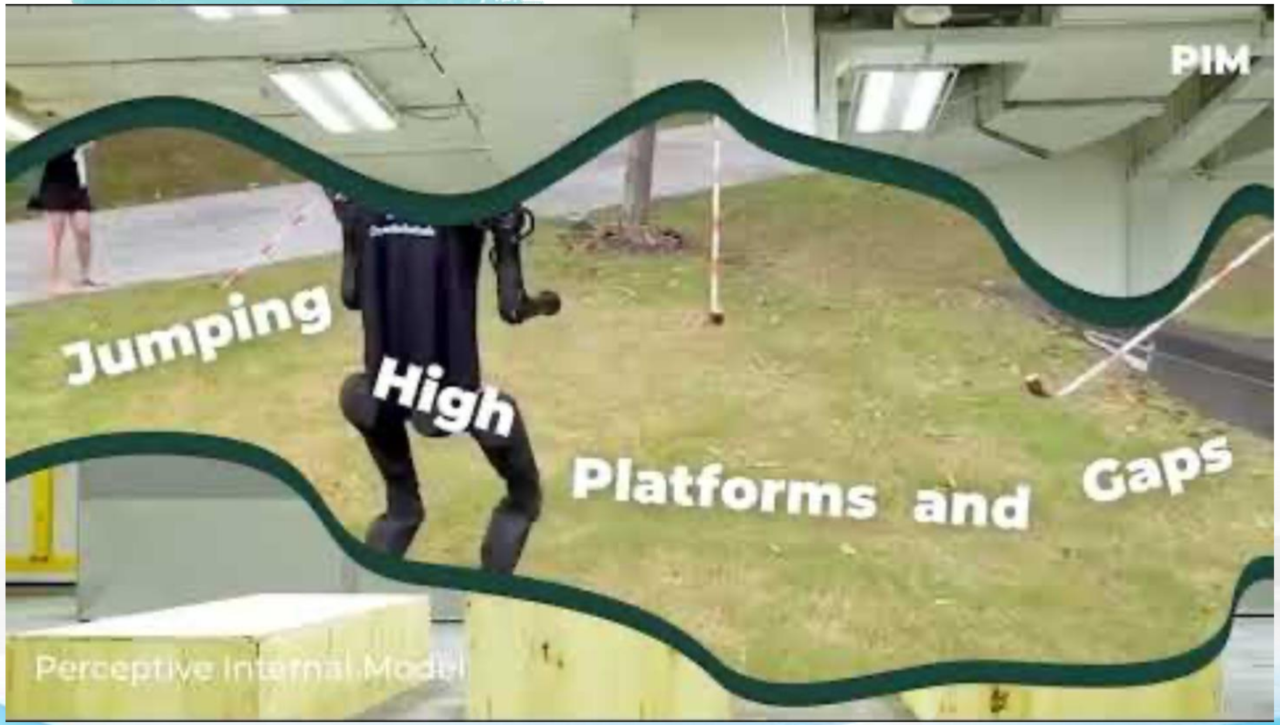
Learning Humanoid Locomotion with Perceptive Internal Model (PIM)

GROUP 3:
MENDOZA SANTIAGO
CHADLI NEIL
CHABOD JULES

Motivation: Quadrupeds vs Humanoids

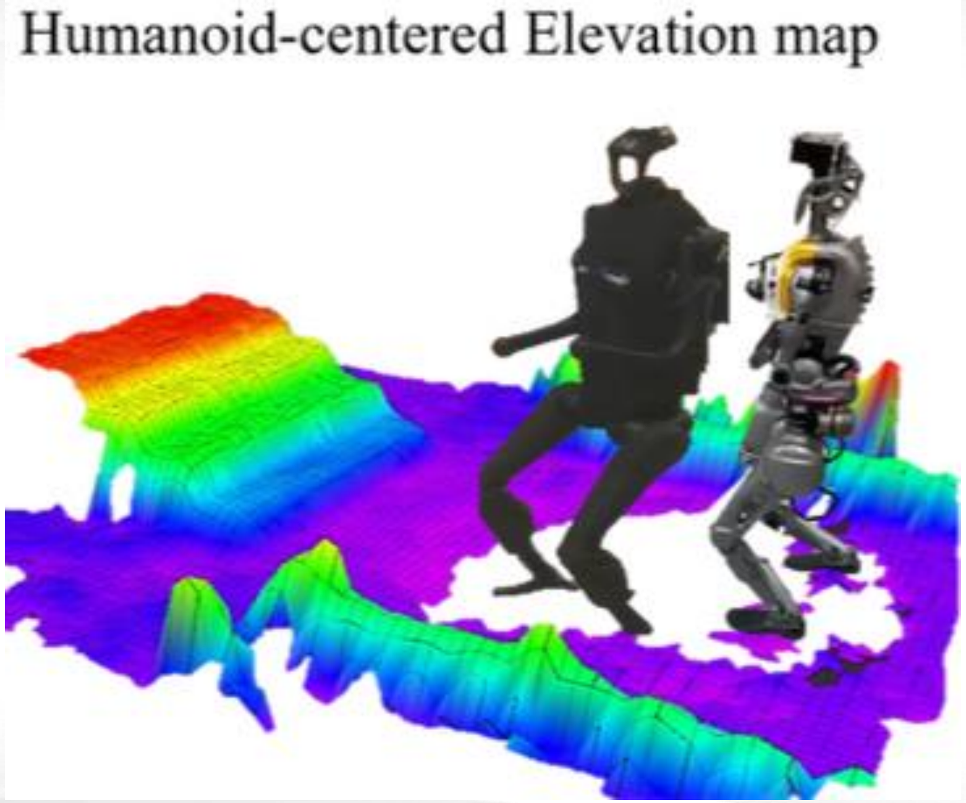


- Quadrupeds walk blind: low CoM, 4 contacts, stable morphology.
- Humanoids: tall, unstable, high DoF, narrow support polygon.
- Blind locomotion fails immediately on stairs, platforms, uneven terrain.

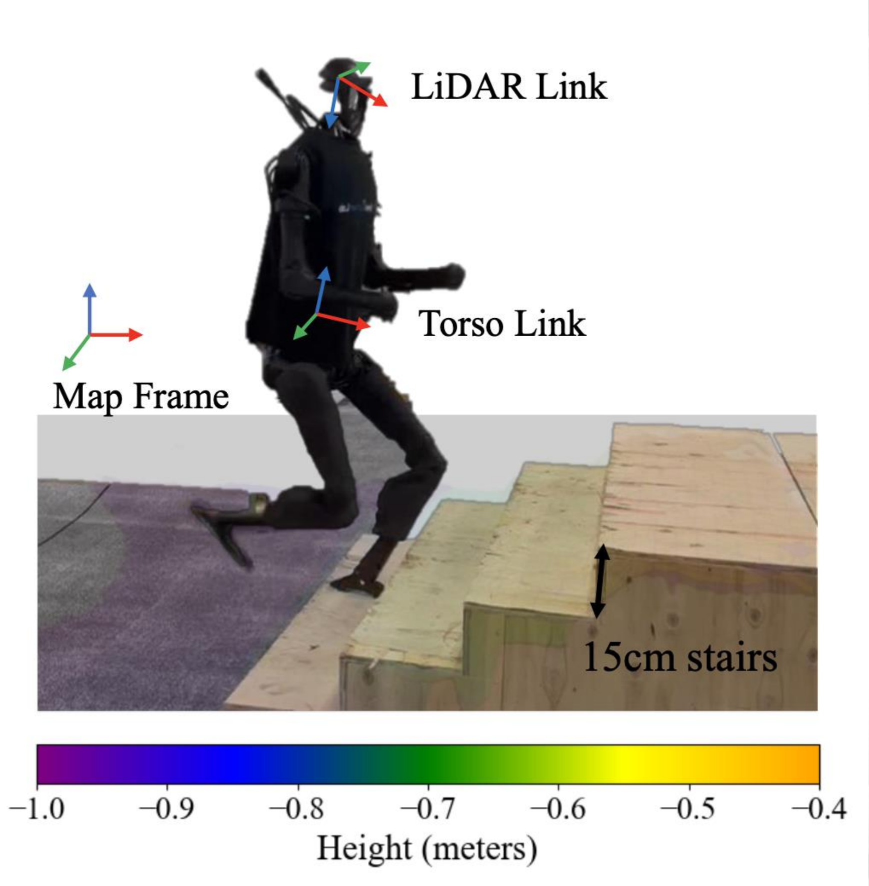
Why Not Just Add Vision?

Why Depth-Based Perception Fails on Humanoids:

- **Sensor noise & motion artifacts** (blur, shake, rolling distortion)
- **Large sim-to-real gap** (simulated depth ≠ real sensor data)
- **Includes non-terrain points** (walls, furniture, obstacles)
- **High computation & latency**, destabilizes control
- **Blind spots under the feet**



Current Problems

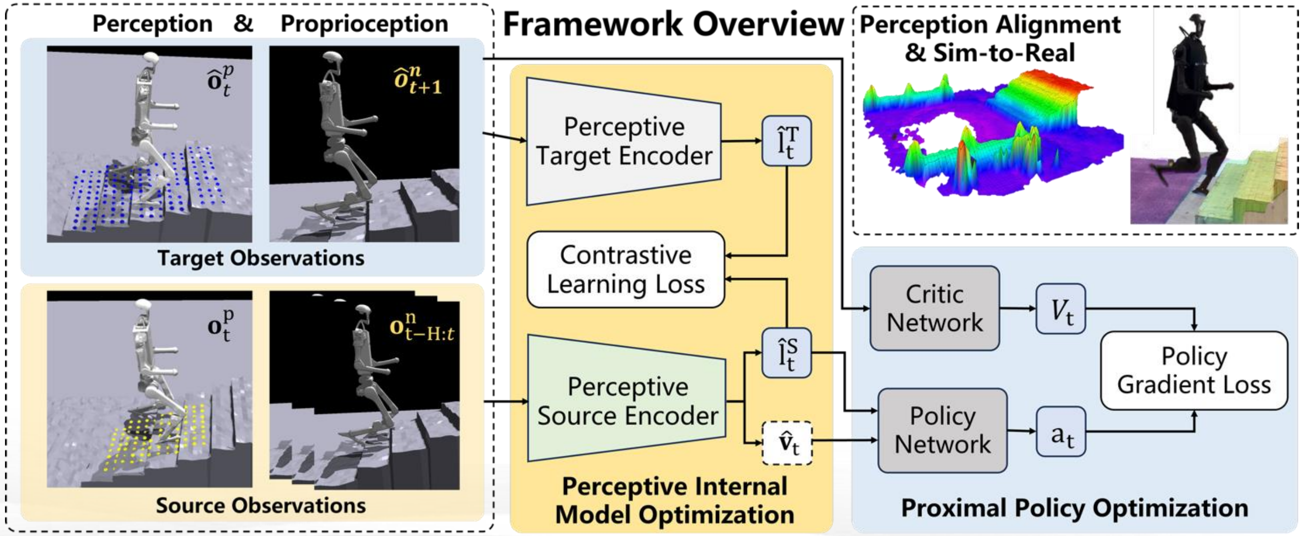


- Current methods use perception only for **foothold selection**.
- They do **not** predict how terrain affects **next body motion**.
- But stable humanoid walking requires **terrain-aware next-state prediction**.

Main Contribution

Why Depth-Based Perception Fails on Humanoids:

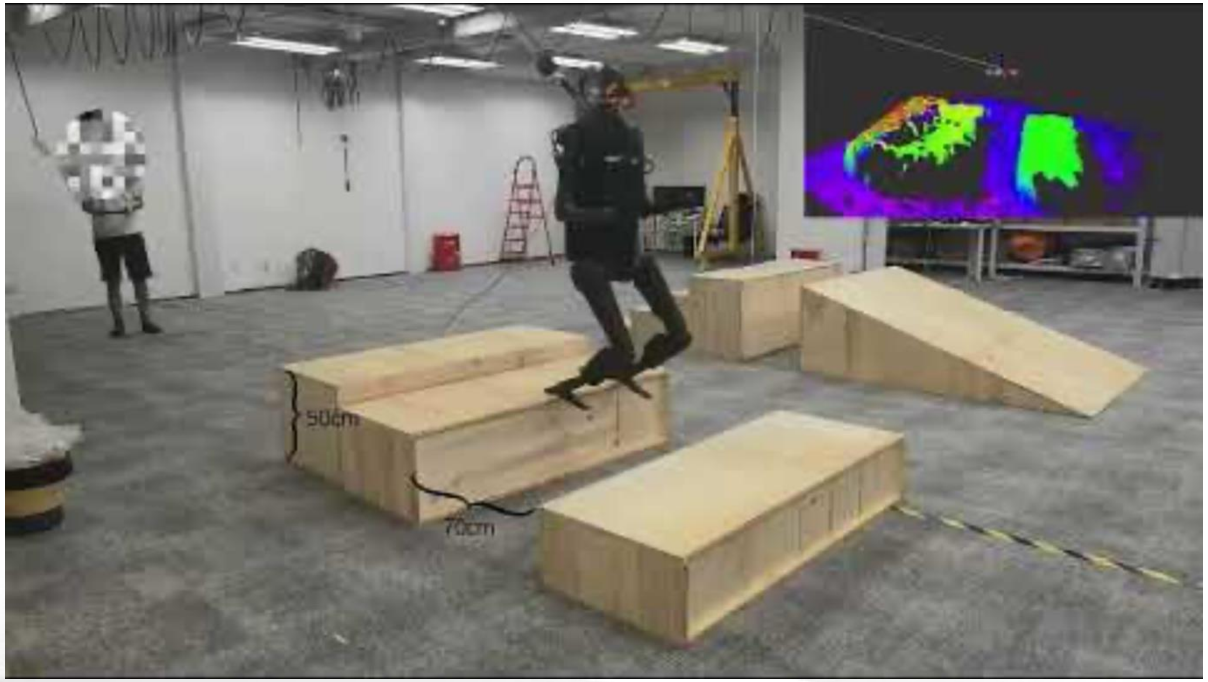
- Introduce **PIM**: Perception-aware Internal Model.
- Predicts the **next robot state** from proprioception + terrain geometry.
- Used inside a PPO locomotion framework.
- Handles **15 cm stairs**, **40 cm platforms**, gaps, uneven outdoor terrain.
- Extremely fast training: **~3 hours** on a single 4090.



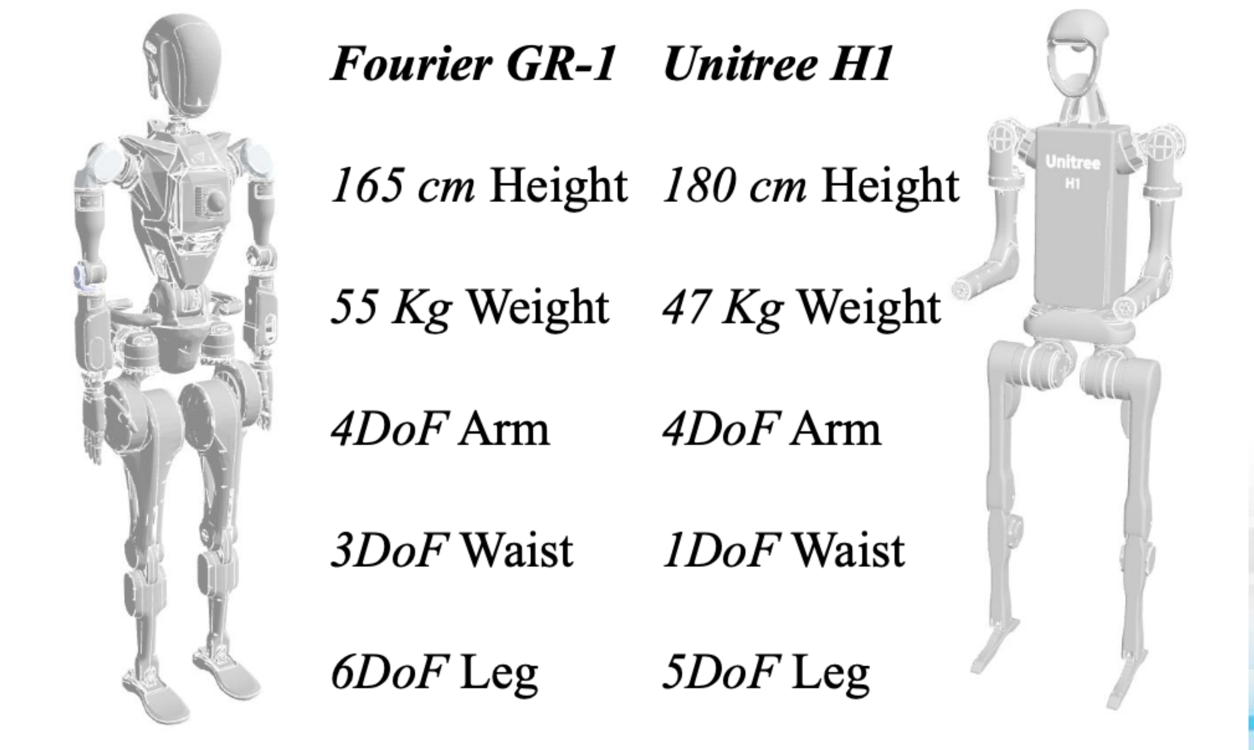
Main Contribution

Why Depth-Based Perception Fails on Humanoids:

- Introduce **PIM**: Perception-aware Internal Model.
- Predicts the **next robot state** from proprioception + terrain geometry.
- Used inside a PPO locomotion framework.
- Handles **15 cm stairs**, **40 cm platforms**, gaps, uneven outdoor terrain.
- Extremely fast training: **~3 hours** on a single 4090.



Technical Overview



Fourier GR-1	Unitree H1
165 cm Height	180 cm Height
55 Kg Weight	47 Kg Weight
4DoF Arm	4DoF Arm
3DoF Waist	1DoF Waist
6DoF Leg	5DoF Leg

- Control: **Torque control** → essential for balance & whole-body behavior.
- Proprioception: joint encoders + IMU.
- Perception: **Elevation maps** built from LiDAR FAST-LIO or RealSense.
- Elevation map downsampled to **96 height samples** → compact and stable.

Control Method

Perception and proprioception → PIM → Policy → Torques

Perception (terrain understanding):
RealSense / LiDAR builds an elevation map
=> extract 96 height samples around the robot
Proprioception:
IMU + joint encoders
A short history of states is stacked

Control Method

Perception and proprioception ⟶ PIM → Policy → Torques

Perception (terrain understanding):
RealSense / LiDAR builds an elevation map
=> extract 96 height samples around the robot
Proprioception:
IMU + joint encoders
A short history of states is stacked
Perceptive Internal Model (PIM):
Takes proprioception + height samples
Predicts:
- next-step linear velocity, and
- a latent representation of the next proprioceptive state

Gives the policy **foresight** about how terrain will affect the robot

Control Method

Perception and proprioception ⟶ PIM → **Policy** → **Torques**

Perception (terrain understanding):
RealSense / LiDAR builds an elevation map
=> extract **96 height samples** around the robot
Proprioception:
IMU + joint encoders
A short history of states is stacked
Perceptive Internal Model (PIM):
Takes proprioception + height samples
Predicts:

- **next-step linear velocity**, and
- a **latent representation of the next proprioceptive state**

Gives the policy **foresight** about how terrain will affect the robot

Locomotion Policy (neural network trained with PPO):
Input: current observations + PIM predictions
Output: **joint torque commands**
Runs at high frequency on the robot
Key idea:
The controller is no longer blind and acts based on *"what will happen next on this terrain."*

Training Framework: Alternating PIM and Policy Optimization

trained in simulation using two components that learn *together but not at the same time*:
PIM (Perceptive Internal Model)
→ learns to predict how the robot's body will evolve on the terrain
Locomotion Policy
→ learns how to act, using PIM's predictions as additional information

Training is alternating:

The PPO takes as input the robot's current proprioception, the terrain height samples, and the PIM's predicted future state;

the PIM is optimized separately using velocity regression and contrastive learning

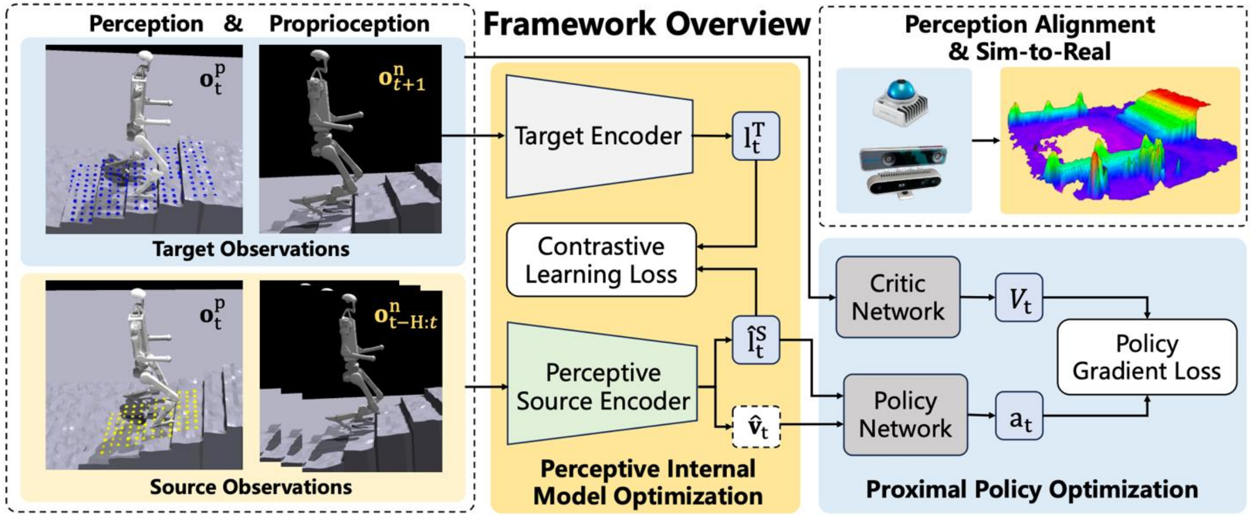


Fig. 2: Overview of our framework. Within PIM, we integrate perceptive information into the state predictor to achieve more comprehensive and accurate state prediction. A LiDAR-based elevation map serves as the perception model, enabling more precise perception alignment between simulation and real-world environments.

Training Tricks: Symmetry & Action-Space Curriculum

Symmetry Regularization (Mirroring)
To encourage natural, human-like walking, the policy is trained with left–right mirroring:
Observations and actions are duplicated in a mirrored version
The policy learns that swapping left/right should produce equivalent behavior
This prevents asymmetric gaits and improves stability on uneven terrain

Action-Space Curriculum

To stabilize learning:
The policy starts by controlling only **the legs**
Then the **waist** is enabled
Finally the **arms** are unlocked

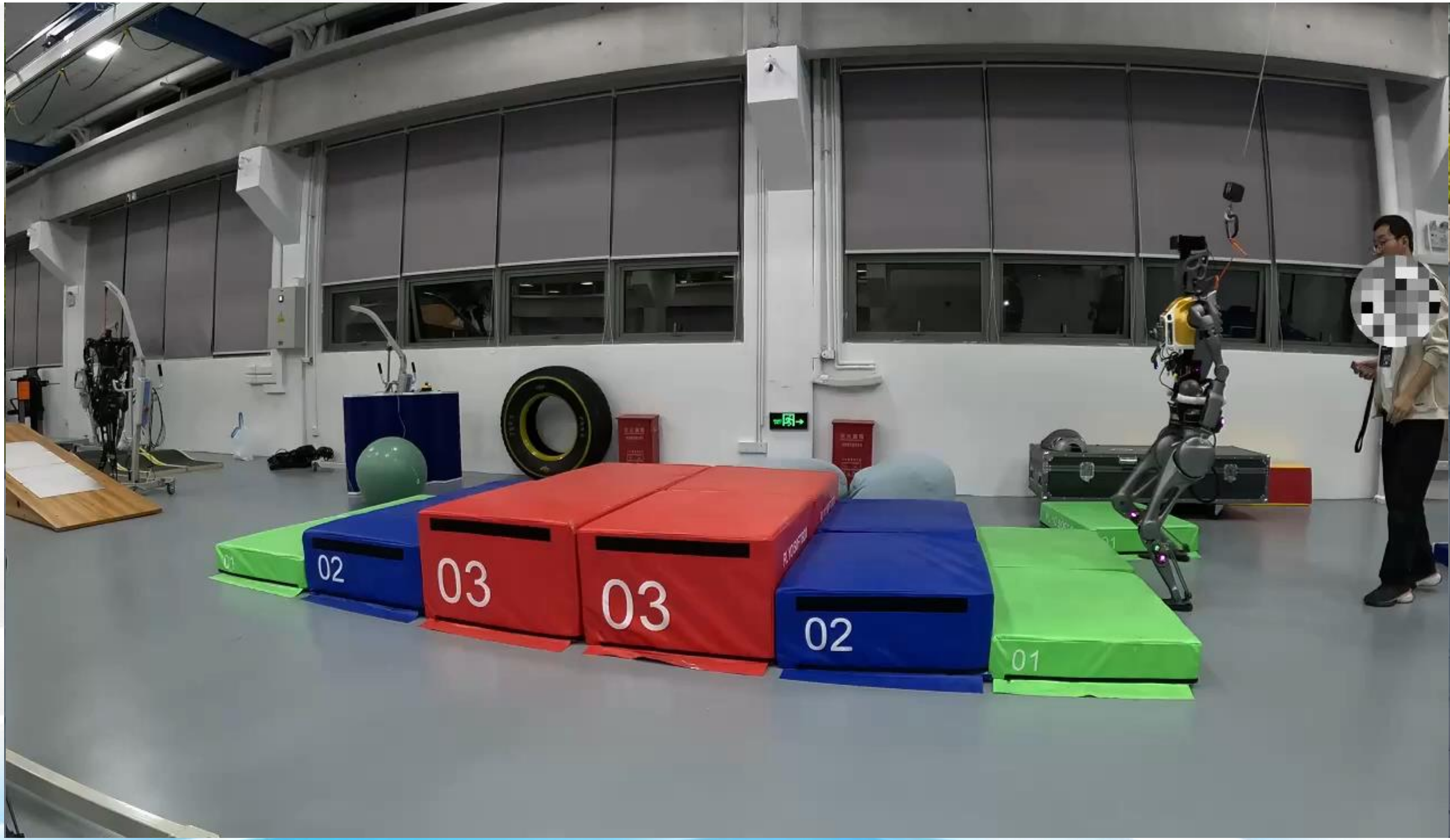
the robot masters balance first, then whole-body coordination.

Results: Unitree H1

- Climbs **15 cm** stairs (previous SOTA ~10 cm).
- Steps onto **40 cm** platforms and crosses gaps.
- Walks on uneven outdoor terrain.
- Emergent behaviors: arm swing, torso rotation.

Results: Fourier GR-1

- Same architecture works after retraining.
- Generalizes across morphology, mass distribution, feet design.
- Similar stair/platform performance.

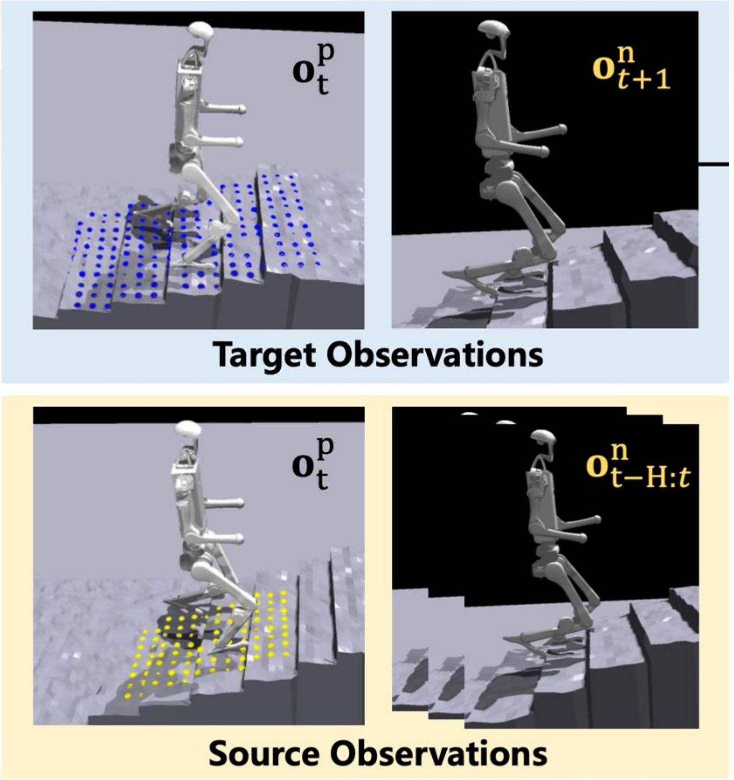


Training Efficiency

- Entire system trains in **≈ 3 hours** on a single RTX 4090.

- Elevation maps avoid depth rendering → training is **fast & stable**.

- Single-stage pipeline (no multi-phase sim-to-real).



Citations

- Already **39 citations** (less than one year after release).

- Cited in *Learning Humanoid Standing-up Control across Diverse Postures* (Hyung et al.)

Limitations

Perception & Mapping Constraints

- Odometry accuracy
- Elevation map is 2.5D
- Sensitive to sparse or noisy LiDAR
- Assumes rigid, static terrain.

Challenging Real-World Terrains

- Soft or deformable ground
- Moving or dynamic terrain
- Slippery surfaces
- Perception dropout

questions