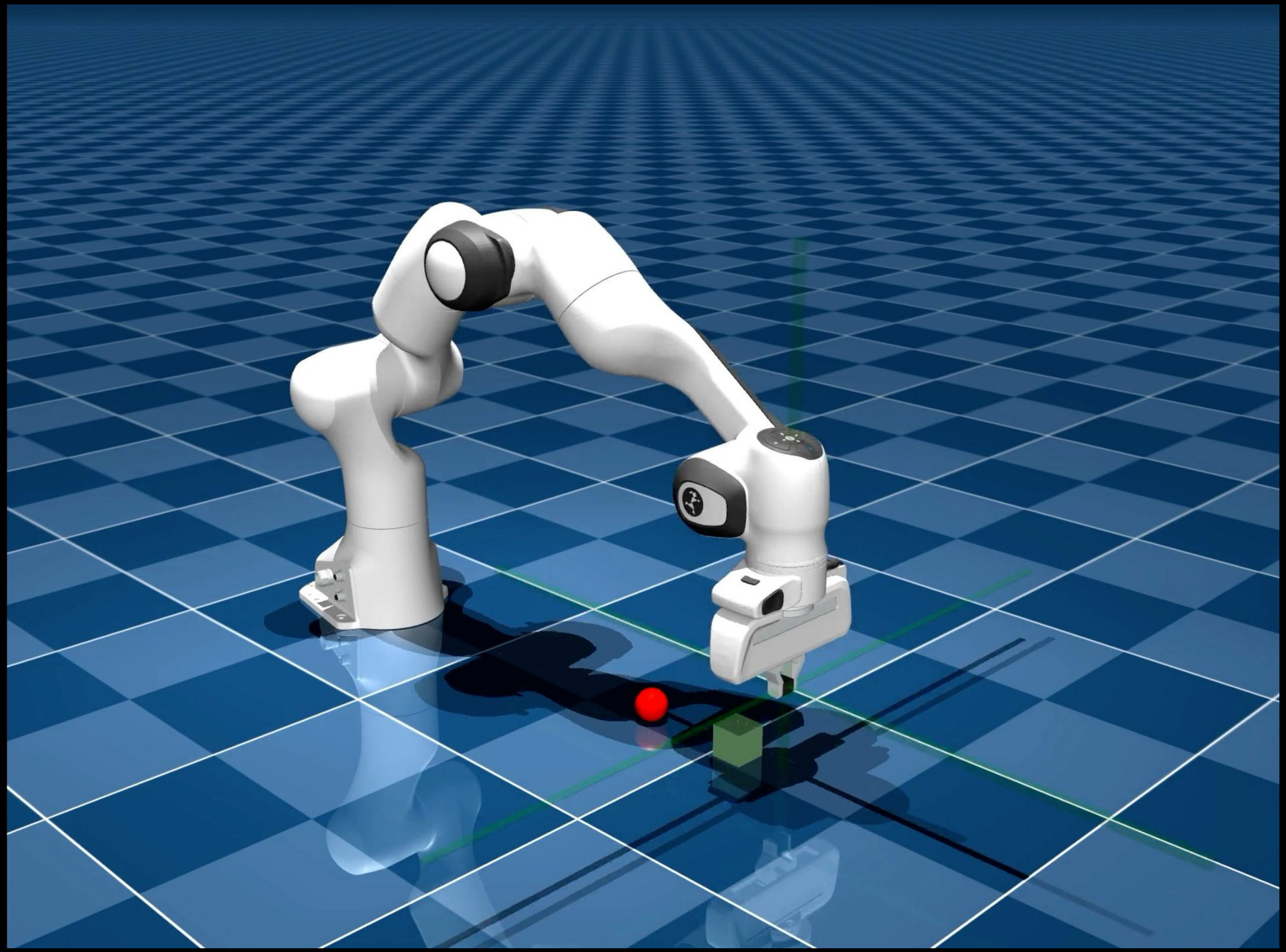
N

What’s in a Robot Simulation?

Hand-made Objects

Hand-crafted Scenes

Hand-designed Tasks



Xu et al. 2023

What's in a Robot Simulation?

Hand-made
Objects

↓

Generate 3D

Text-to-3D models

BEKO

What's in a Robot Simulation?

Hand-made Objects

↓

Generate 3D

Text-to-3D models

Hand-crafted Scenes

↓

Generate USD

Stable Diffusion

What's in a Robot Simulation?

Hand-made Objects

↓

Generate 3D

Text-to-3D models

Hand-crafted Scenes

↓

Generate USD

Stable Diffusion

ChatUSD

What's in a Robot Simulation?

Hand-made Objects	Hand-crafted Scenes	Hand-designed Tasks
↓	↓	↓
Generate 3D	Generate USD	Generate Code
Text-to-3D models	Stable Diffusion, ChatUSD	**GPT-4o**, **Claude-3.5**

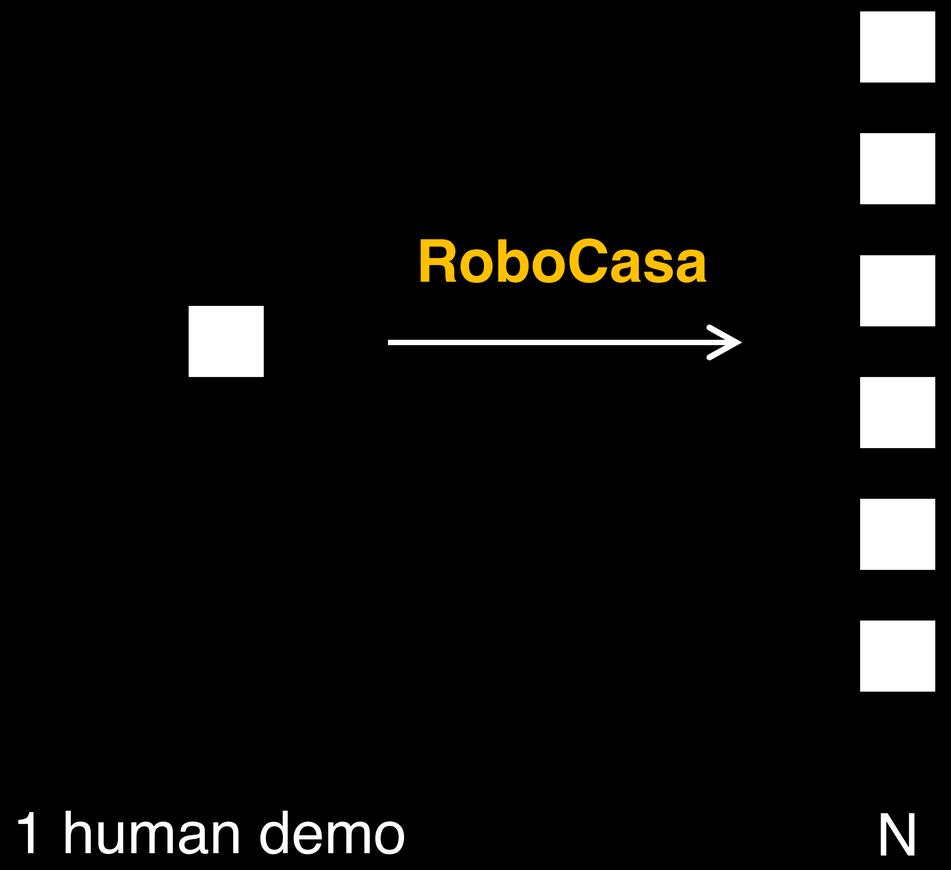
FRIGIDAIRE

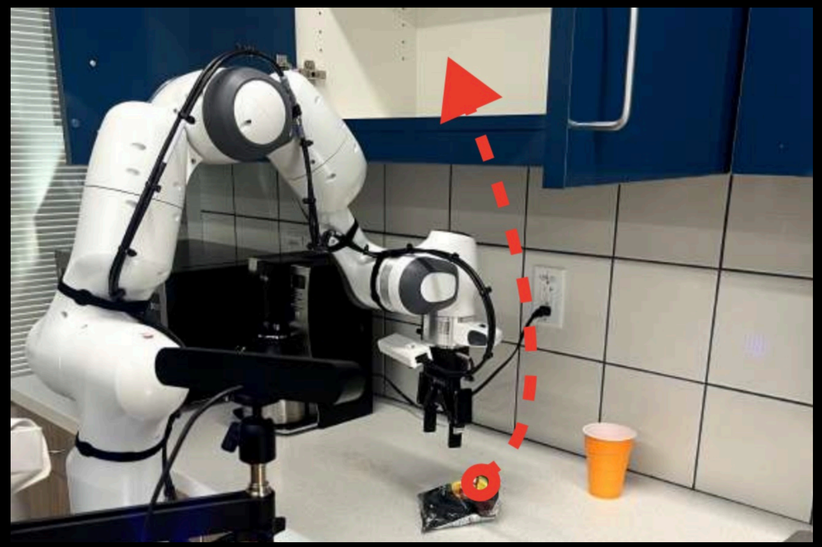
RoboCasa

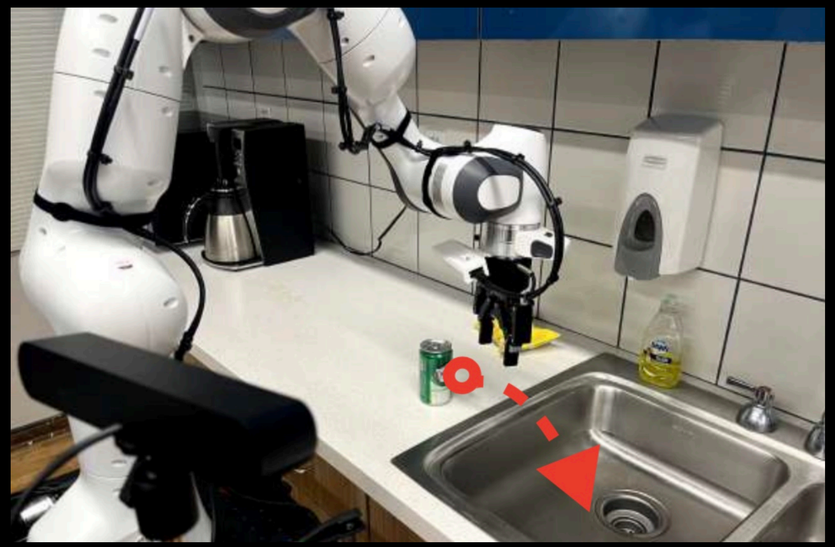
RoboCasa: Generative Simulation Framework

Hand-made Objects	Hand-crafted Scenes	Hand-designed Tasks
↓	↓	↓
Generate 3D	Generate USD	Generate Code
Text-to-3D models	Stable Diffusion, ChatUSD	GPT-4o, Claude-3.5

Nasiriany et al. RSS 2024.


RoboCasa
1 human demo
N

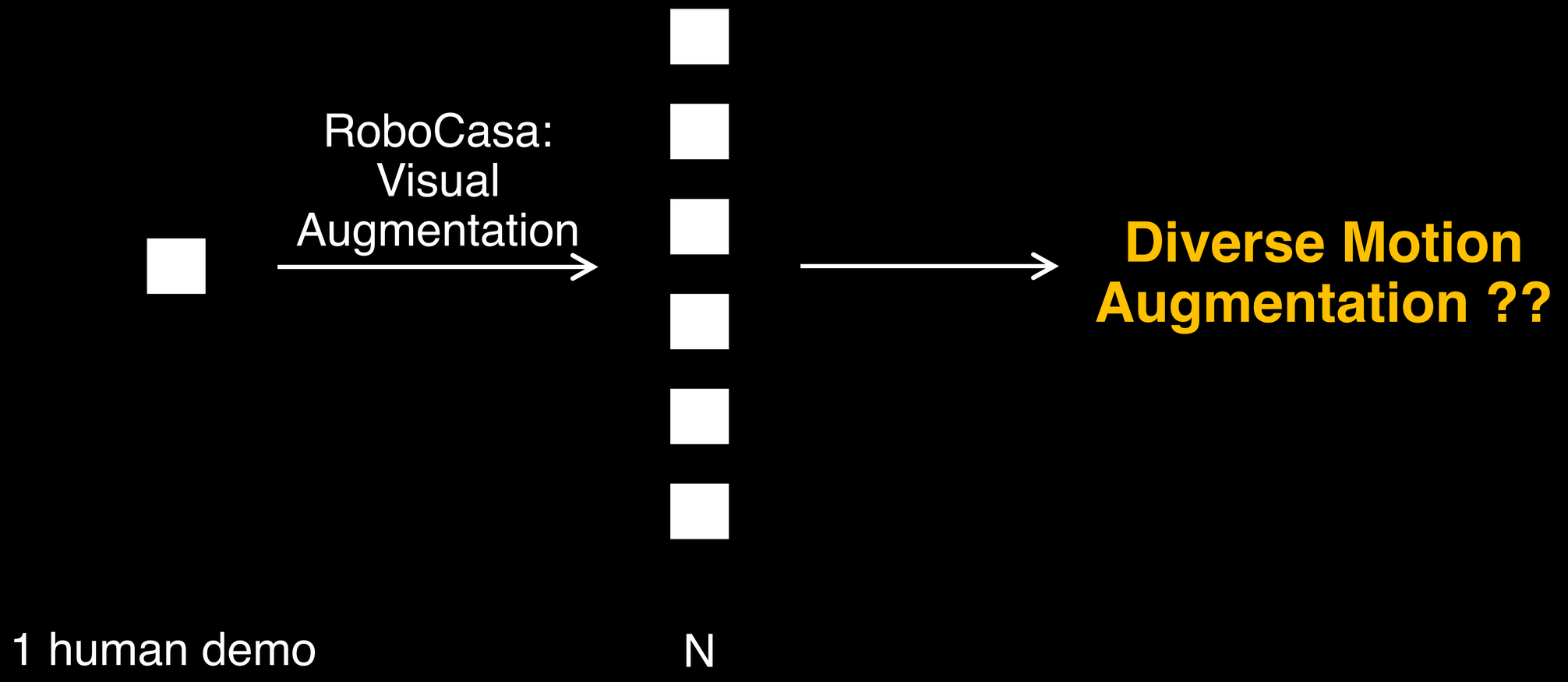


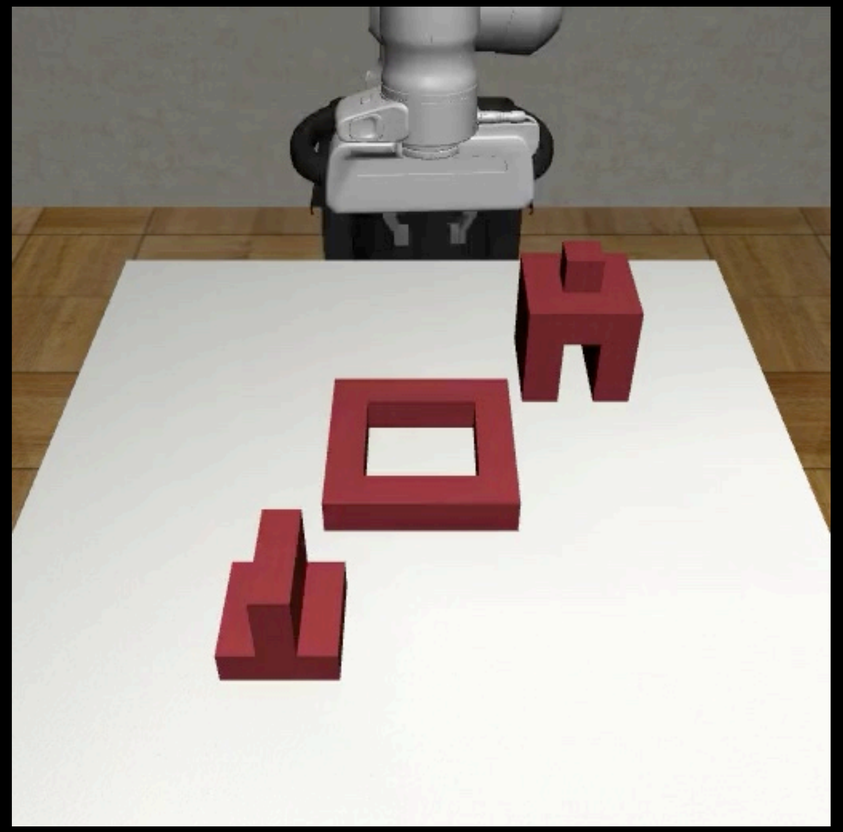


Setting	Task	Real only	Real + Sim (Ours)
Seen Obj	Counter to sink	12.7 ± 2.5	**22.0 ± 2.8**
	Sink to counter	20.0 ± 5.9	**29.3 ± 4.1**
	Counter to cabinet	8.0 ± 1.6	**22.0 ± 5.8**
	Task average	13.6	**24.4**
Unseen Obj	Counter to sink	3.3 ± 4.7	**8.9 ± 7.9**
	Sink to counter	1.1 ± 1.6	**7.8 ± 4.2**
	Counter to cabinet	3.3 ± 4.7	**11.1 ± 11.0**
	Task average	2.6	**9.3**

Nasiriany et al. RSS 2024.

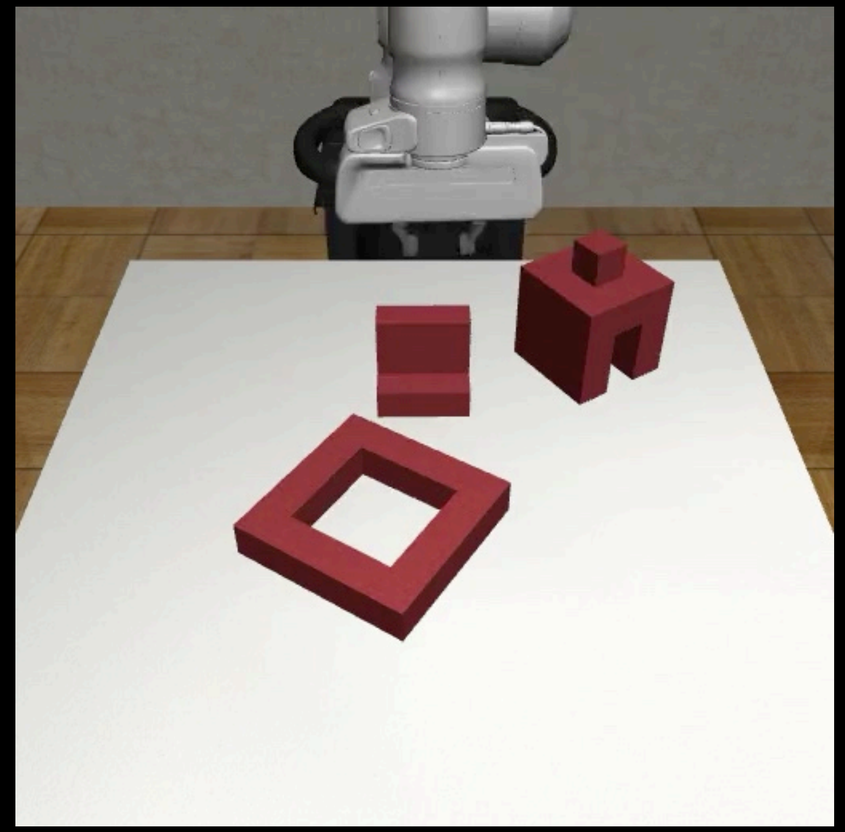
Omniverse


RoboCasa:
Visual
Augmentation
Diverse Motion
Augmentation ??
1 human demo
N



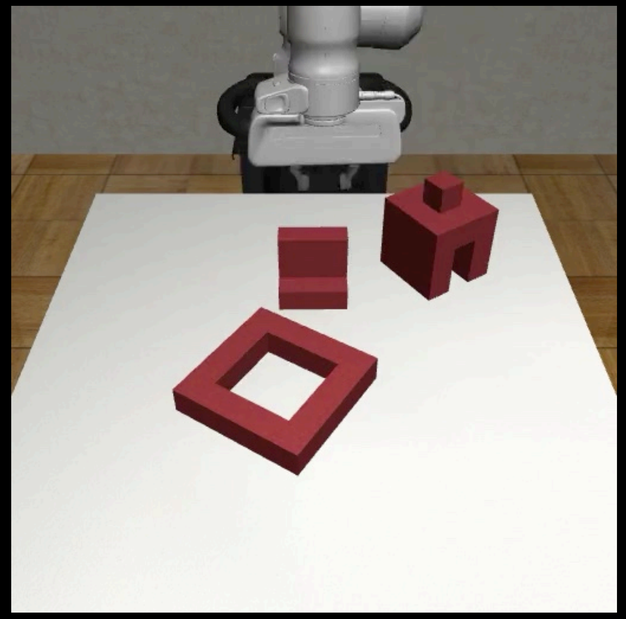
1 human demo (seed)

MimicGen

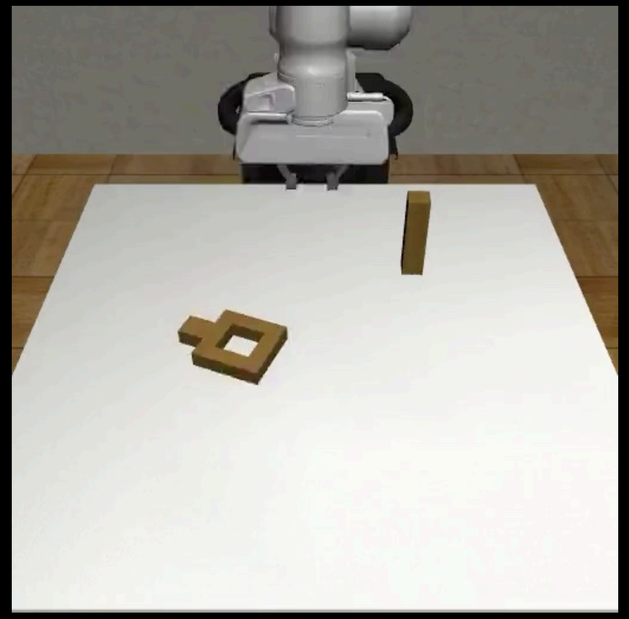


Machine-generated demos

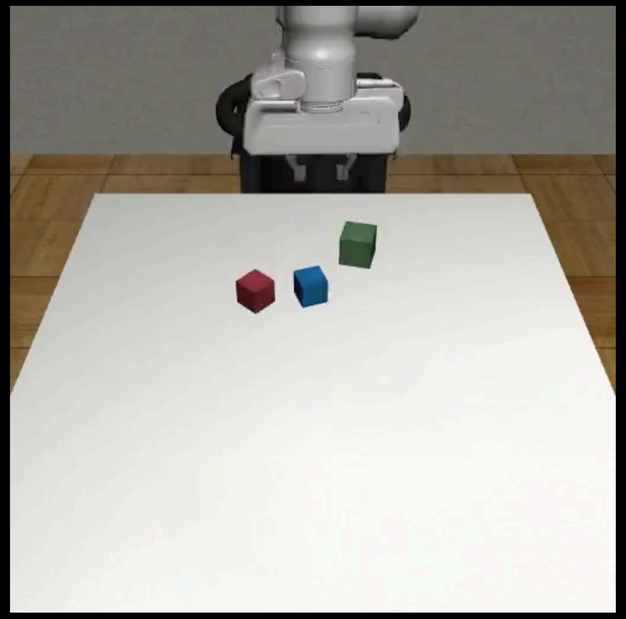
Mandlekar et al. CoRL 2023.



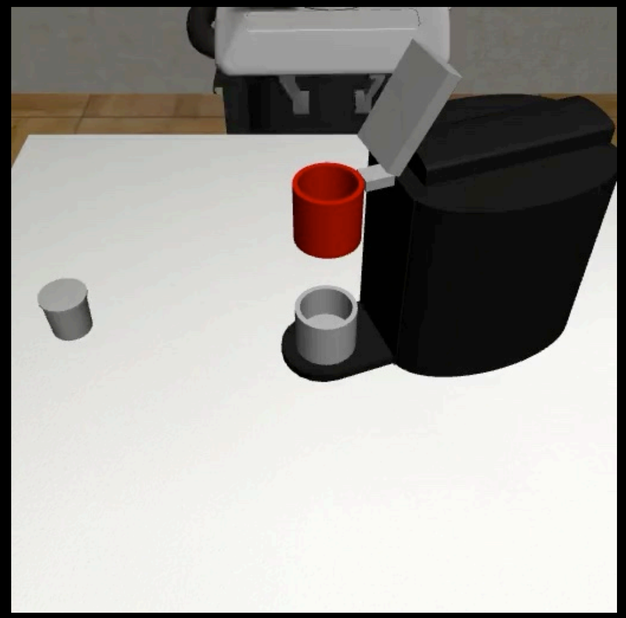
3-Piece Assembly



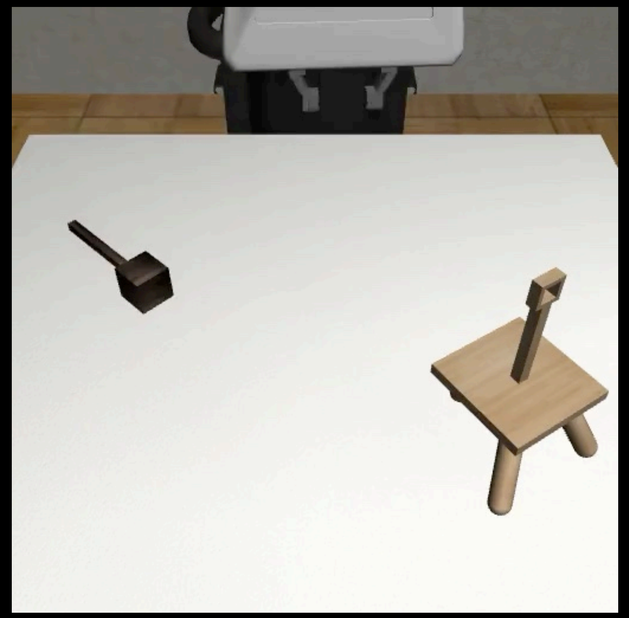
Square Insertion



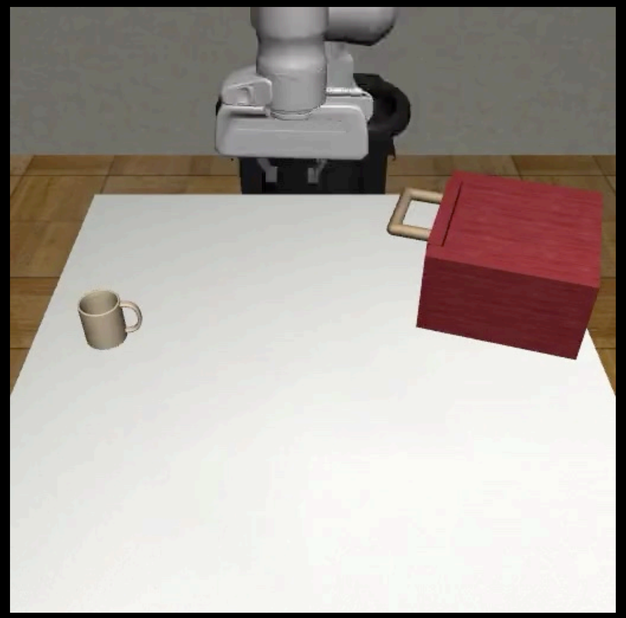
3-Cube Stacking



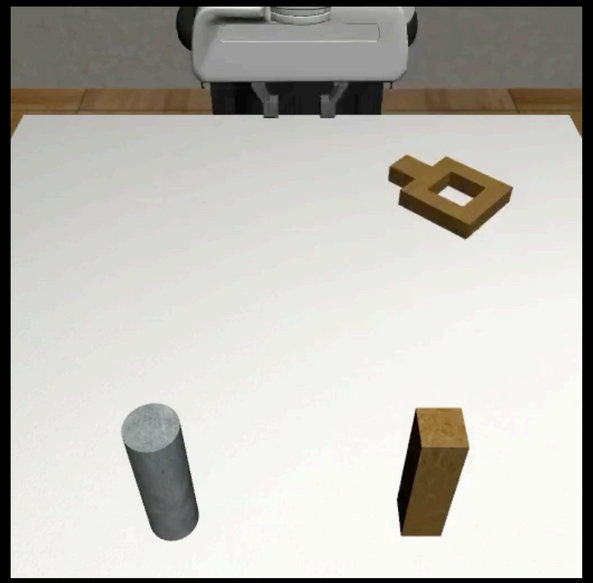
Making Coffee



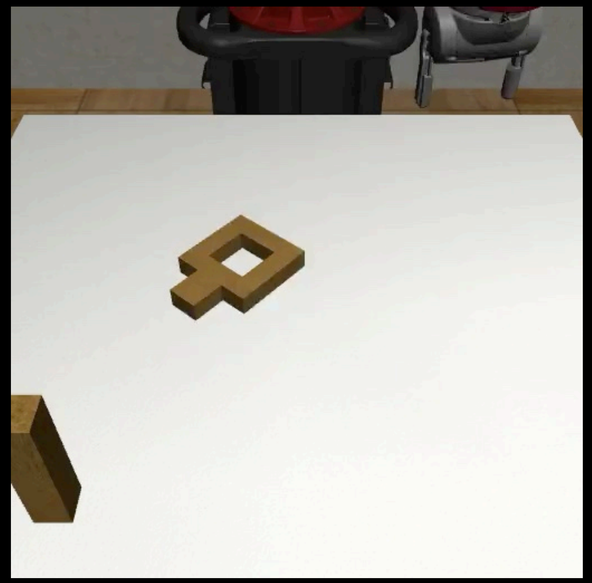
Threading



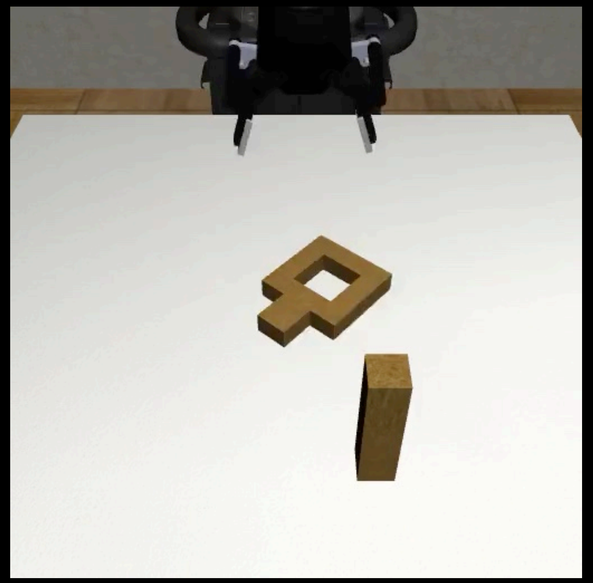
Put Mug in Drawer



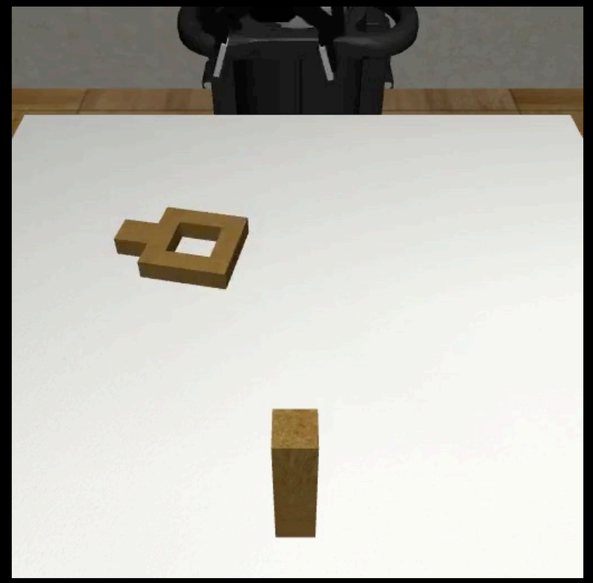
Panda



Sawyer

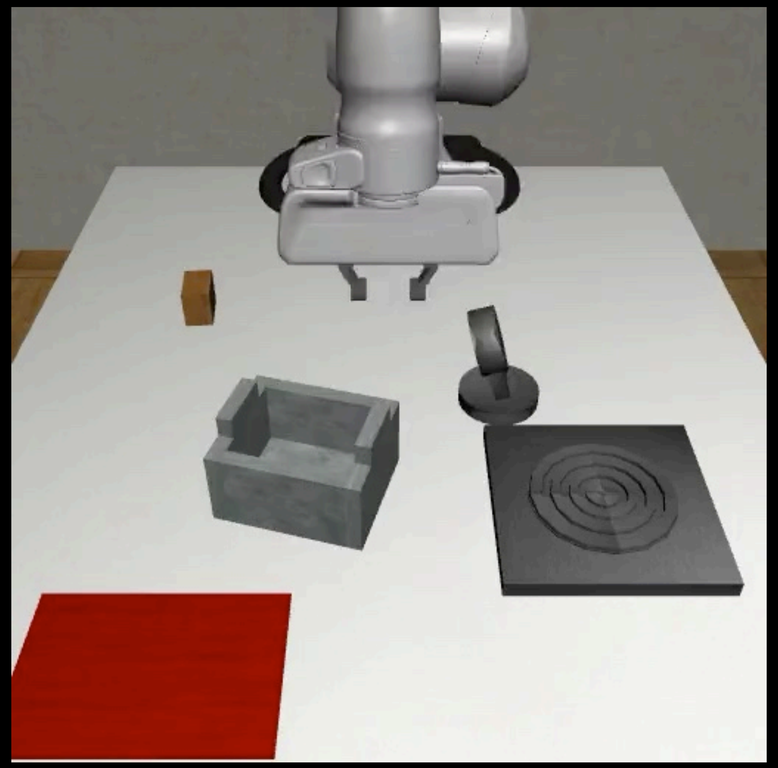


IIWA



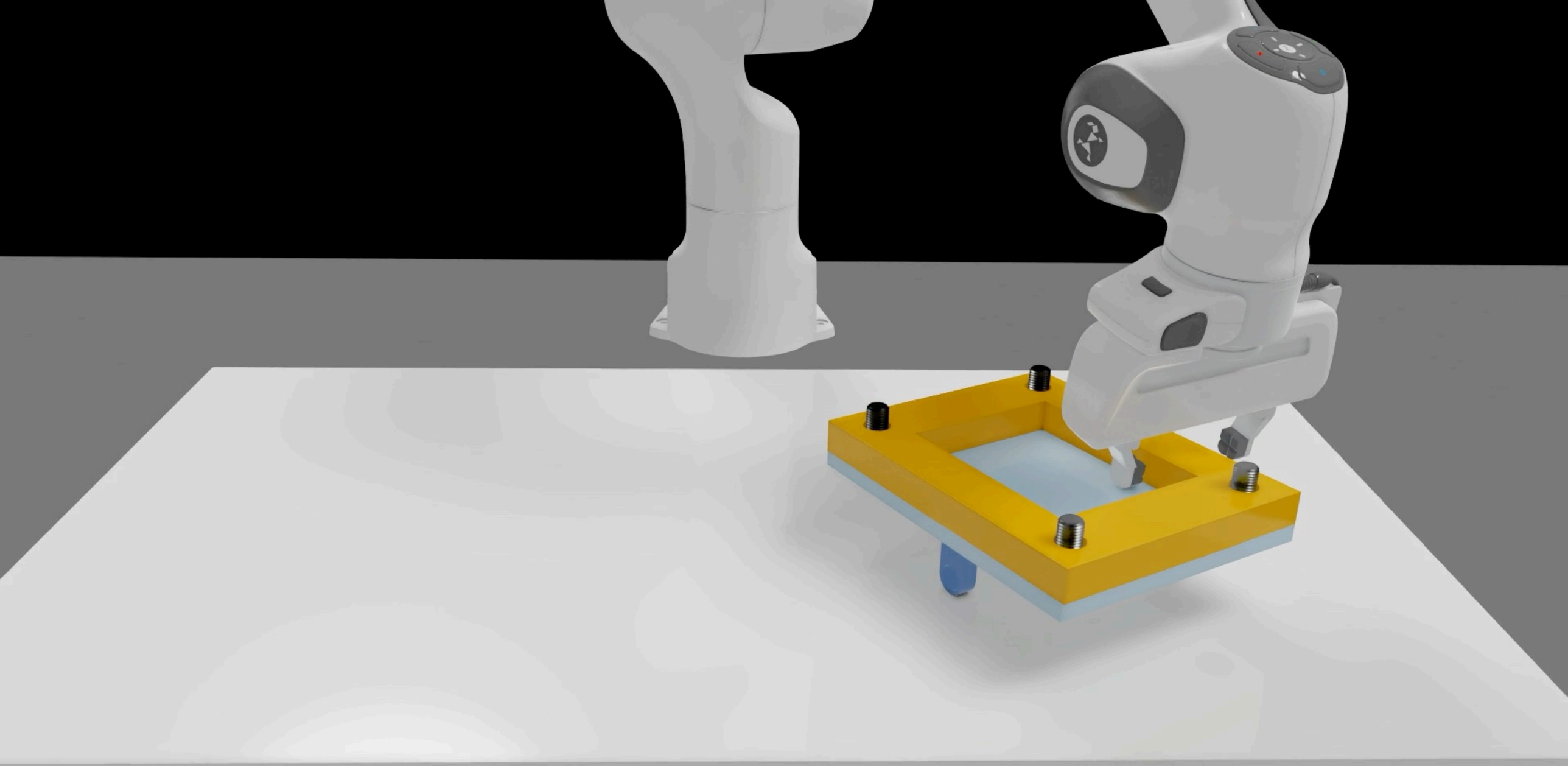
UR5e

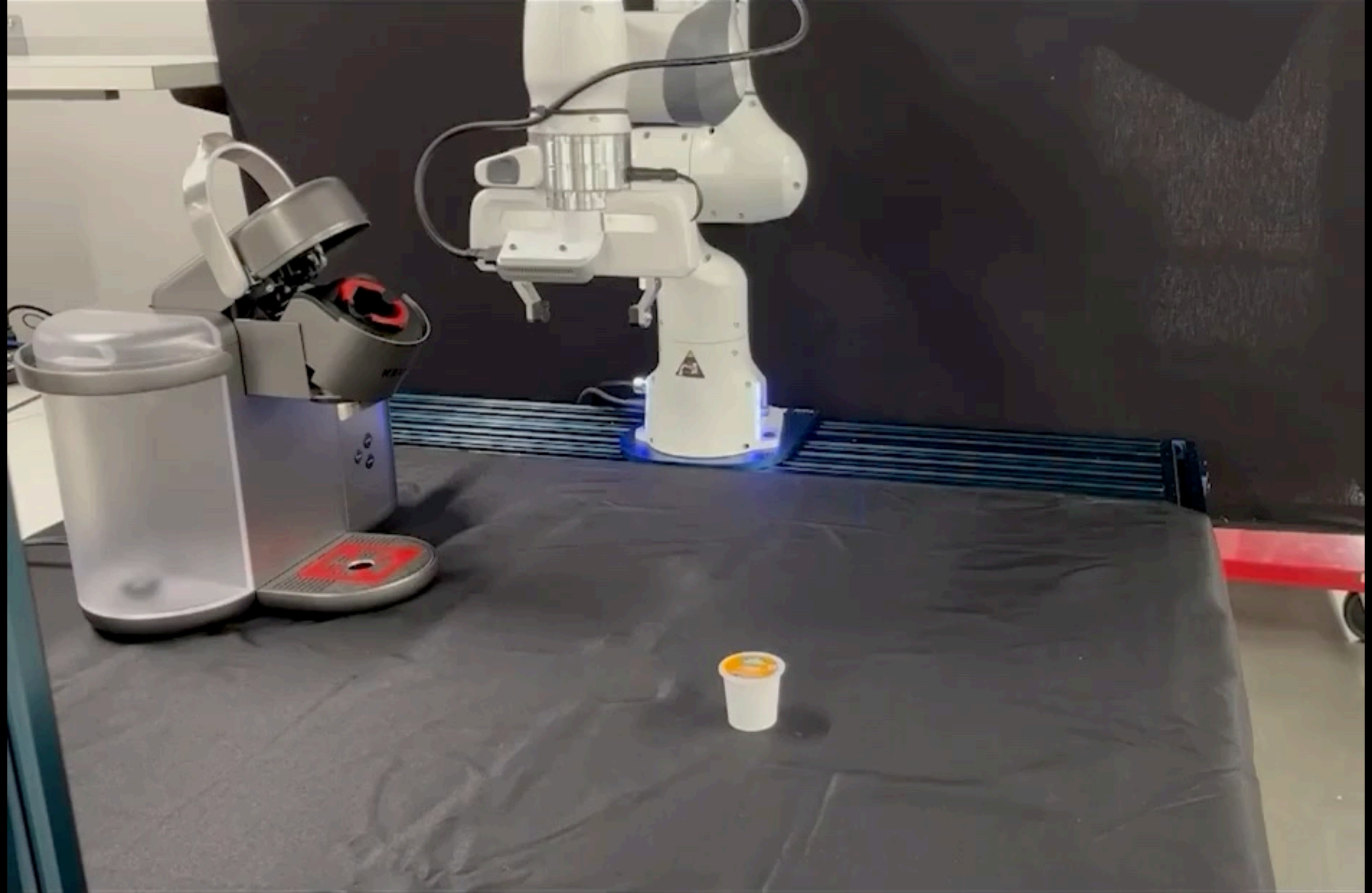
Prepare Coffee

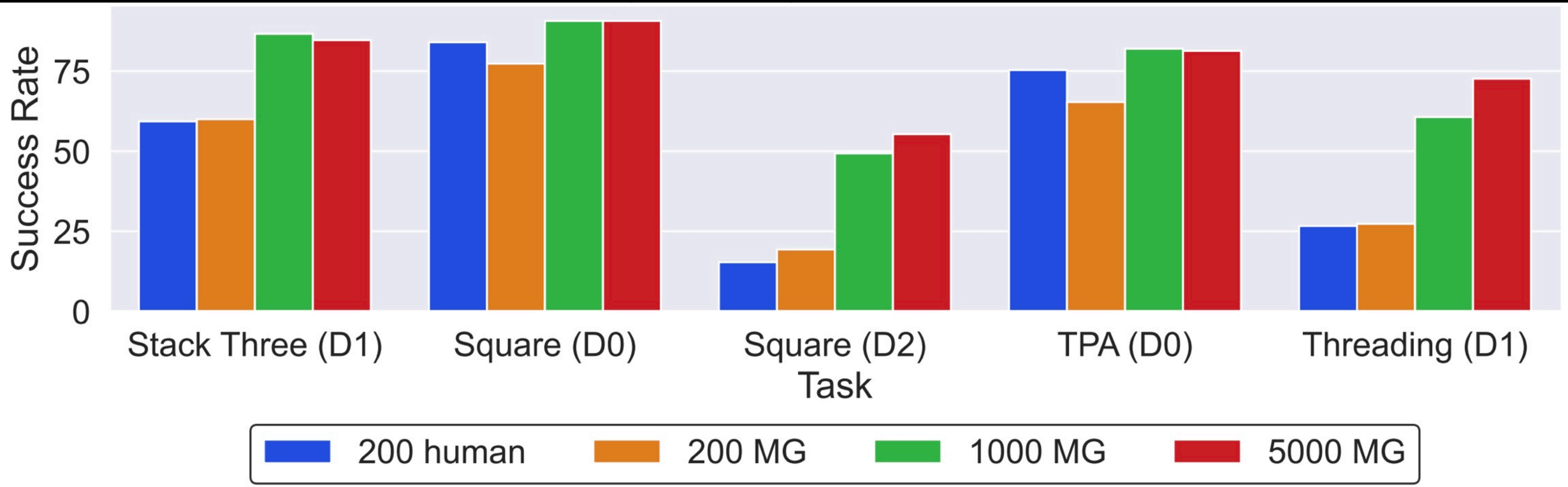


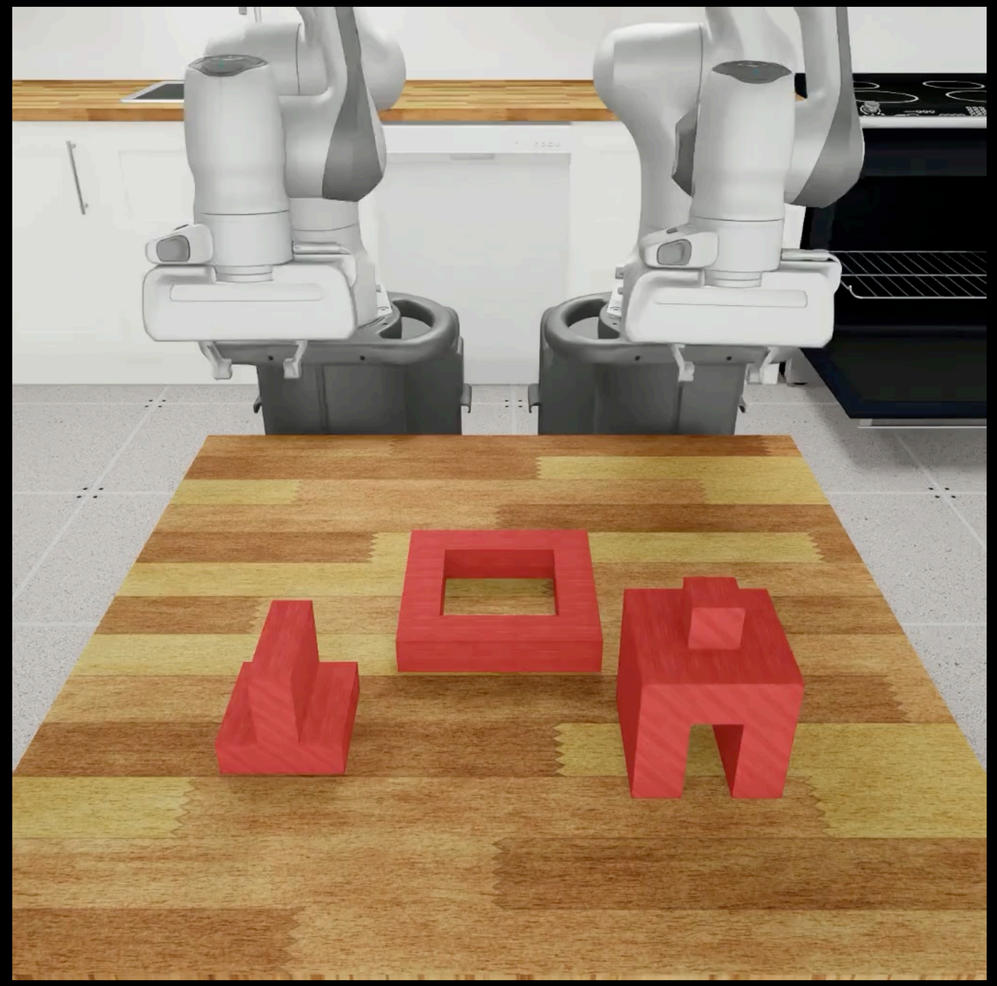
Kitchen Chores

Bin Sorting






Success Rate
0
25
50
75
Stack Three (D1)
Square (D0)
Square (D2)
TPA (D0)
Threading (D1)
Task
200 human
200 MG
1000 MG
5000 MG



Bimanual: **parallel** sub-tasks

- Asynchronous execution

Jiang & Xie et al. 2024.

Bimanual: **coordination** sub-tasks

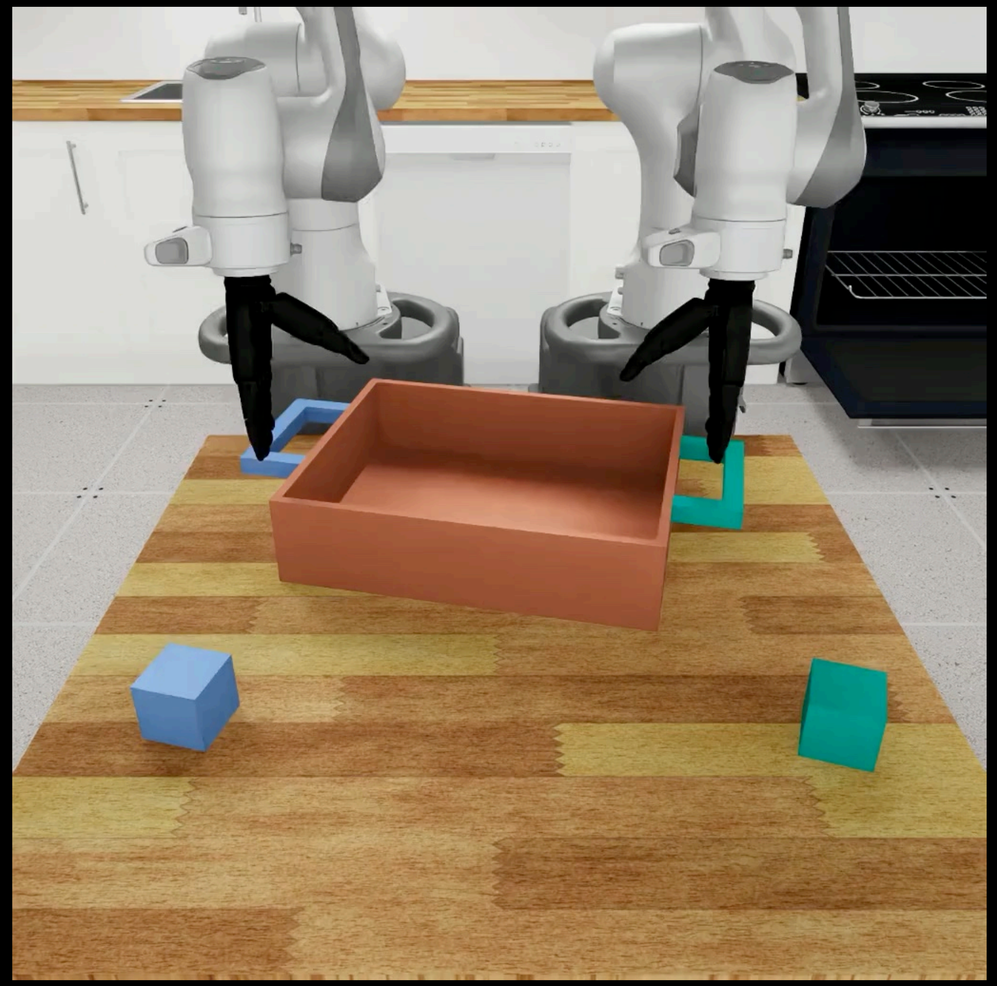
- Must apply same transformations
- Synchronization point

Jiang & Xie et al. 2024.

Bimanual: **sequential** sub-tasks

- Ordering constraint: pre-task, post-task

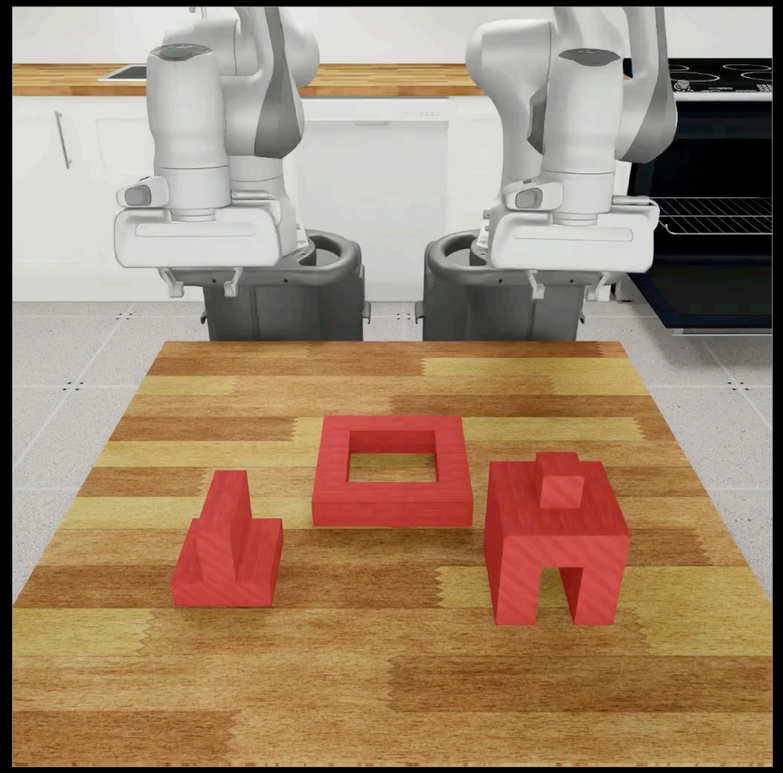
Jiang & Xie et al. 2024.



Sub-task strategies can mix & match

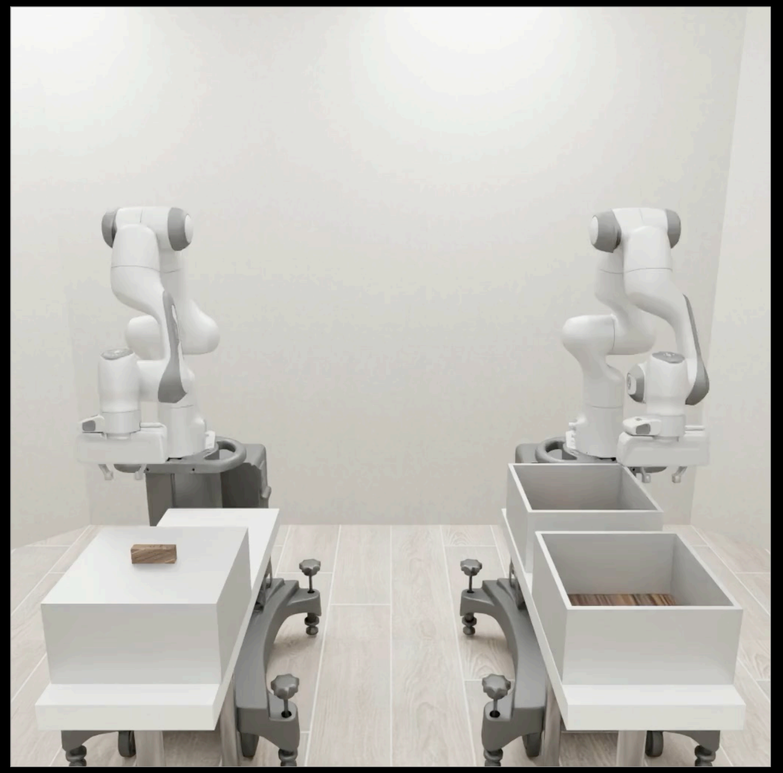
Parallel then **Coordination**.

Jiang & Xie et al. 2024.

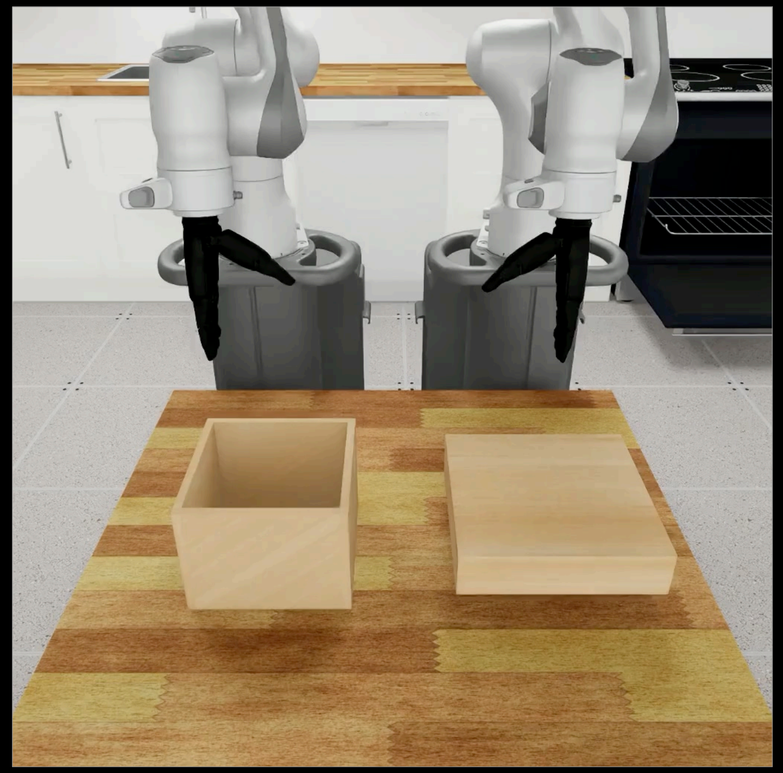


2-Gripper Piece Assembly

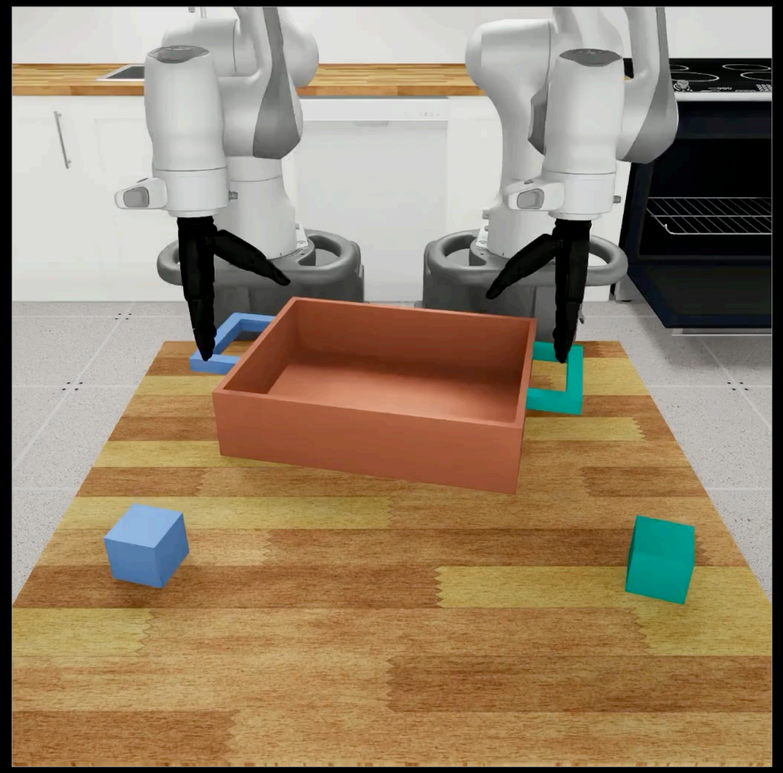
2-Gripper Transport



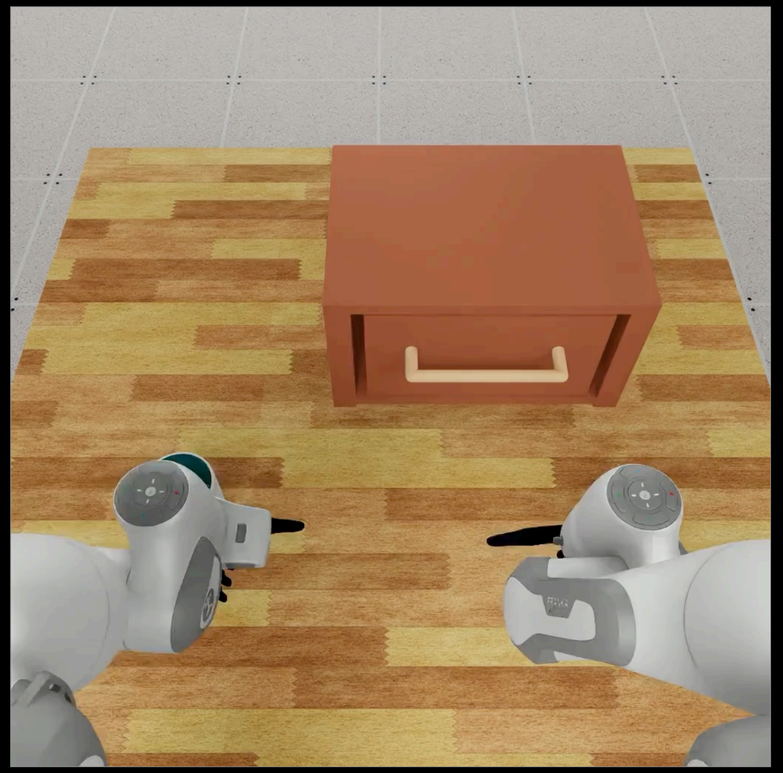
2-Gripper Threading



2-Hand Putting Box Lid



2-Hand Tray Lifting



2-Hand Drawer Cleanup

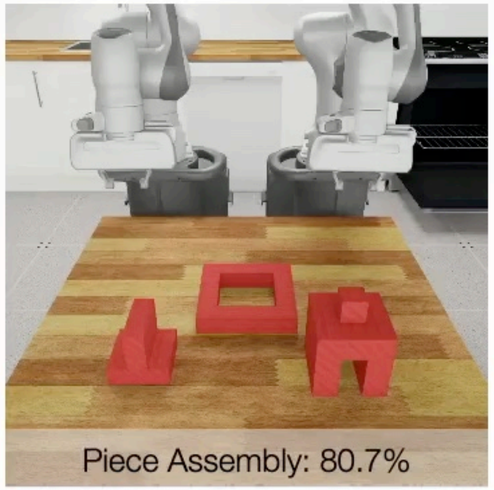
Humanoid Making Coffee

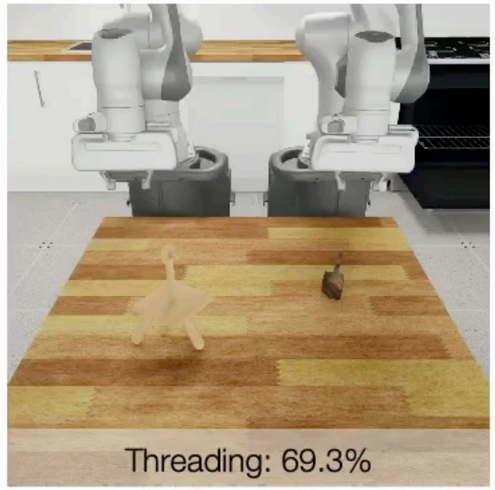
Humanoid Sorting Cans

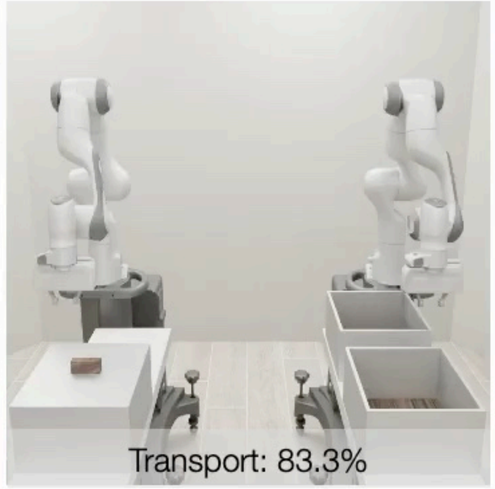
Humanoid Pouring

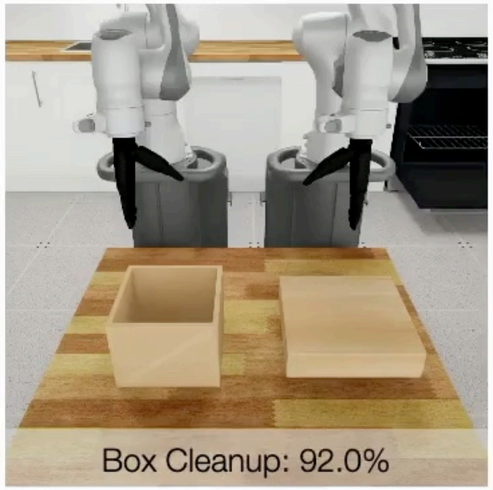
DexMimicGen

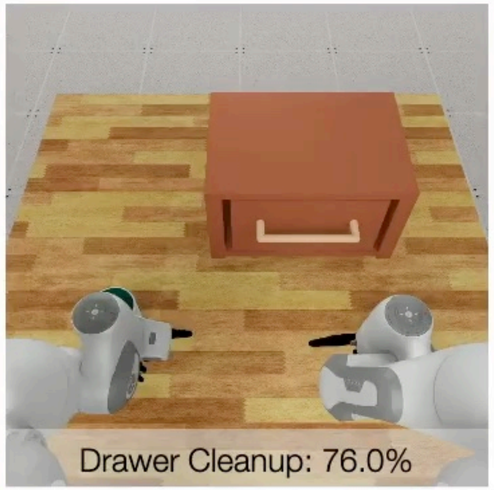
Behavior cloning on DexMimicGen data trains performant policies.

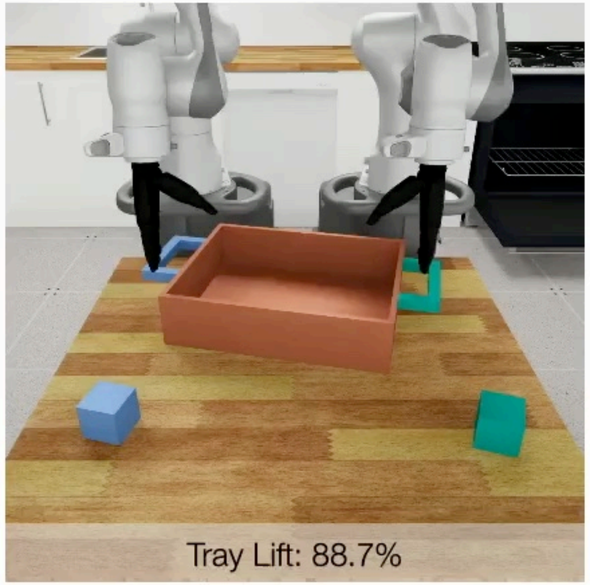


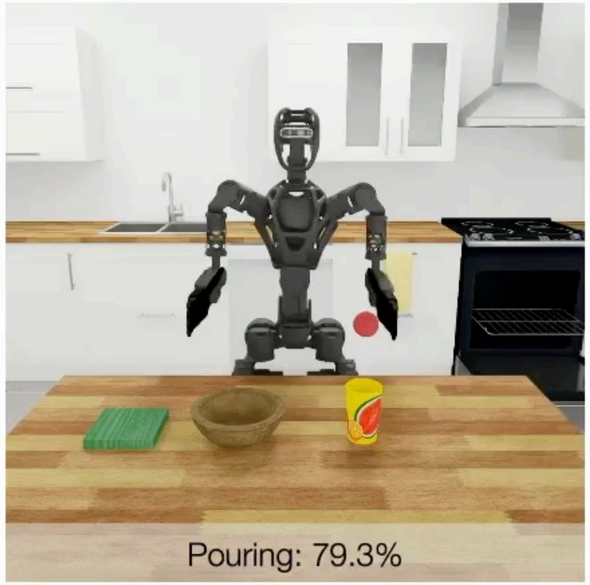


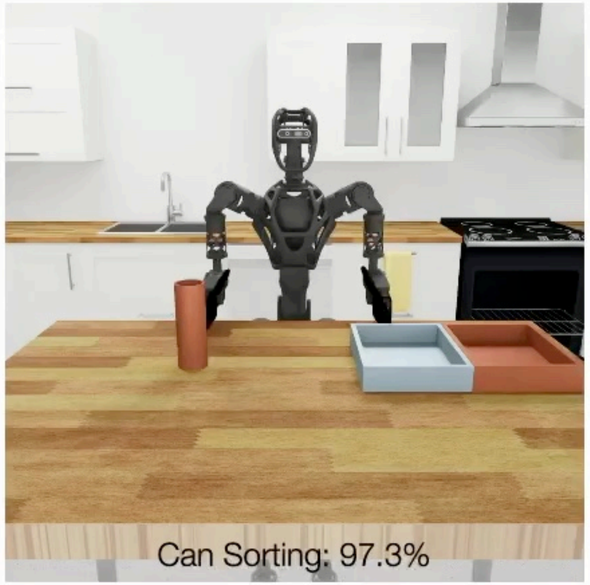




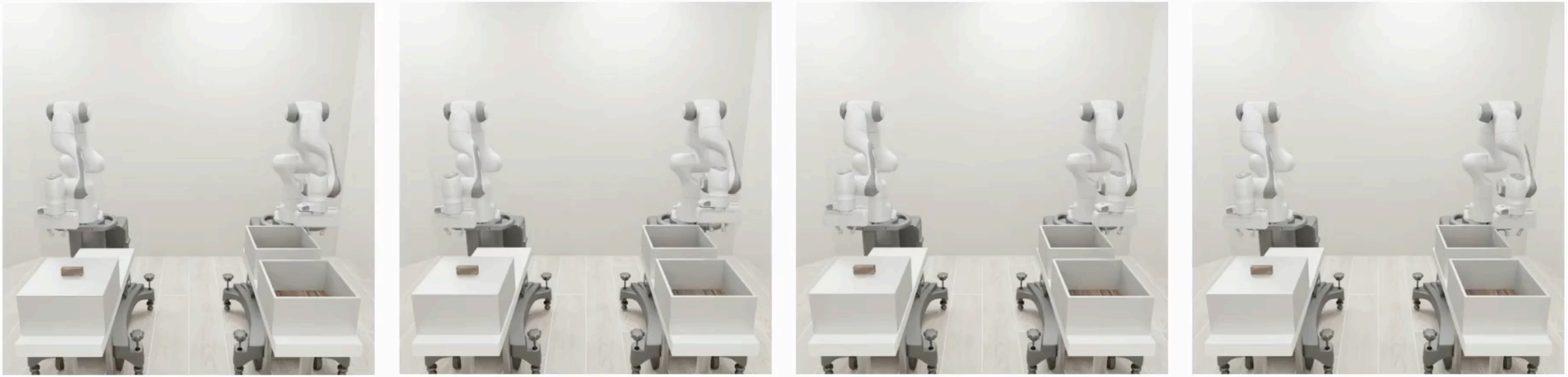


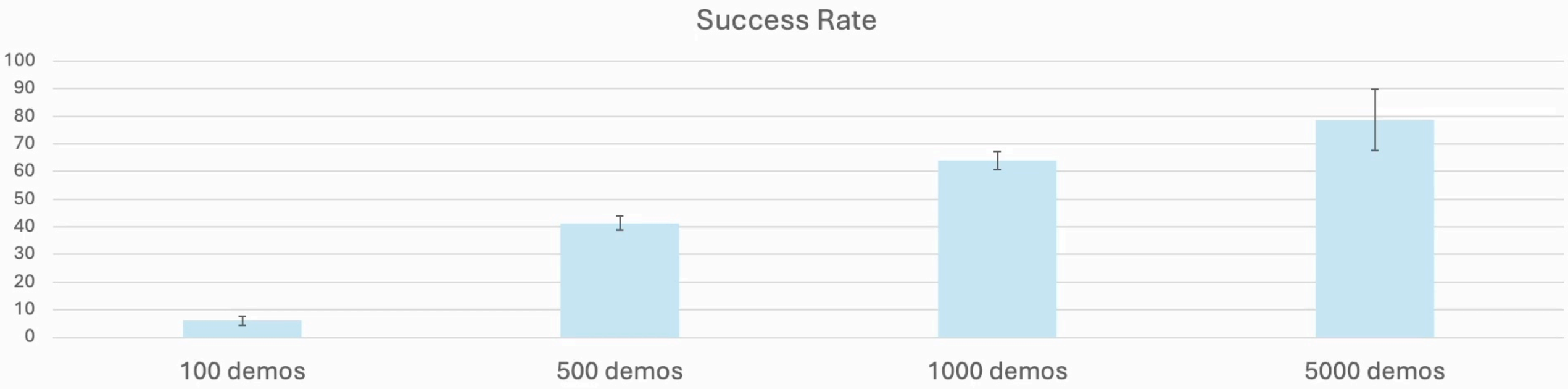






Success rates of Diffusion Policy

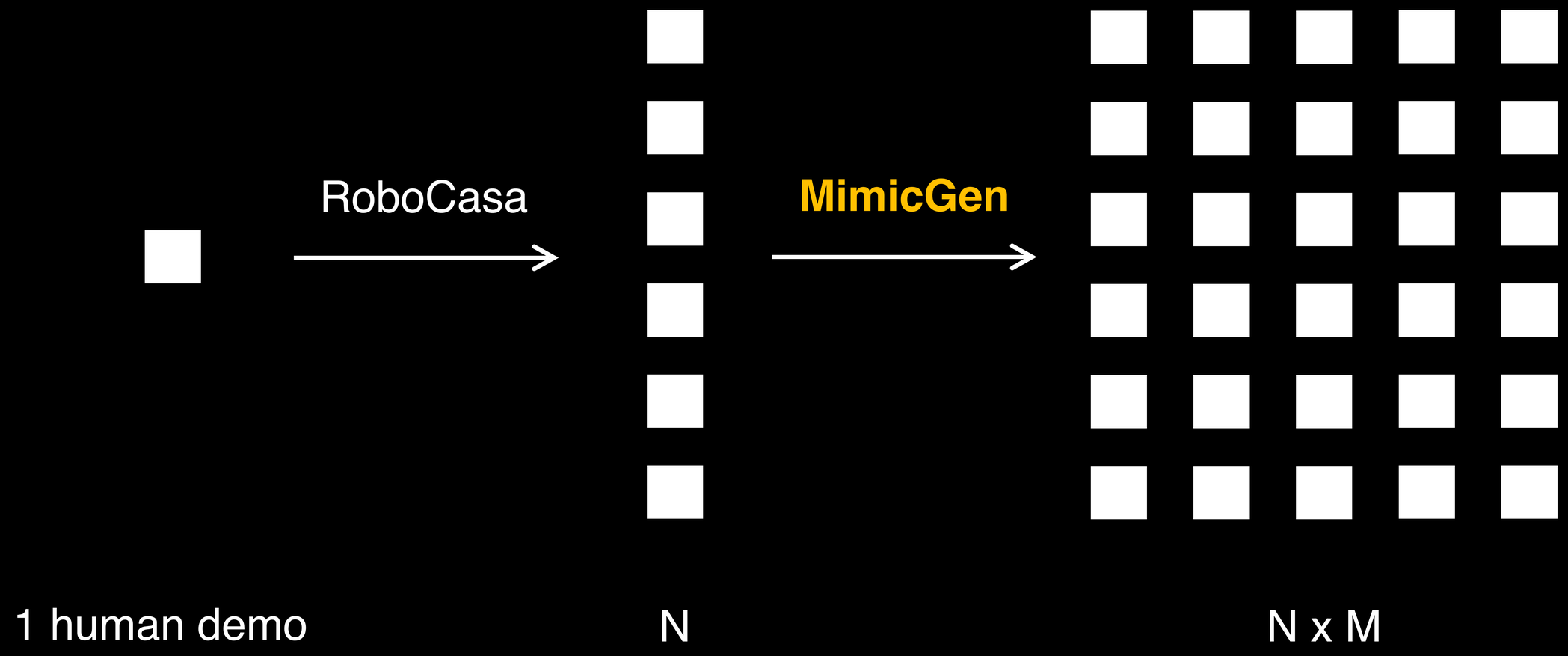



Success Rate
100
90
80
70
60
50
40
30
20
10
0
100 demos
500 demos
1000 demos
5000 demos

Transfer real demo to sim using digital twin to ensure the sim demos are valid in real

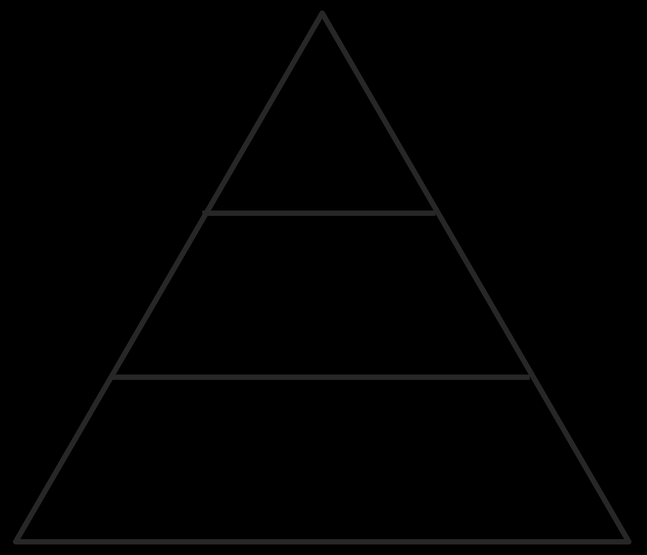
Transfer real demo to sim using digital twin to ensure the sim demos are valid in real

Transfer real demo to sim using digital twin to ensure the sim demos are valid in real

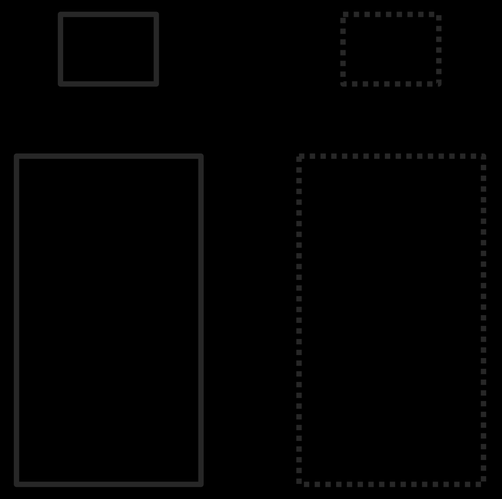

RoboCasa
MimicGen
1 human demo
N
N x M

The training data was collected
via human teleoperation

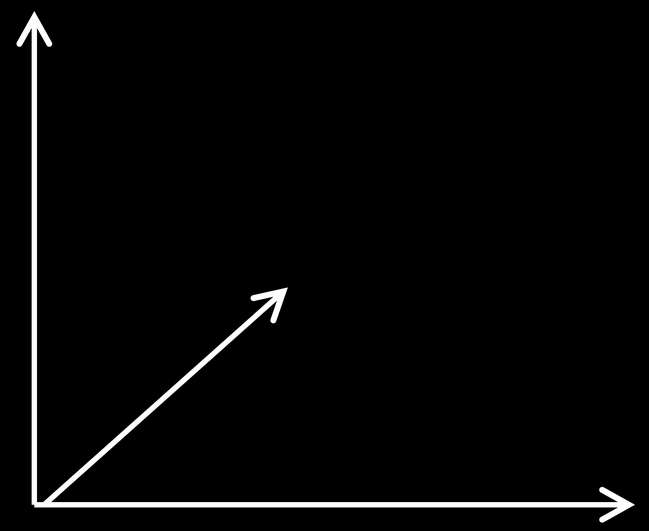
Principle No. 3



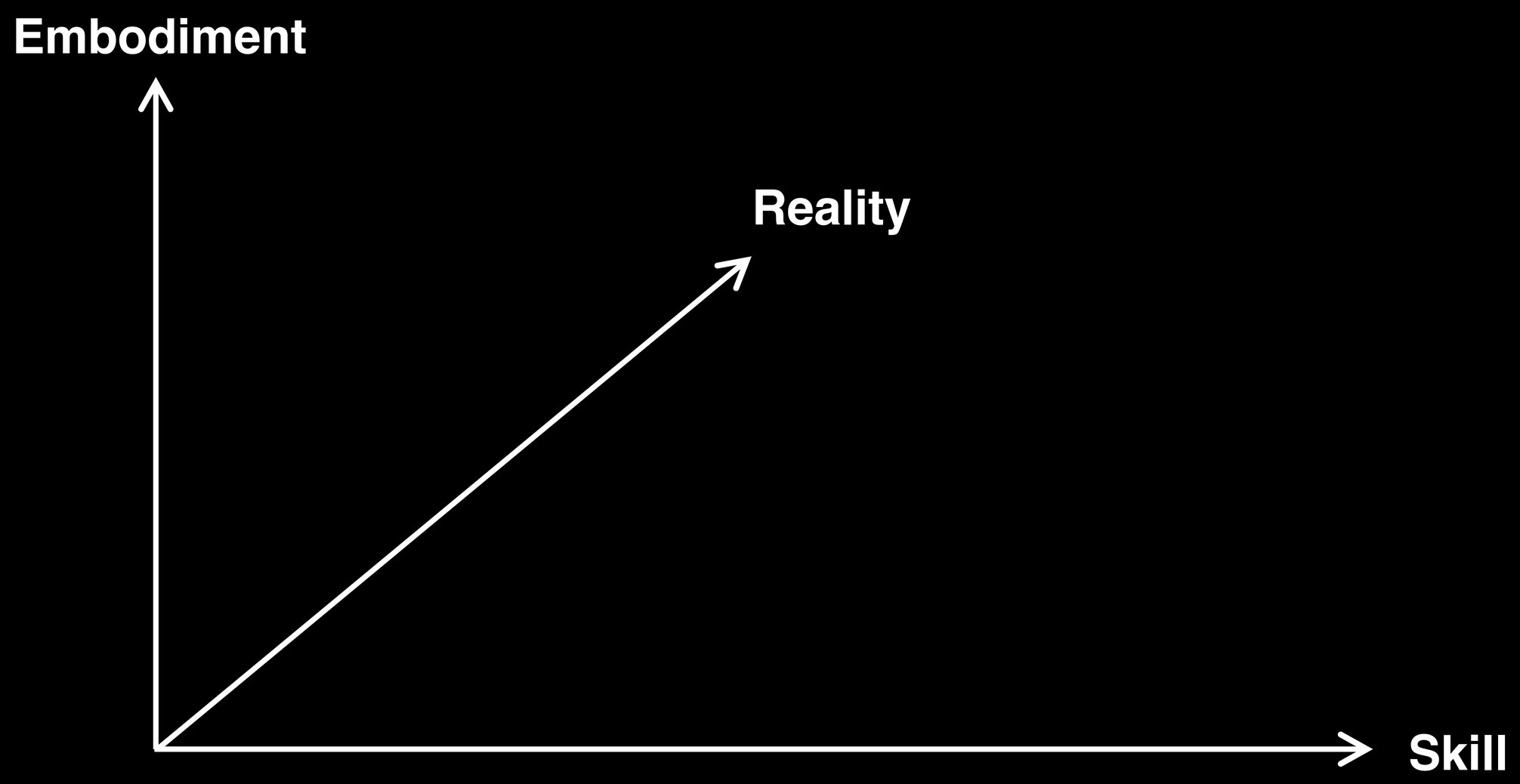
Data Pyramid



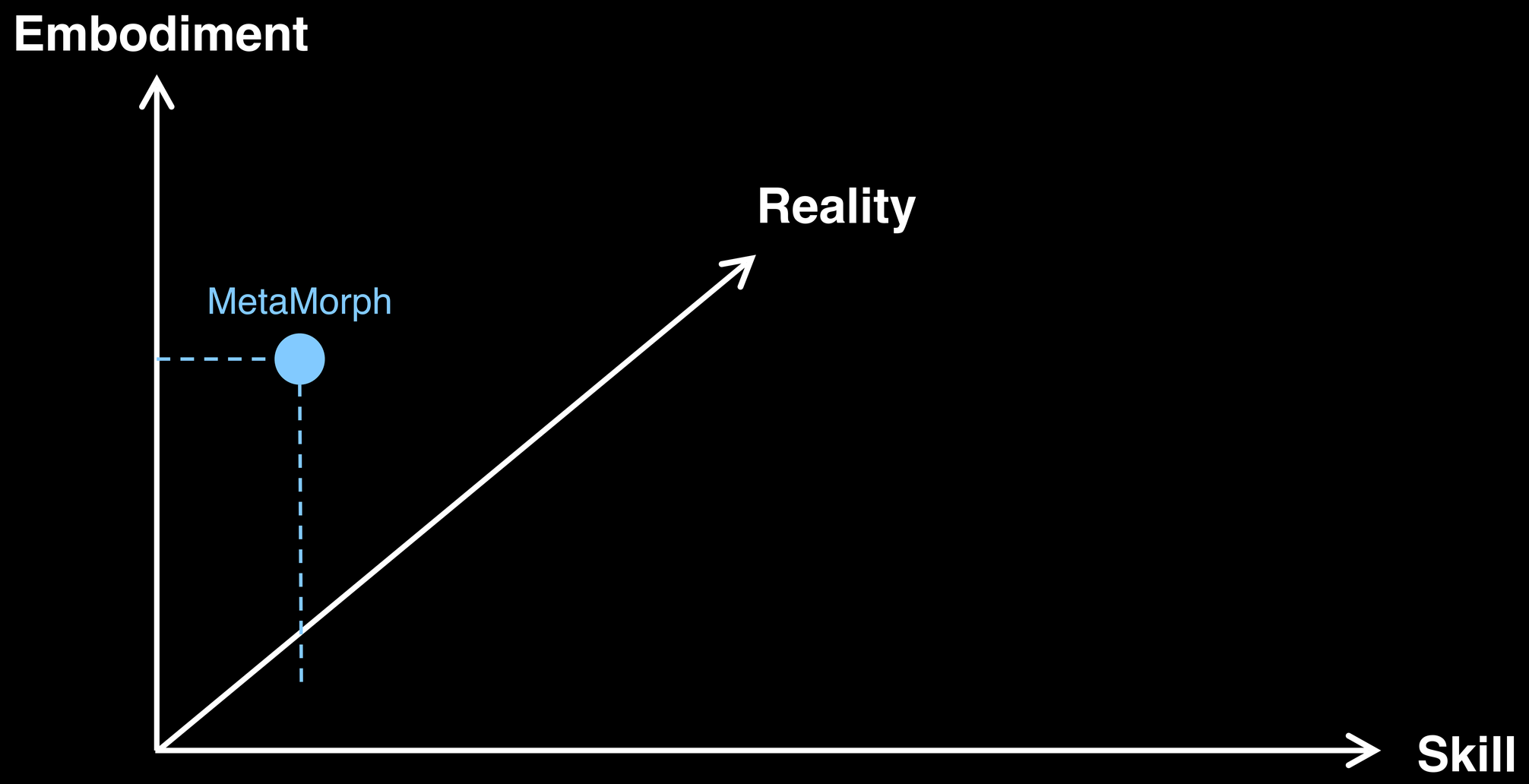
“The Matrix”

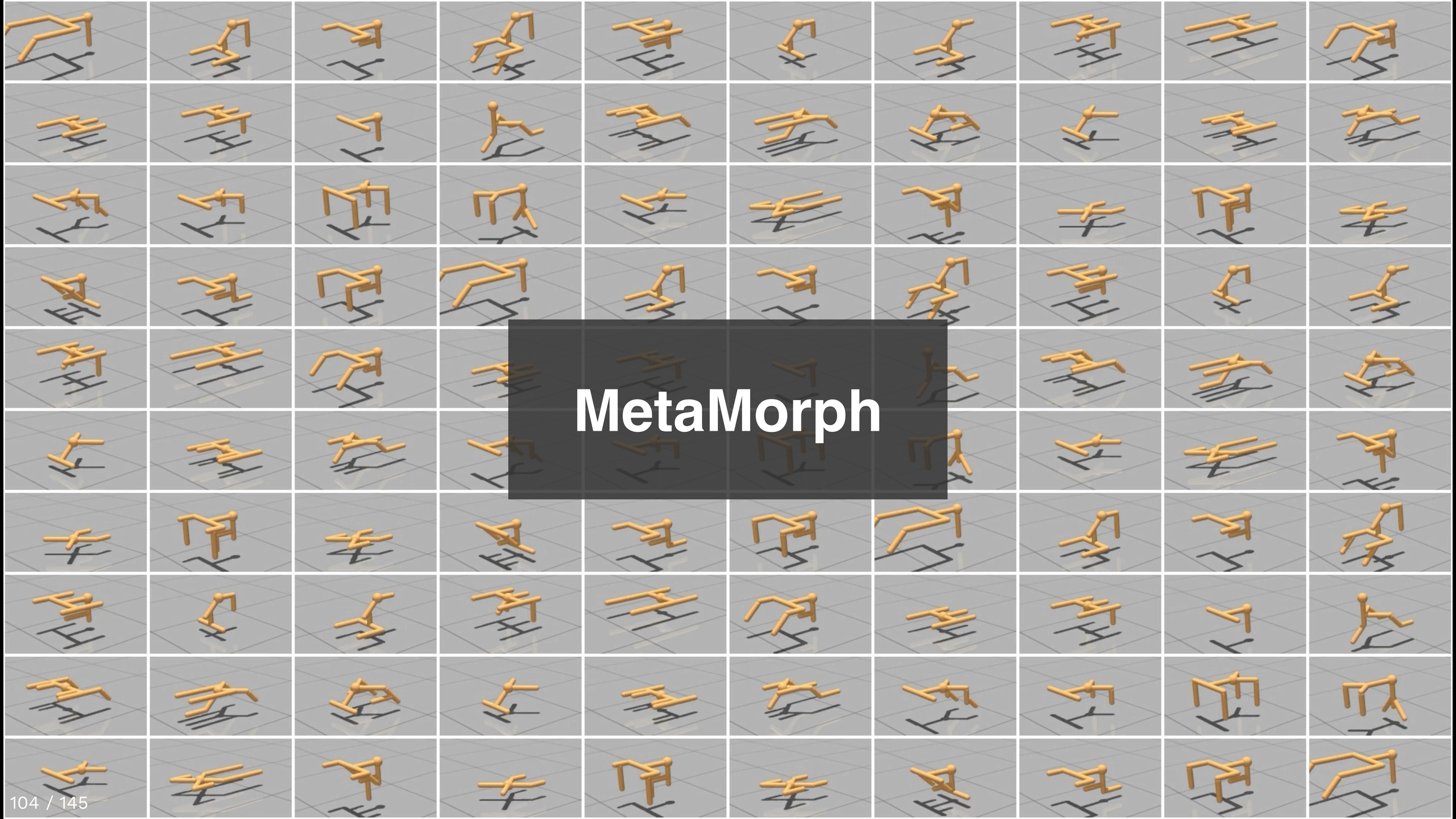


Foundation Agent

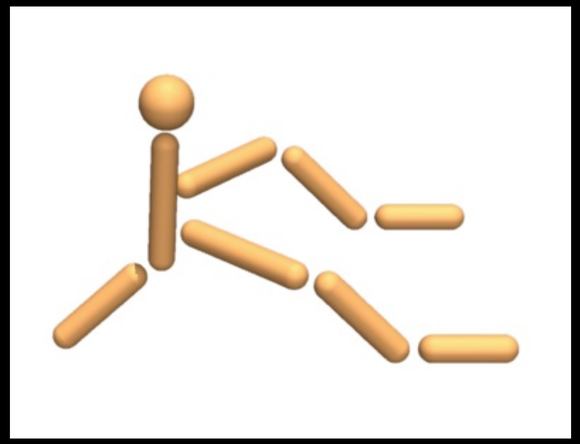

Embodiment
Reality
Skill

Diverse Robot Forms (“Embodiments”)


Embodiment
Reality
MetaMorph
Skill

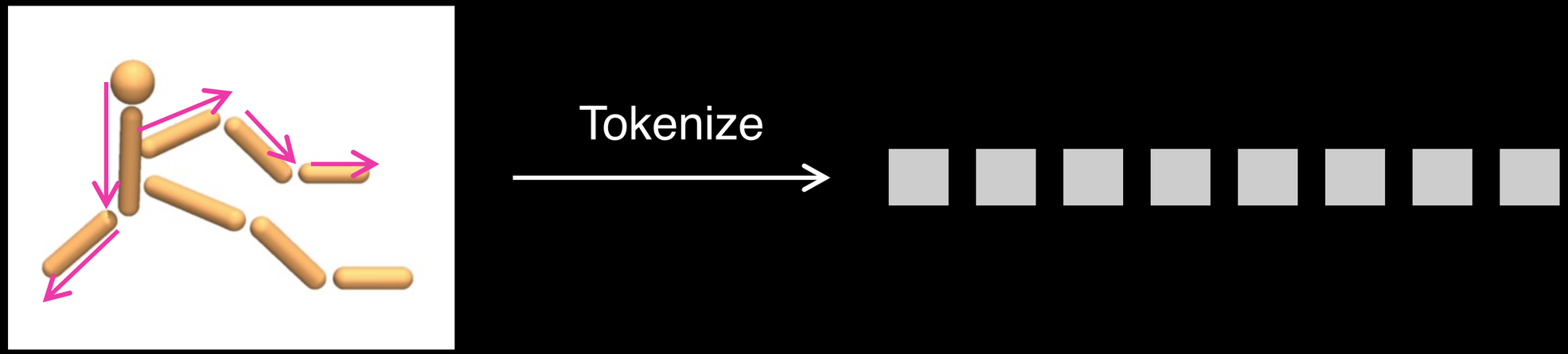


MetaMorph

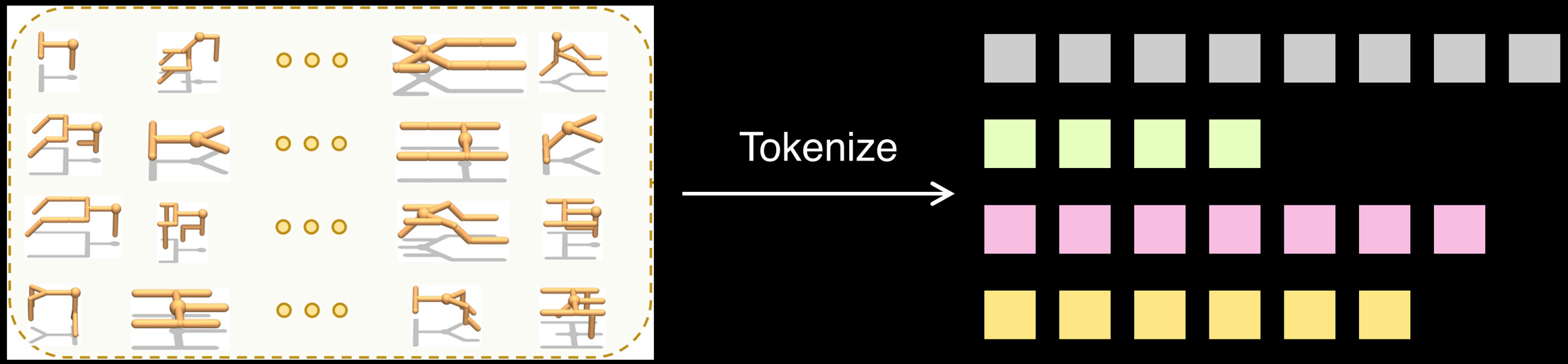


Graph of joints
“Kinematic tree”

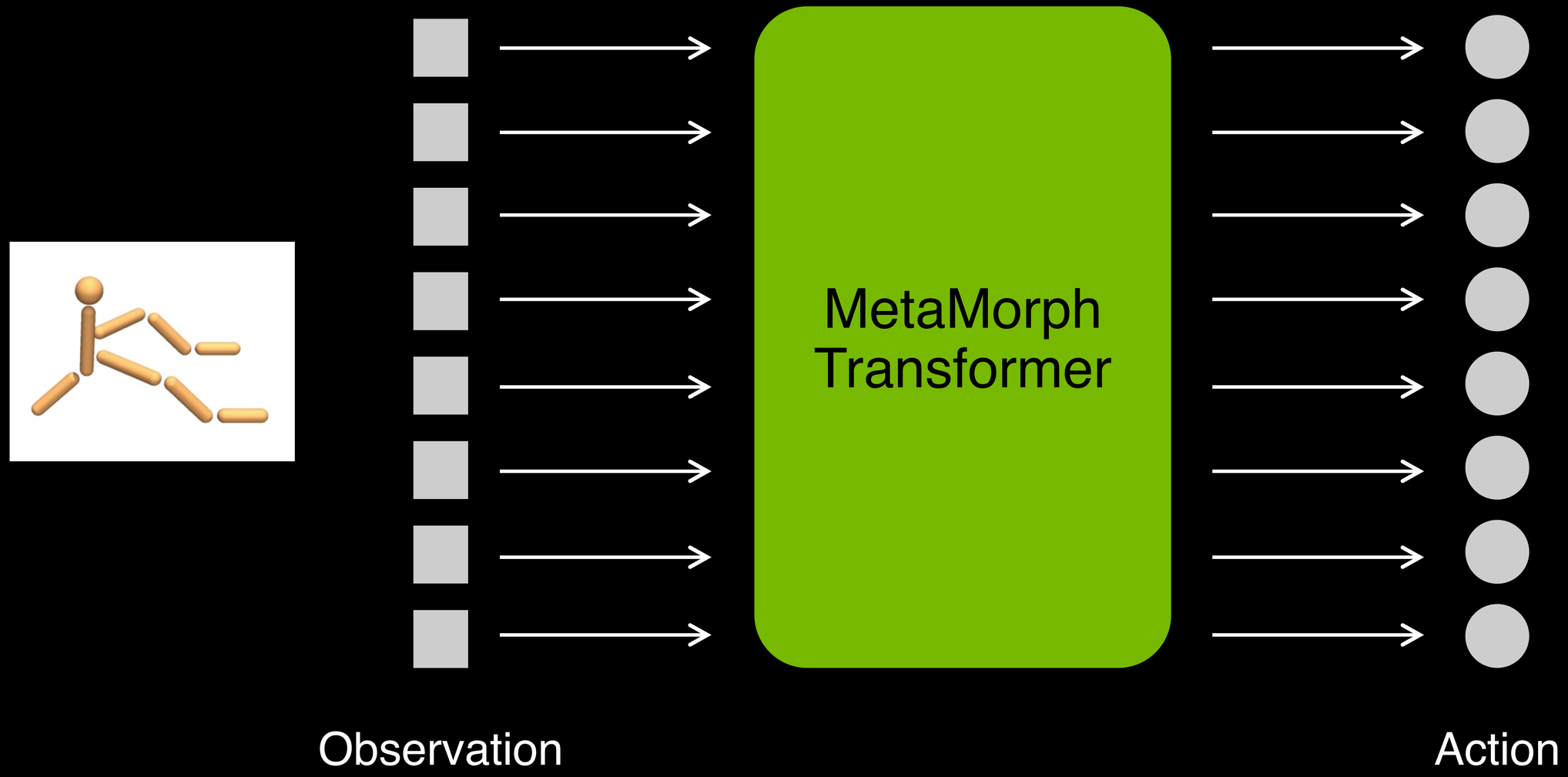
Gupta et al. ICLR 2022.

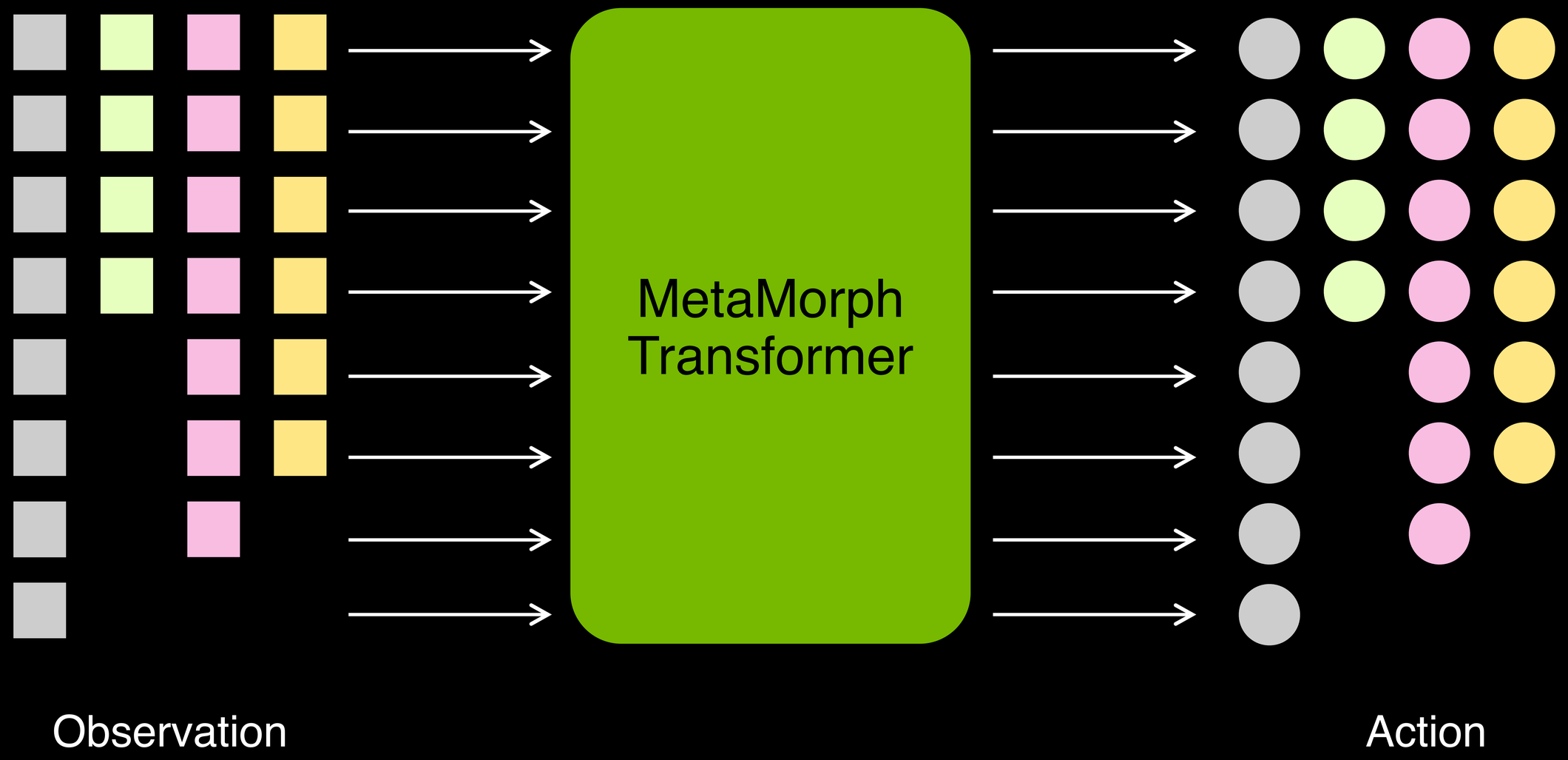


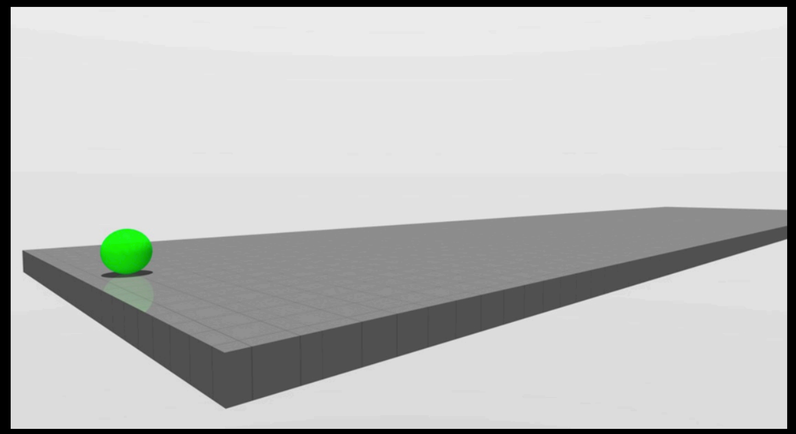
Gupta et al. ICLR 2022.



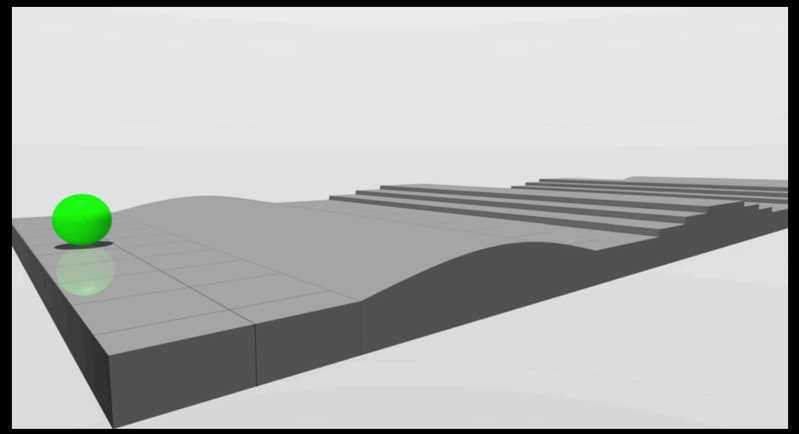
Gupta et al. ICLR 2022.


MetaMorph
Transformer
Observation
Action

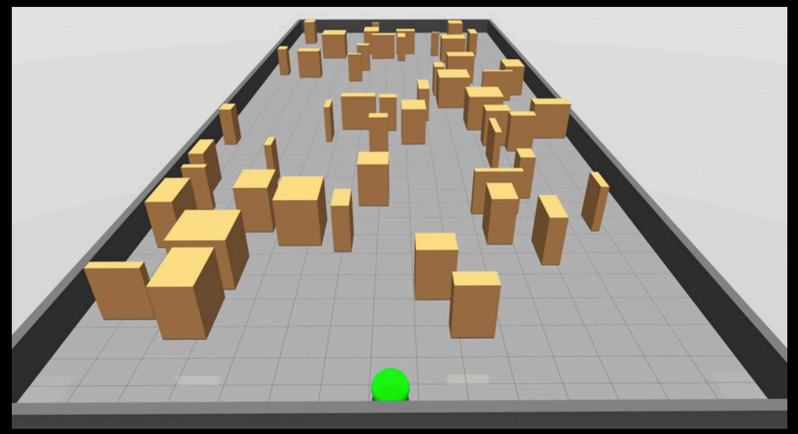

MetaMorph
Transformer
Observation
Action



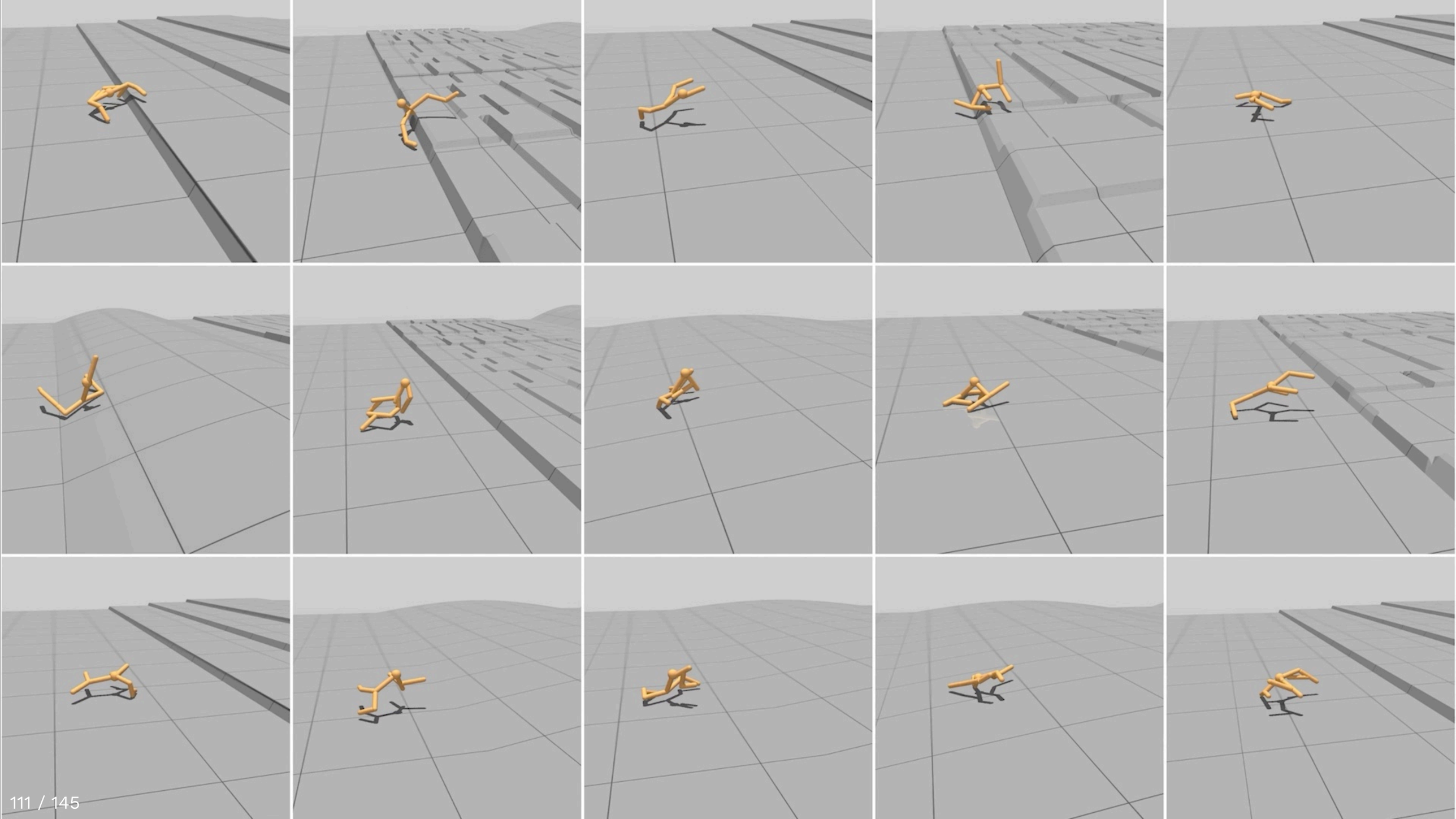
Flat Terrain

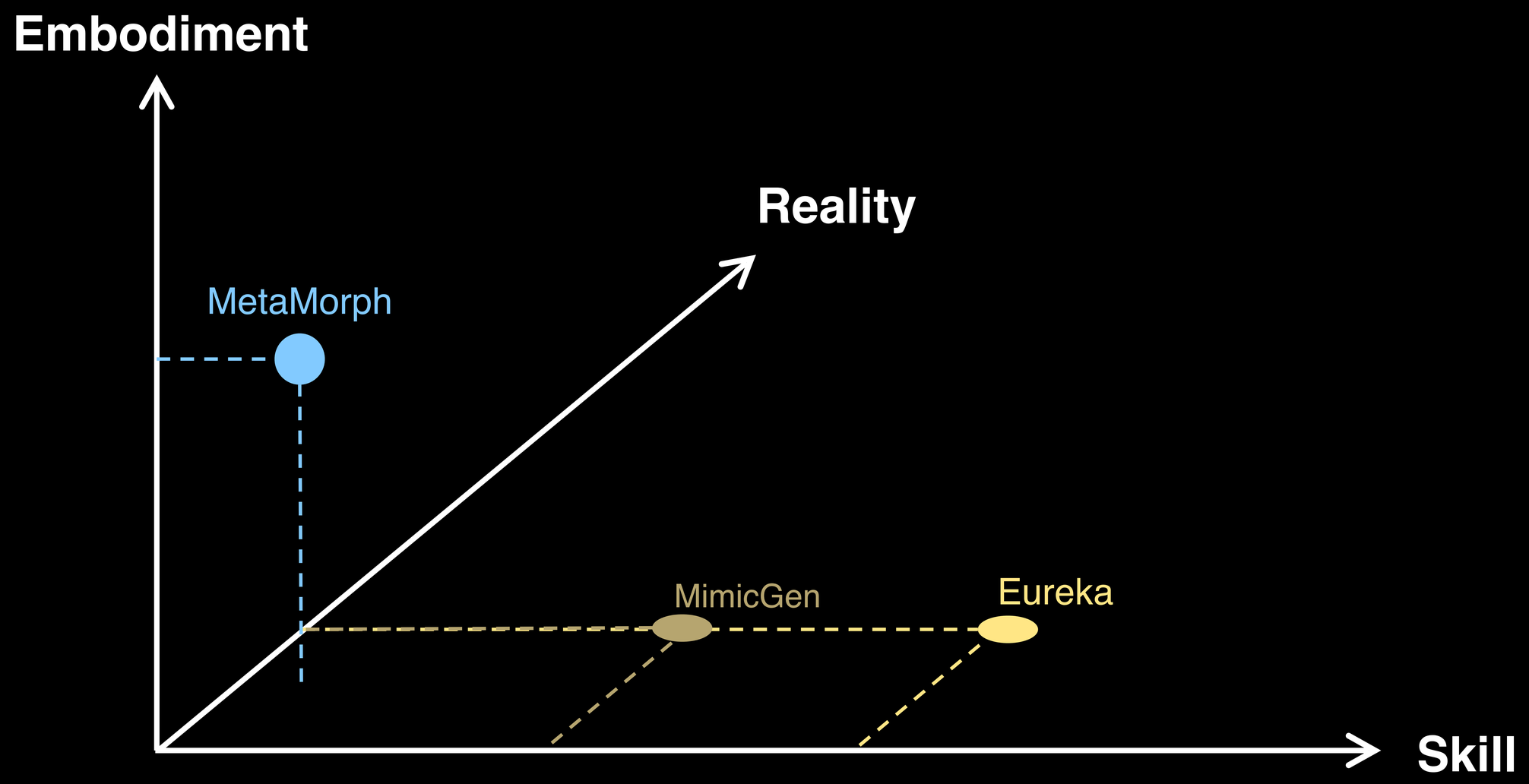


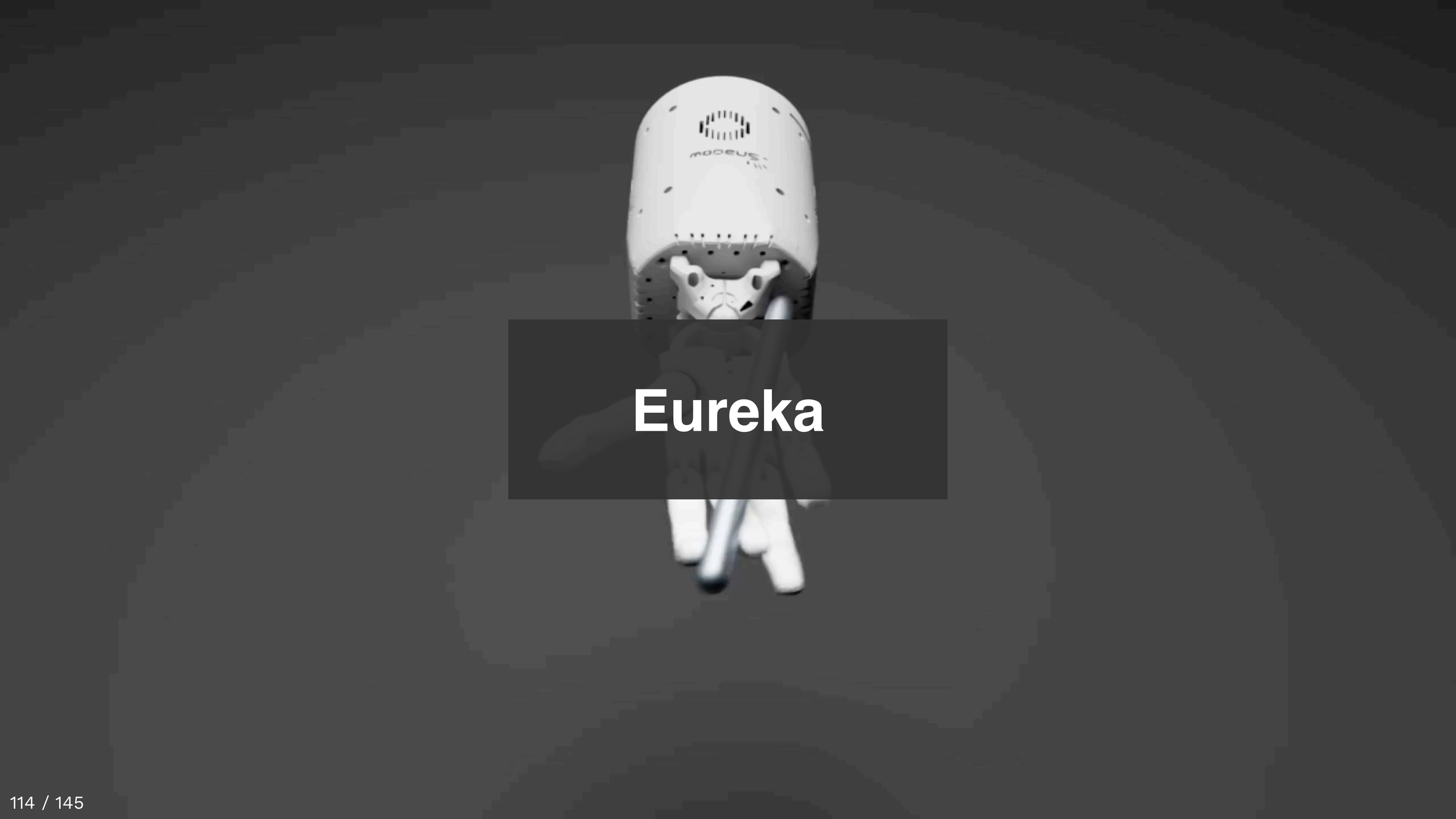
Complex Terrain



Obstacles

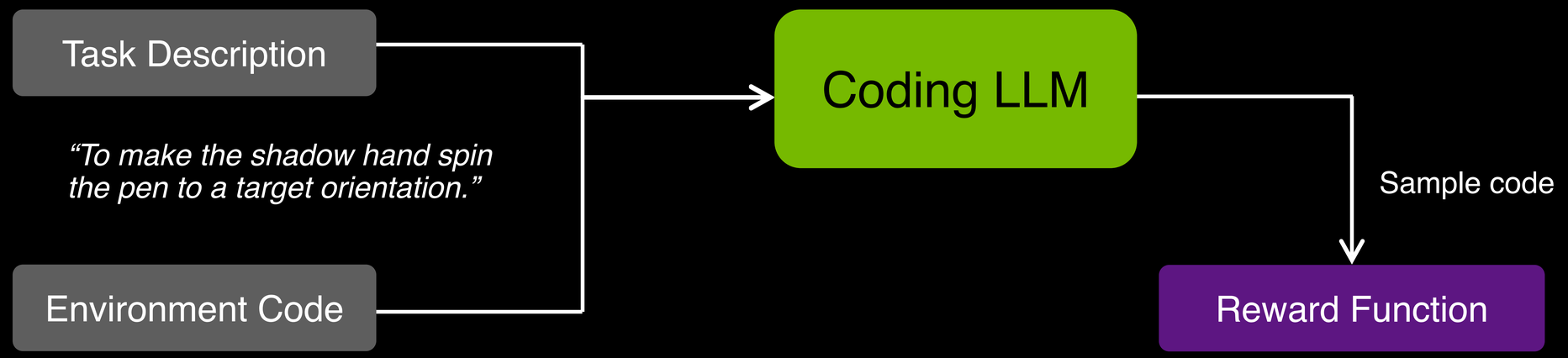



Embodiment
Reality
MetaMorph
MimicGen
Eureka
Skill


Eureka

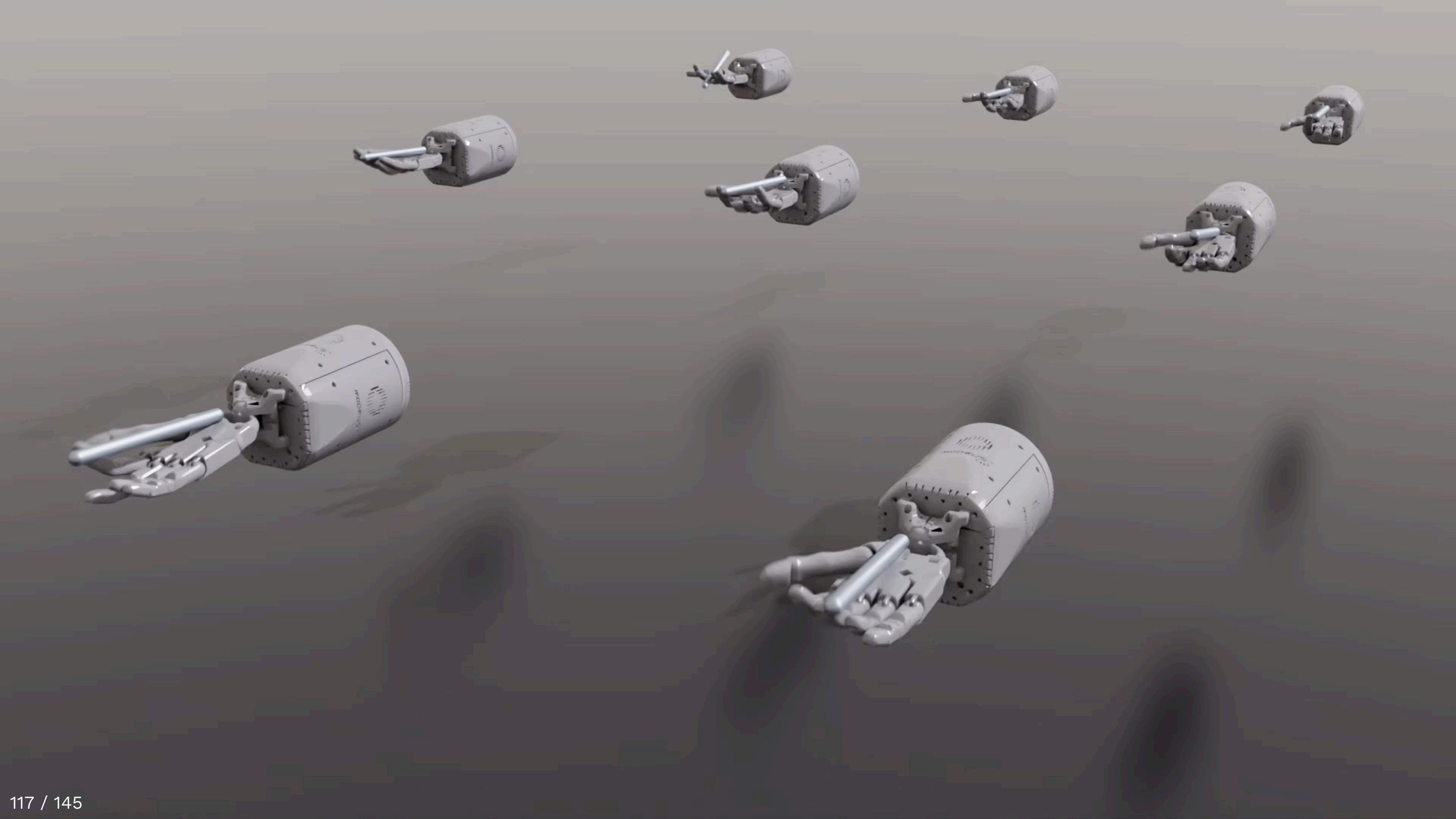
```python
class ShadowHandPenSpin(VecTask):
    def compute_observations(self):
        self.obj_pose = ...
        self.obj_pos = ...
        self.obj_rot = ...
        self.obj_linvel = ...
        self.obj_angvel = ...

        self.tgt_pose = ...
        self.tgt_pos = ...
        self.tgt_rot = ...

        self.fingertip_state = ...
        self.fingertip_pos = ...

        self.compute_full_state()

    def compute_full_state(self):
        ...
```

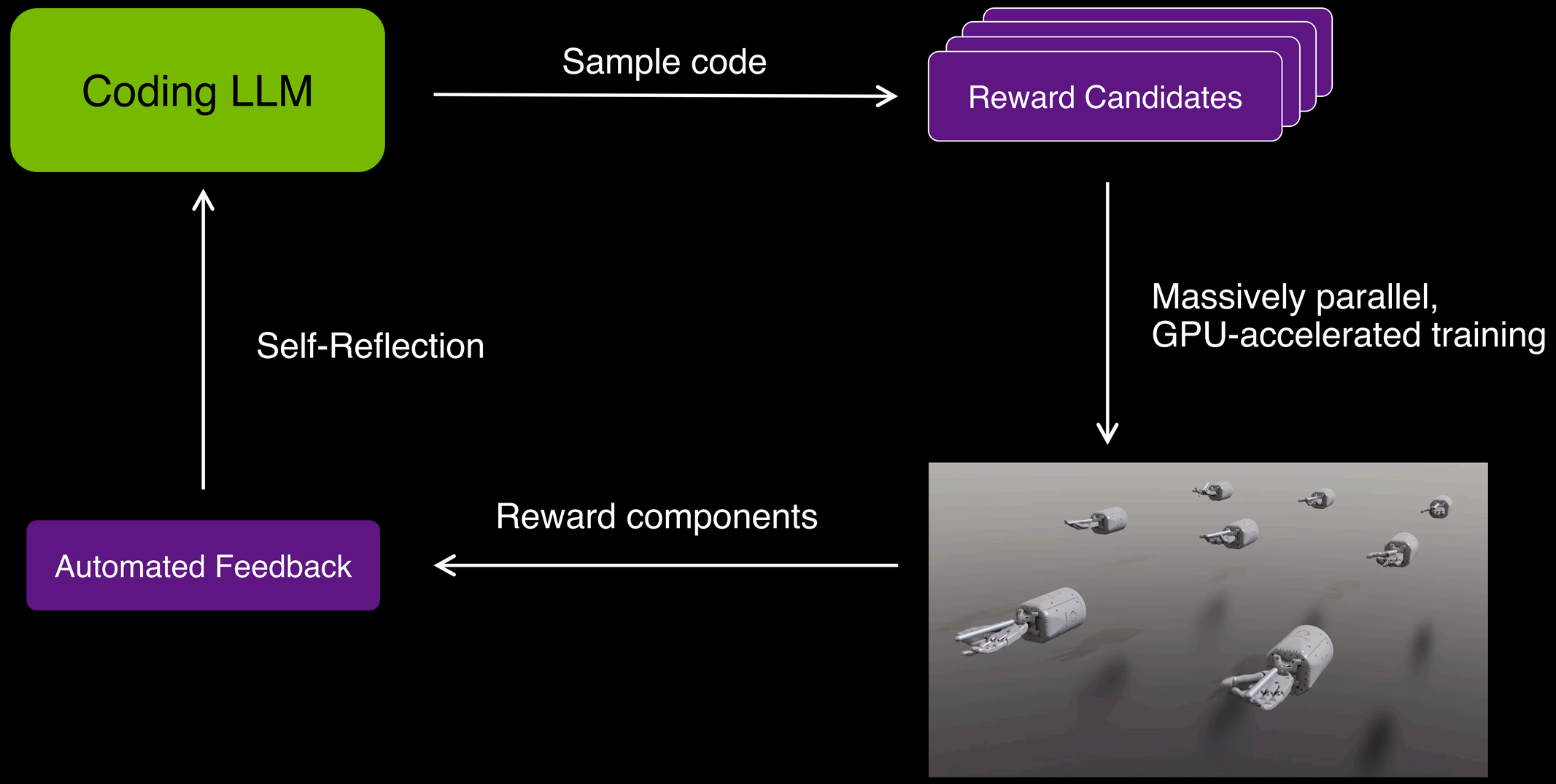
Ma et al. ICLR 2024.




```python
class ShadowHandPenSpin(VecTask):
    def compute_observations(self):
        self.obj_pose = ...
        self.obj_pos = ...
        self.obj_rot = ...
        self.obj_linvel = ...
        self.obj_angvel = ...

        self.tgt_pose = ...
        self.tgt_pos = ...
        self.tgt_rot = ...
```

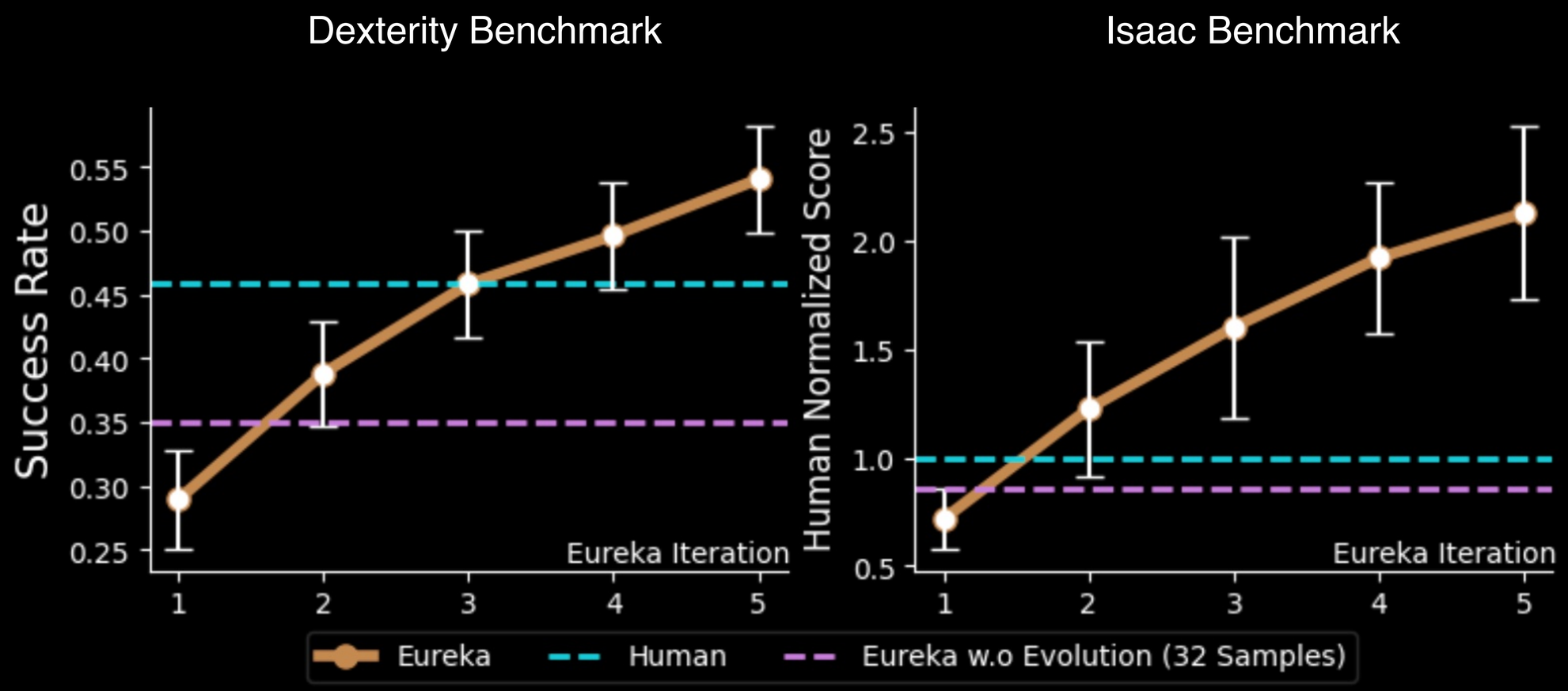
```python
def compute_reward(object_rot, goal_rot, object_angvel, object_pos, fingertip_pos):
    # Rotation reward
    rot_diff = torch.abs(torch.sum(object_rot * goal_rot, dim=1) - 1) / 2
    rotation_reward_temp = 20.0
    rotation_reward = torch.exp(-rotation_reward_temp * rot_diff)

    # Distance reward
    min_distance = torch.min(torch.norm(fingertip_pos - object_pos[:, None], dim=2), dim=1).values
    distance_reward = min_distance

    total_reward = rotation_reward + distance_reward

    reward_components = {
        "rotation_reward": rotation_reward,
        "distance_reward": distance_reward
    }

    return total_reward, reward_components
```

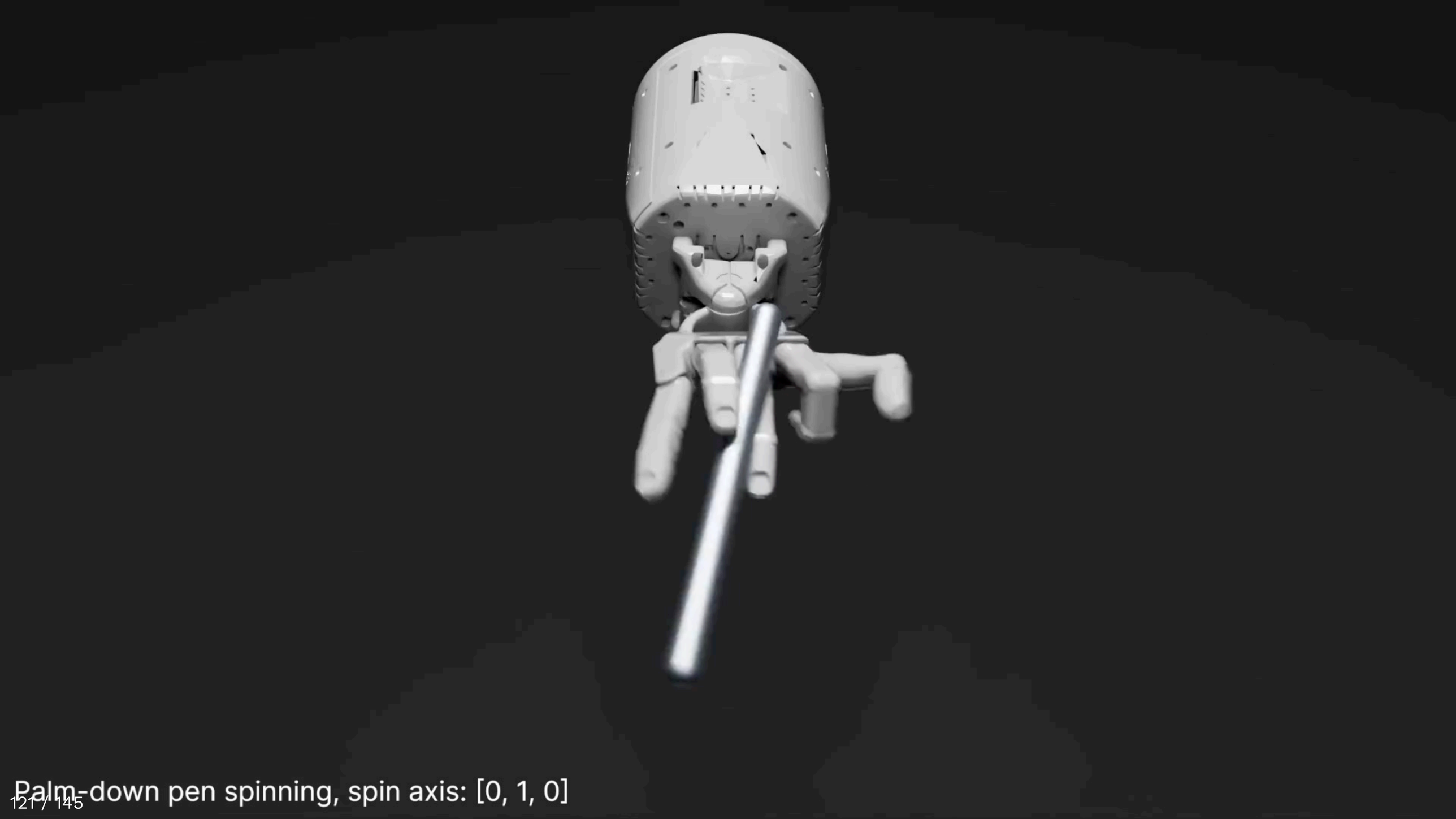




Ma et al. ICLR 2024.

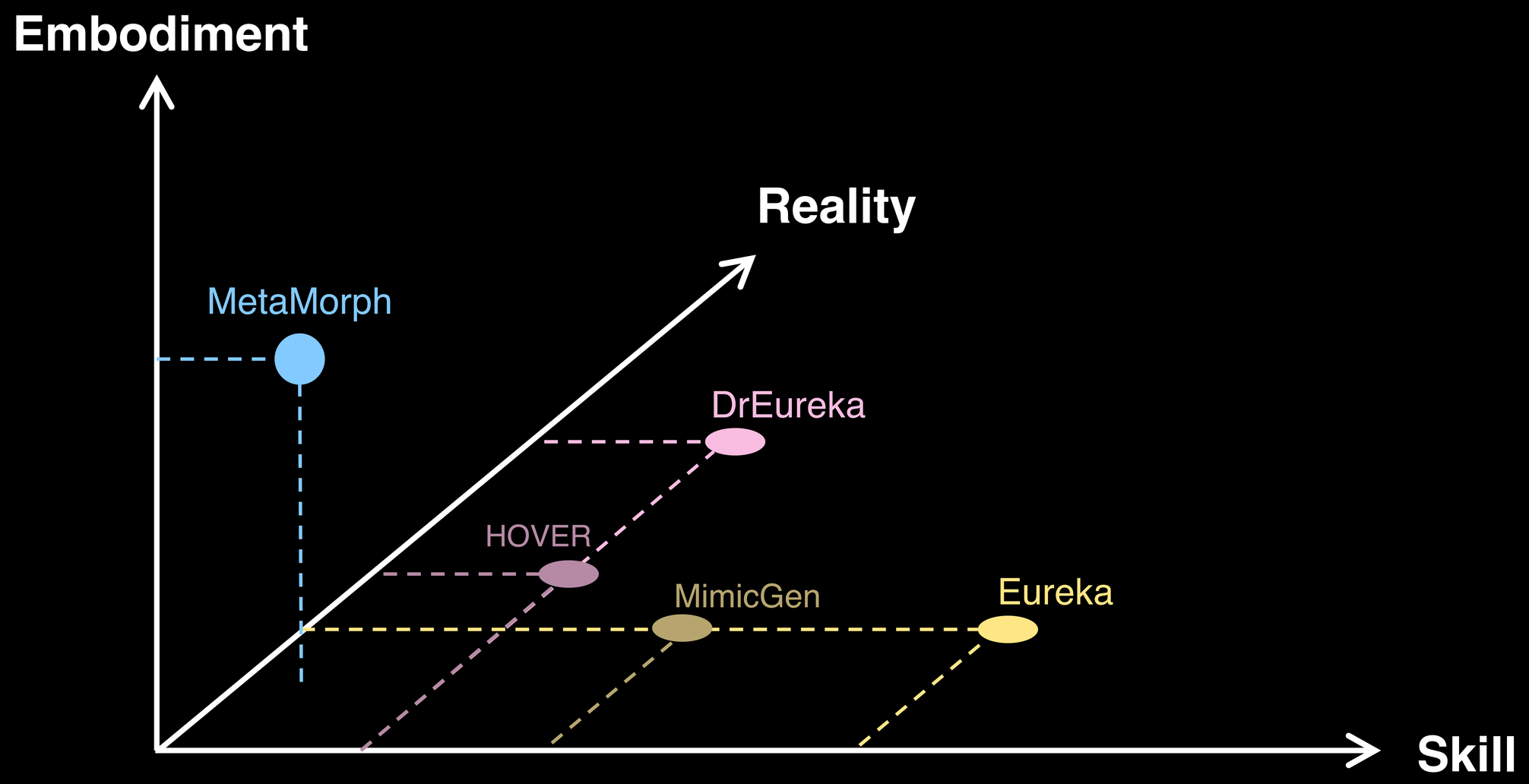
```
  def compute_reward(object_rot, goal_rot, object_angvel, object_pos, fingertip_pos):
      # Rotation reward
      rot_diff = torch.abs(torch.sum(object_rot * goal_rot, dim=1) - 1) / 2
-     rotation_reward_temp = 20.0
+     rotation_reward_temp = 30.0                                          Changing hyperparameter
      rotation_reward = torch.exp(-rotation_reward_temp * rot_diff)

      # Distance reward
+     min_distance_temp = 10.0
      min_distance = torch.min(torch.norm(fingertip_pos - object_pos[:, None], dim=2), dim=1).values
-     distance_reward = min_distance
+     uncapped_distance_reward = torch.exp(-min_distance_temp * min_distance)
+     distance_reward = torch.clamp(uncapped_distance_reward, 0.0, 1.0)    Changing functional form

-     total_reward = rotation_reward + distance_reward
+     # Angular velocity penalty                                           Adding new component
+     angvel_norm = torch.norm(object_angvel, dim=1)
+     angvel_threshold = 0.5
+     angvel_penalty_temp = 5.0
+     angular_velocity_penalty = torch.where(angvel_norm > angvel_threshold,
+         torch.exp(-angvel_penalty_temp * (angvel_norm - angvel_threshold)), torch.zeros_like(angvel_norm))
+
+     total_reward = 0.5 * rotation_reward + 0.3 * distance_reward - 0.2 * angular_velocity_penalty

      reward_components = {
          "rotation_reward": rotation_reward,
          "distance_reward": distance_reward,
+         "angular_velocity_penalty": angular_velocity_penalty,
      }

      return total_reward, reward_components
```

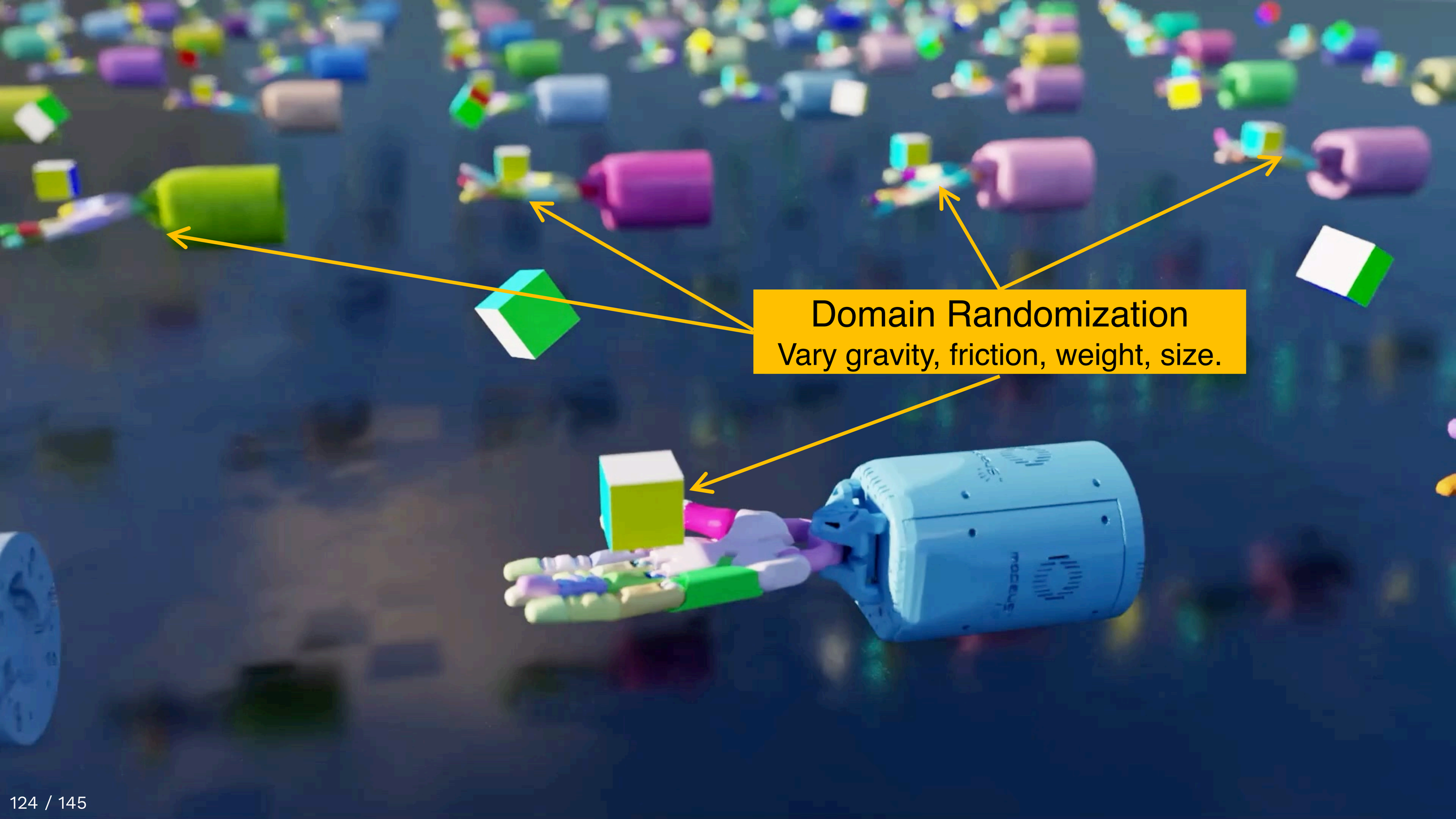

Dexterity Benchmark
Isaac Benchmark
Success Rate
0.55
0.50
0.45
0.40
0.35
0.30
0.25
Eureka Iteration
1
2
3
4
5
Human Normalized Score
2.5
2.0
1.5
1.0
0.5
Eureka Iteration
1
2
3
4
5
Eureka
Human
Eureka w.o Evolution (32 Samples)

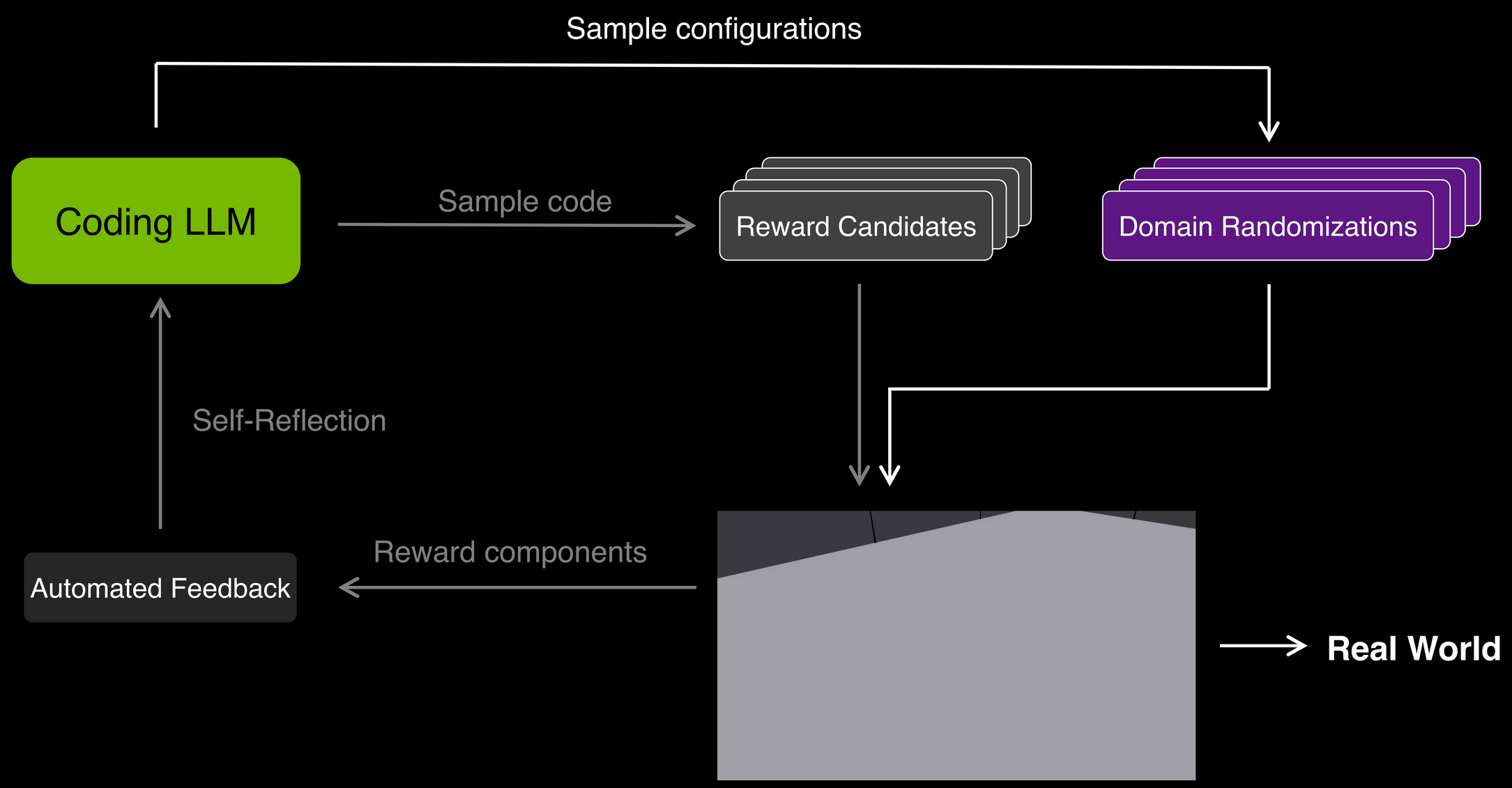


Palm-down pen spinning, spin axis: [0, 1, 0]

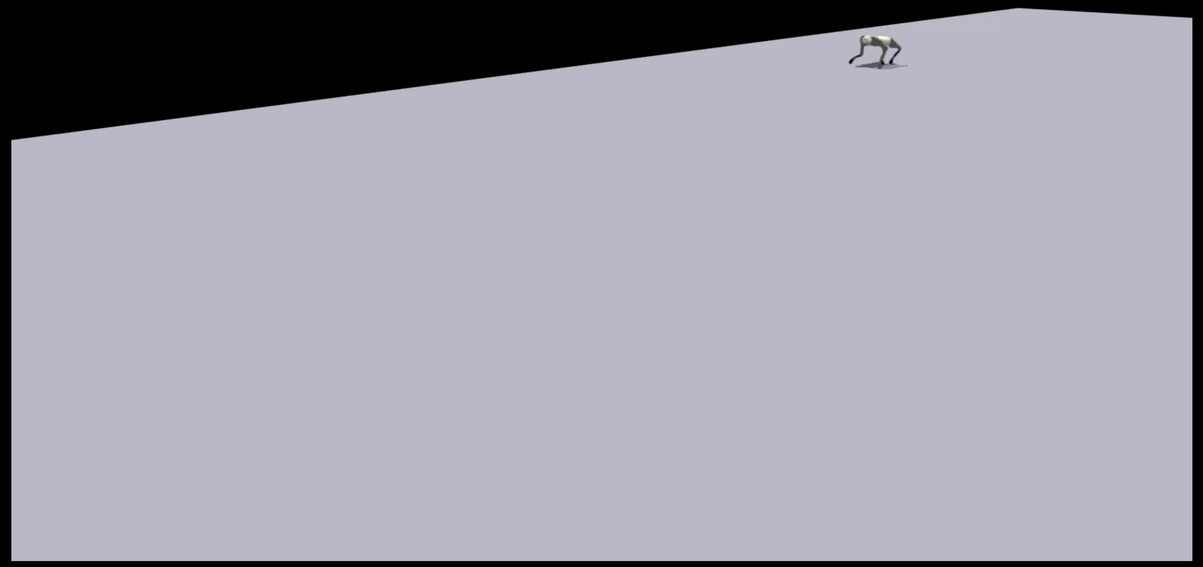
How to transfer from sim to the real world?


Embodiment
Reality
Skill
MetaMorph
DrEureka
HOVER
MimicGen
Eureka


Domain Randomization
Vary gravity, friction, weight, size.


Sample configurations
Coding LLM
Sample code
Reward Candidates
Domain Randomizations
Self-Reflection
Automated Feedback
Reward components
Real World

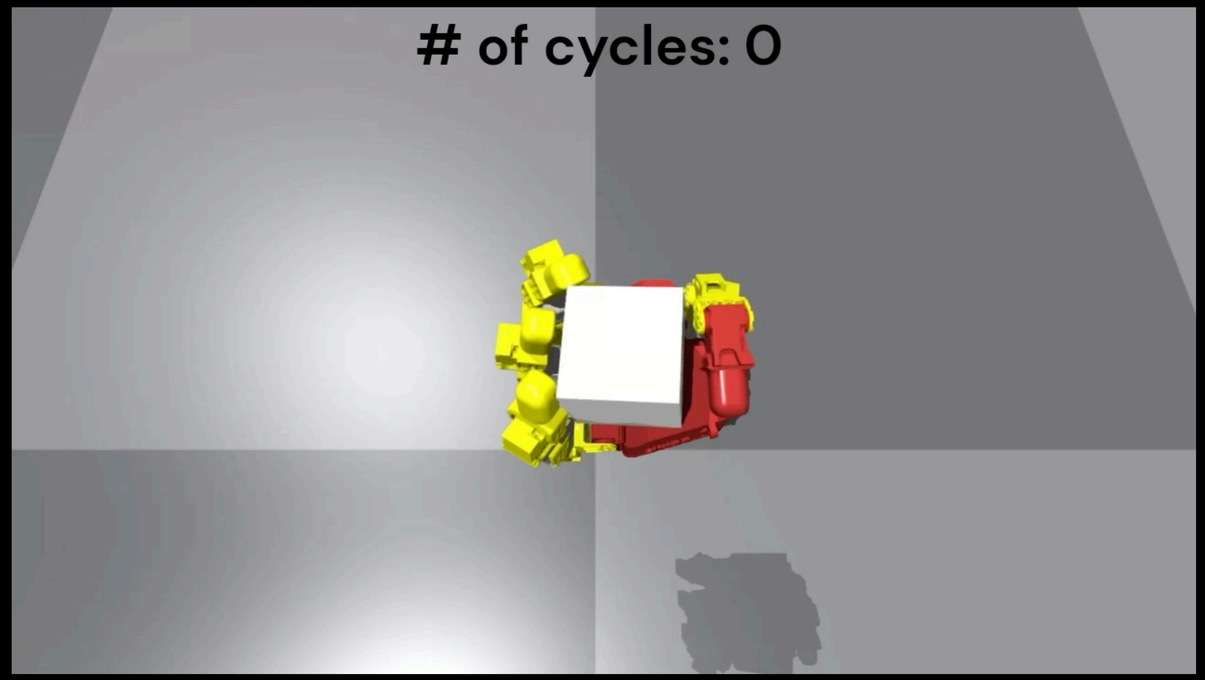
Ma et al, 2024



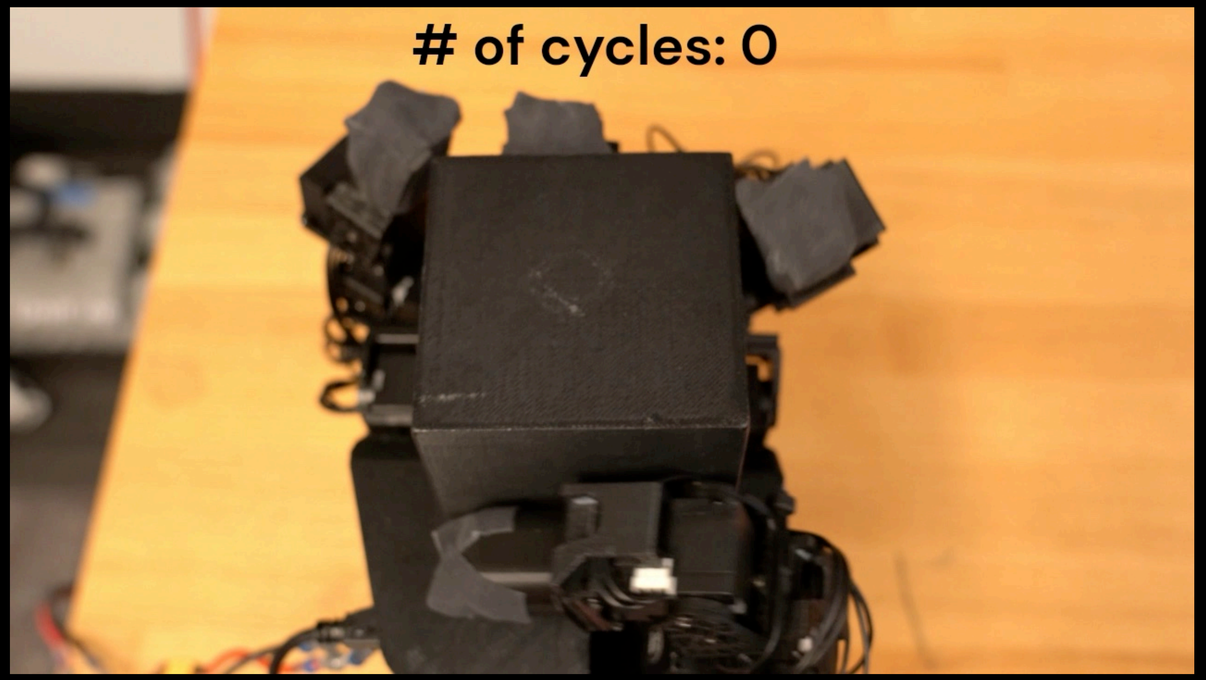
Simulation

Real World

Ma et al, 2023



Simulation

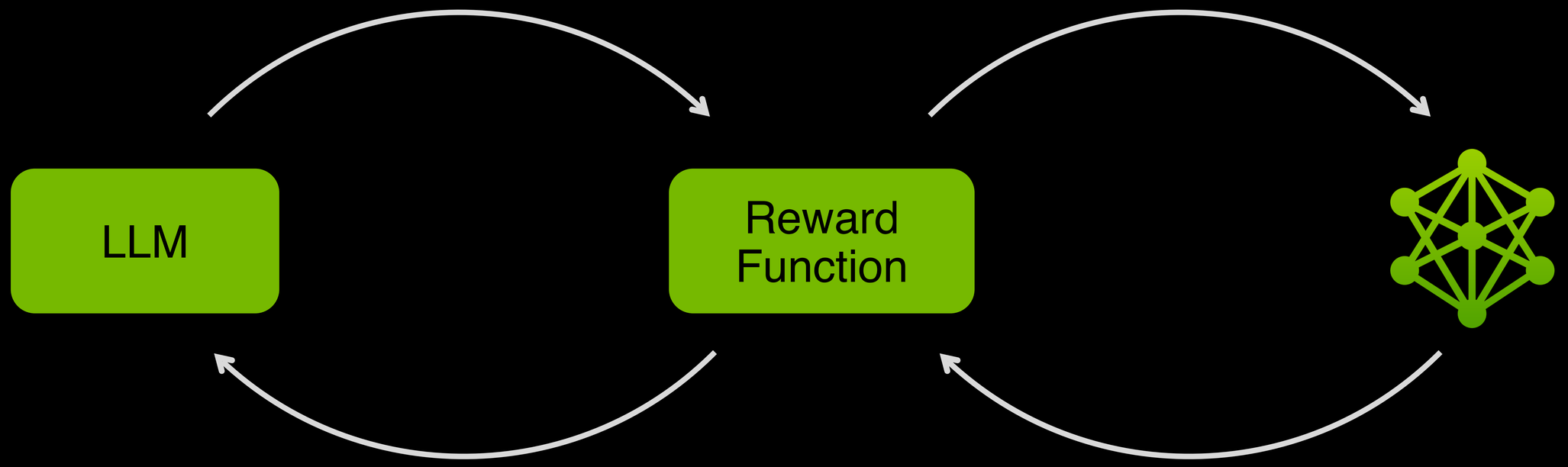


Real World

Ma et al, 2023

DrEureka

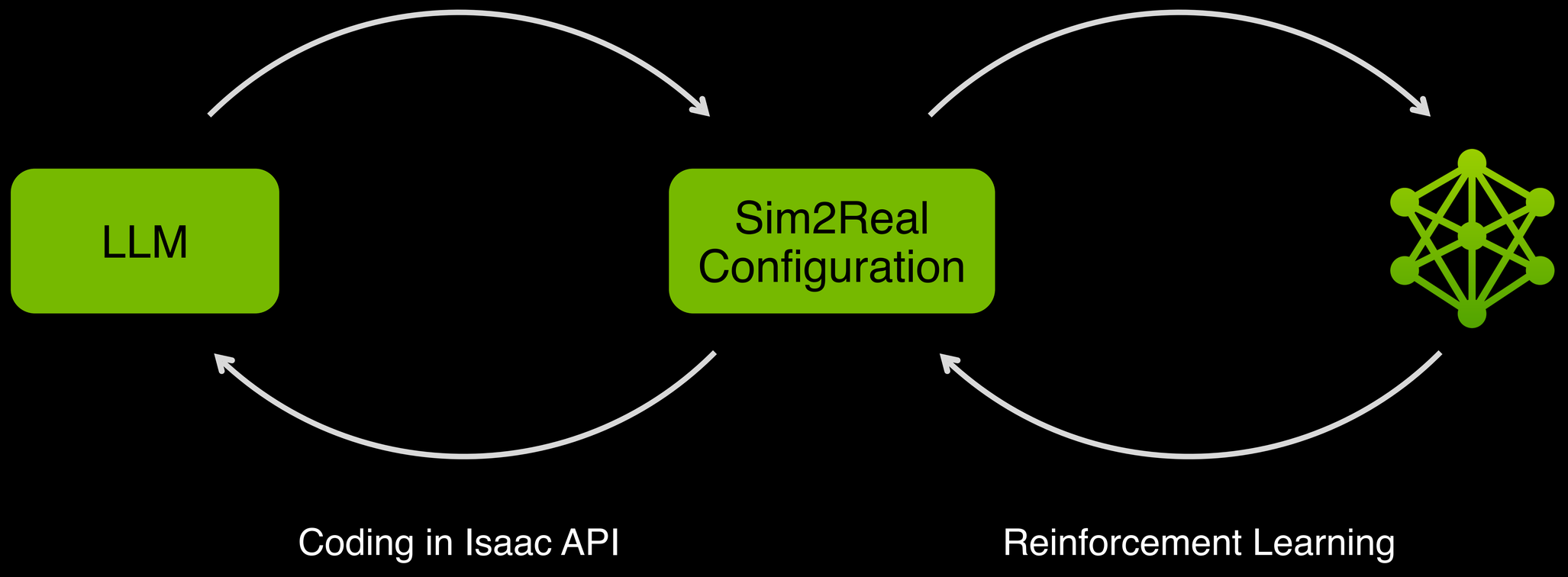
Eureka



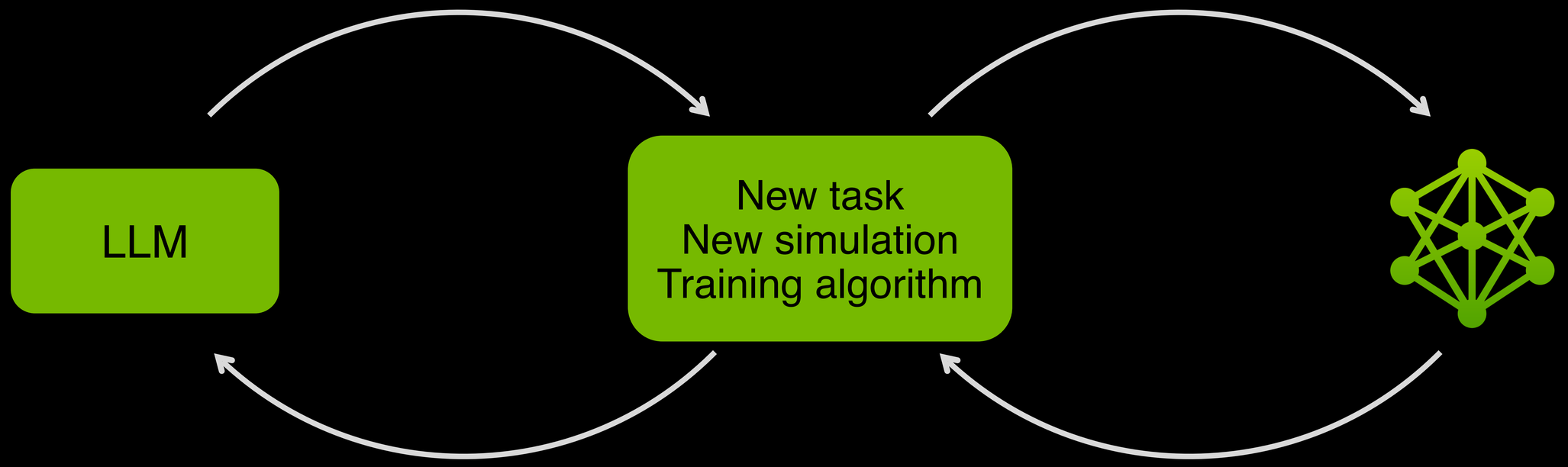
Coding in Isaac API

Reinforcement Learning

DrEureka

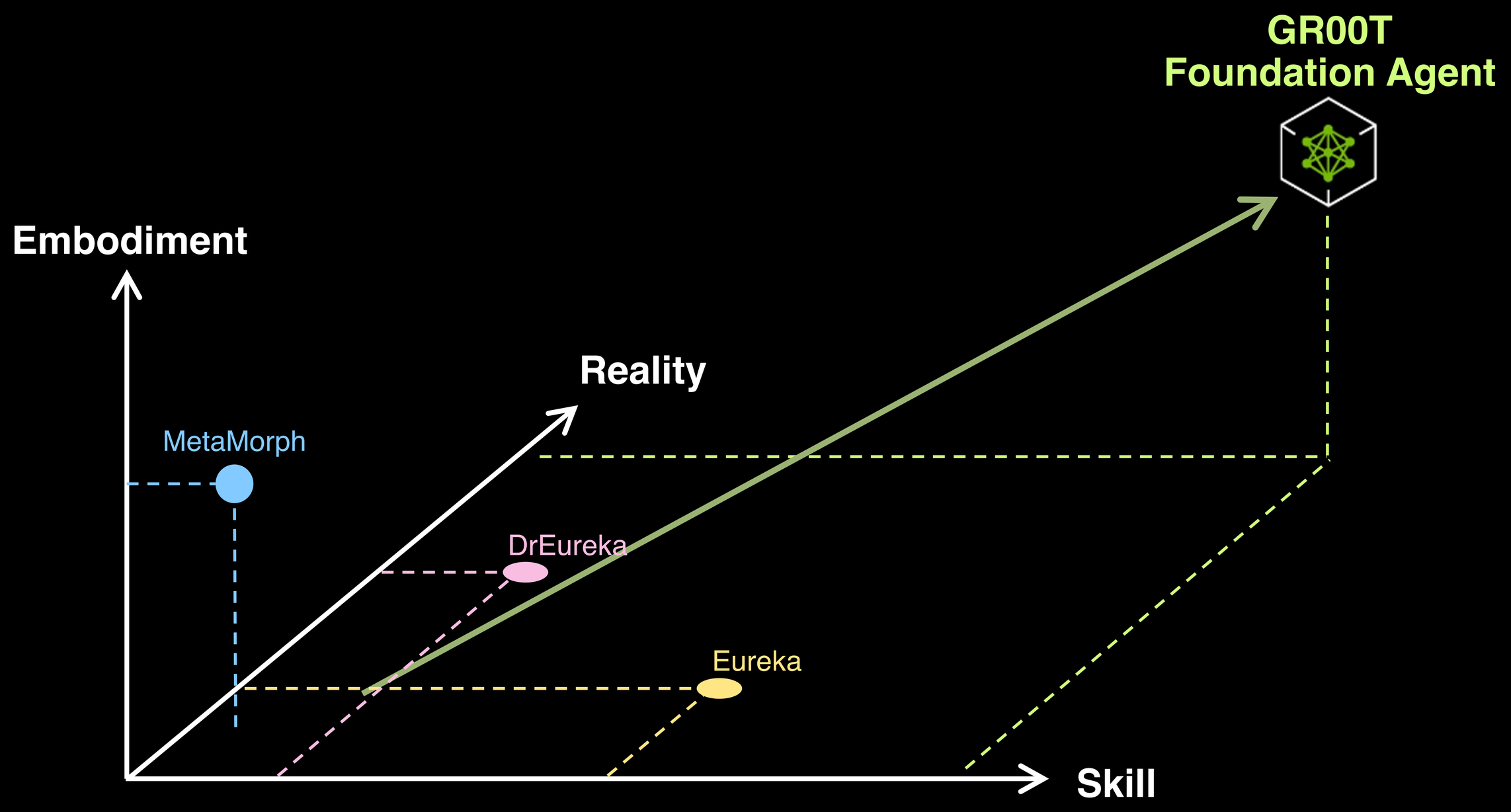

LLM
Sim2Real Configuration
Coding in Isaac API
Reinforcement Learning

Eureka++



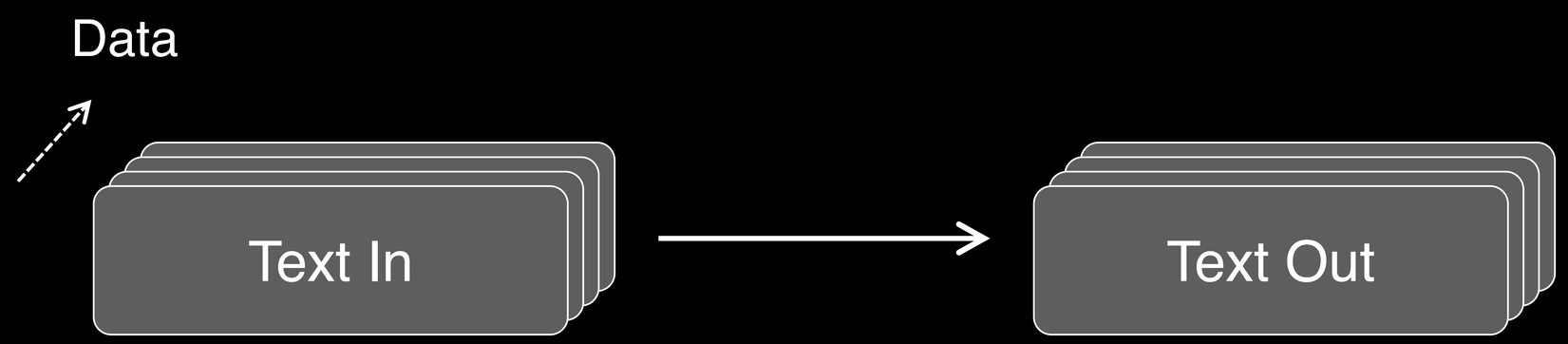
Coding in Isaac API

Reinforcement Learning


GR00T
Foundation Agent
Embodiment
Reality
MetaMorph
DrEureka
Eureka
Skill

ChatGPT

ChatGPT



GR00T

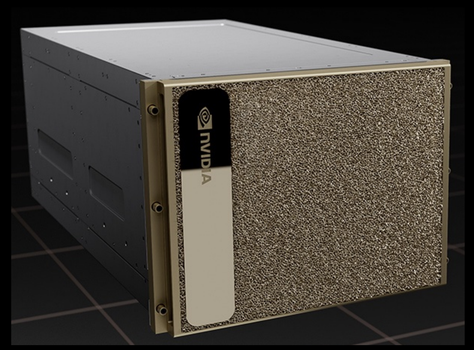
GR00T

OVX: Generate Tokens
RoboCasa, MimicGen, Eureka

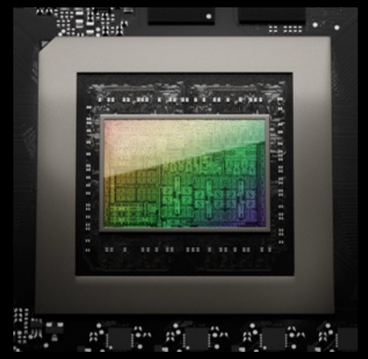
Synthetic data

Digital Twin Eval

OSMO



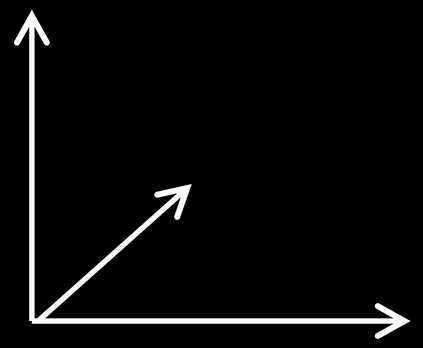
Edge inference



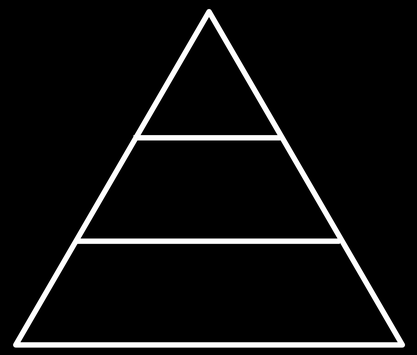
DGX: Learn Tokens
Foundation Agent Models

AGX: Deploy Tokens
HOVER, DrEureka

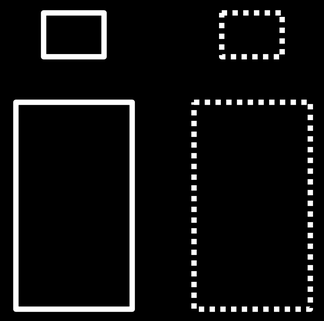
3 Principles



Foundation Agent



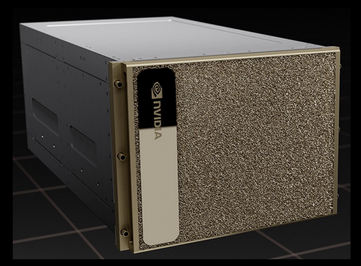
Data Pyramid



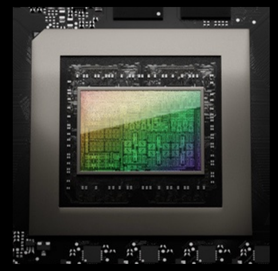
“The Matrix”

3 Computers

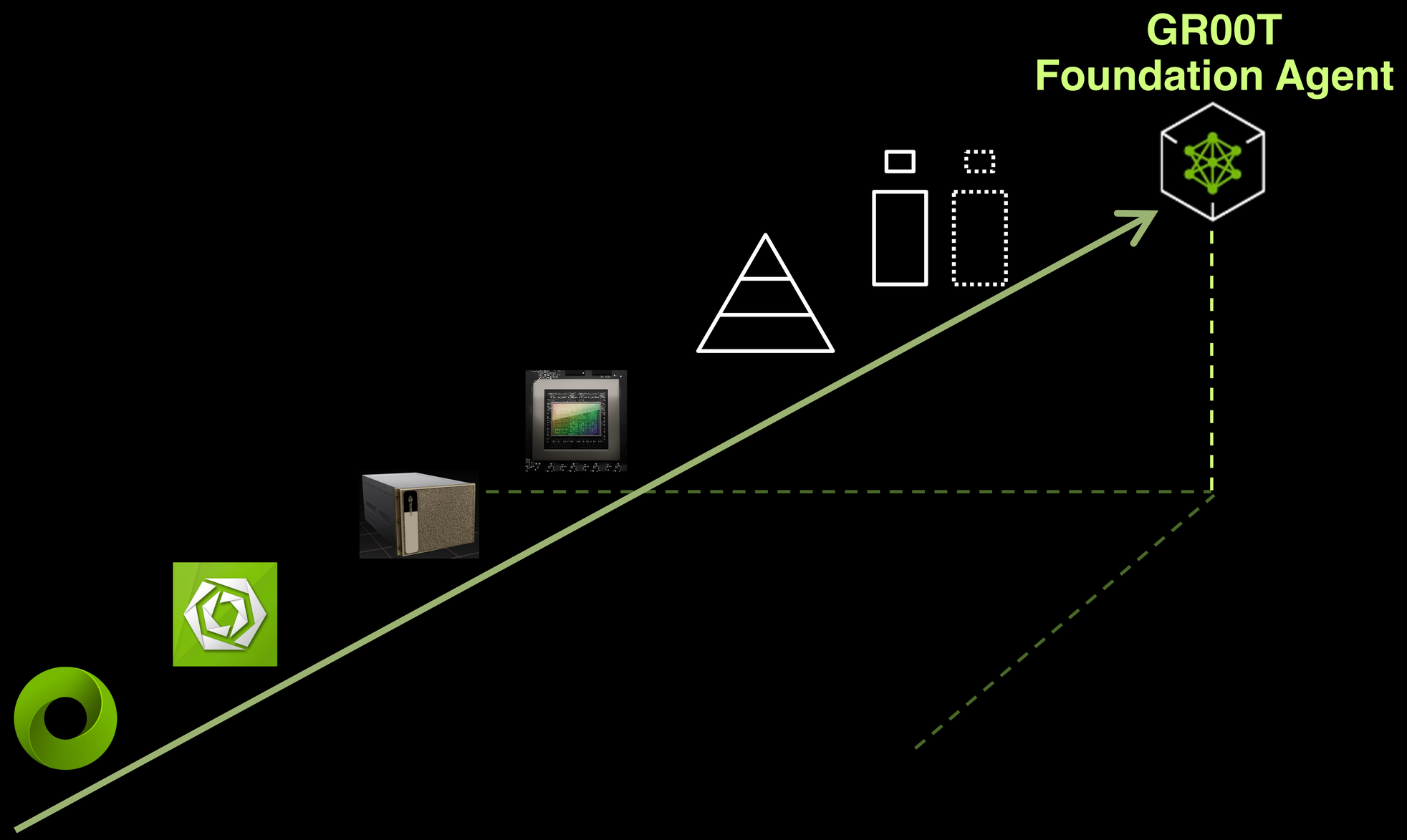
OVX

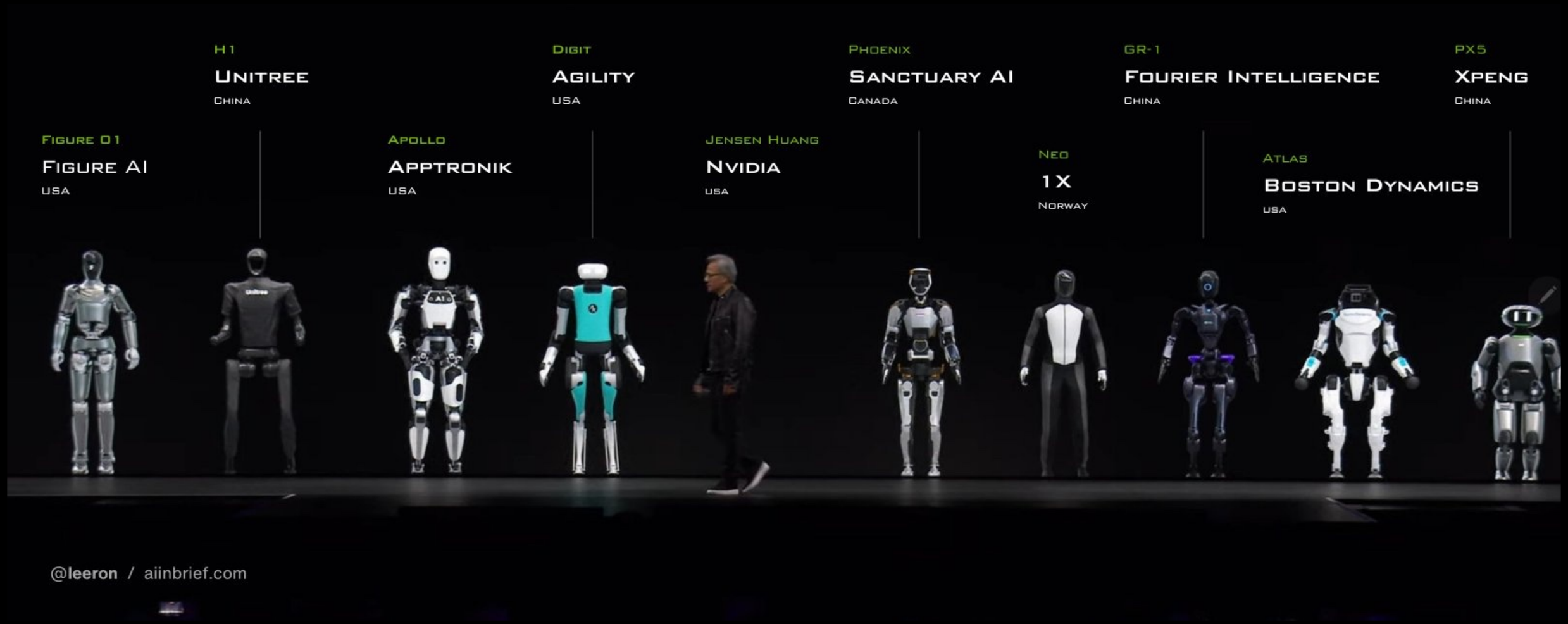


DGX

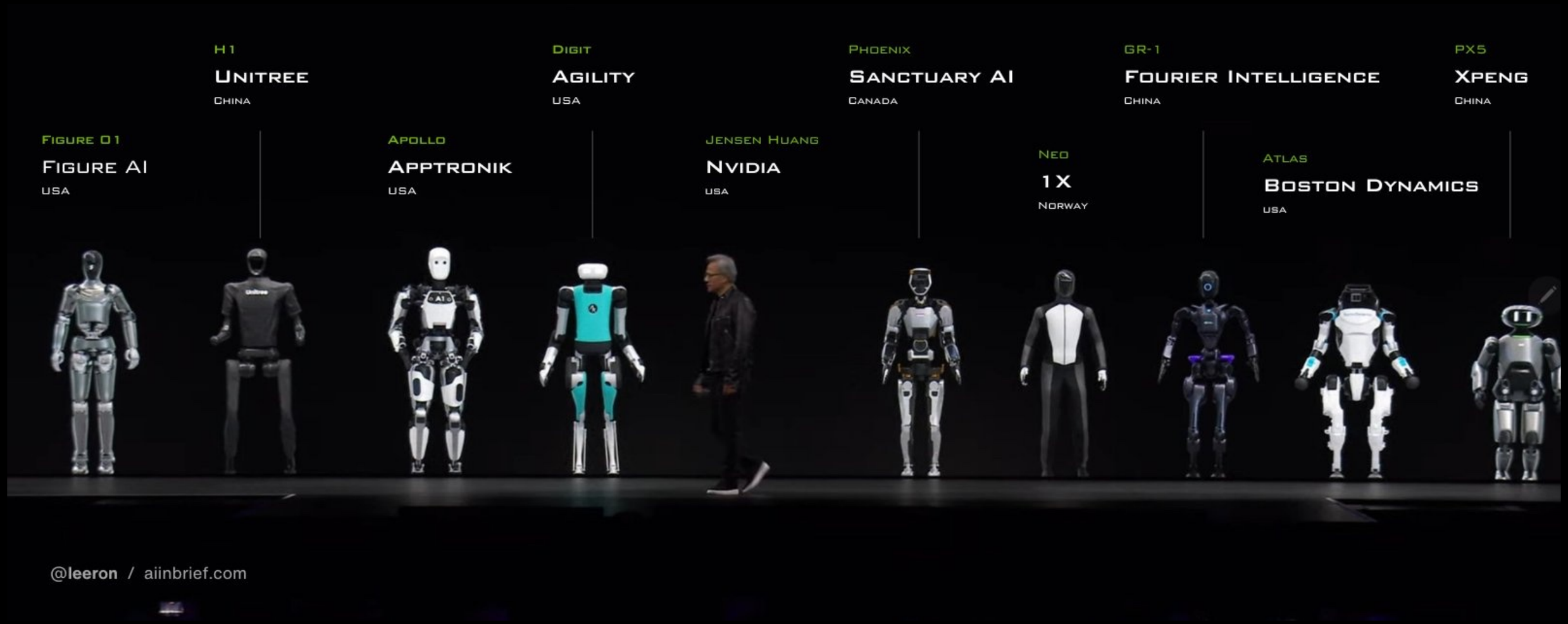


AGX

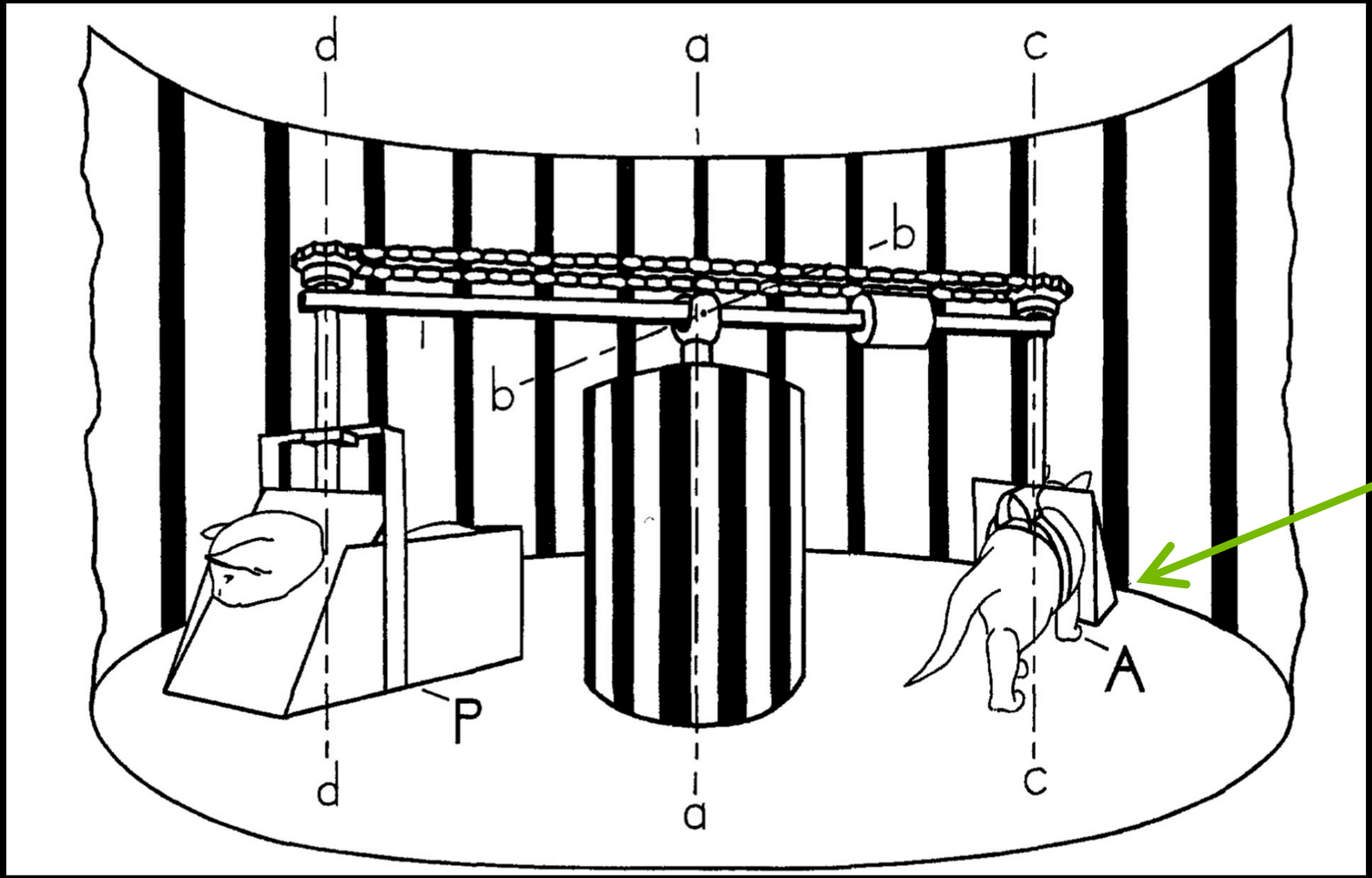

GR00T
Foundation Agent



Trillion-dollar Business



(Zero) Trillion-dollar Business



Embodied AGI

Held & Hein, 1963.

We are HIRING!!

GEAR Lab: **G**eneralist **E**mbodied **A**gent **R**esearch
https://research.nvidia.com/labs/gear

linxif@nvidia.com
dr.jimfan.ai@gmail.com

https://jimfan.me

NVIDIA