GR00T N1 (2025)

NVIDIA GR00T Team

November 12, 2026

Georgia Tech

Outline

1 **The problem** — most manipulation data has no robot action labels — 3

2 **Building blocks** — flow matching · cross-embodiment · learning from unlabelled video · synthetic data — 4–7

3 **GR00T N1** — dual-system architecture, Eagle-2 backbone, per-robot adapters — 8–10

4 **Does it transfer?** — gains by pre-training exposure; is the shared space aligned? — 11–12

5 **Data recipe** — 8,376 h pyramid, 97% synthetic for the humanoid; what sim buys — 13–14

6 **Results under scrutiny** — zero-shot, sim, real GR-1, pseudo-actions; do the gains hold up? — 15–19

7 **Lessons & what came next** — GR00T N2 world-action model; what should the action space be? — 20–22

8 **Xiaomi-Robotics-1** — then: what persisted, limitations, discussion — 23–50

Georgia Tech

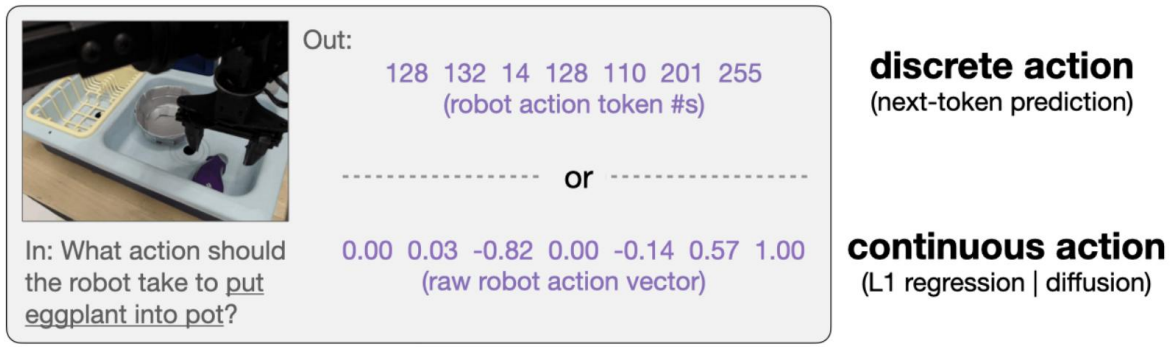
Problem Statement

Most data that shows manipulation has **no robot action labels**: human video, generated video, other robots.

Can a humanoid policy be pre-trained on it anyway, by inferring **pseudo-actions** and scaling **synthetic data**, and still adapt from **a few demonstrations**?

- *π0 pre-trained on ~10K h of action-labelled teleop from 7 robot configurations. GR00T N1 has 88 h of real humanoid teleop.*

Preliminary: precise continuous actions?



OpenVLA-OFT, Kim et al., 2025

Autoregressive action tokenization can become inefficient for high-dimensional, temporally coordinated robot control.



BET



Diffusion Policy

Diffusion Policy, Chi et al., 2023

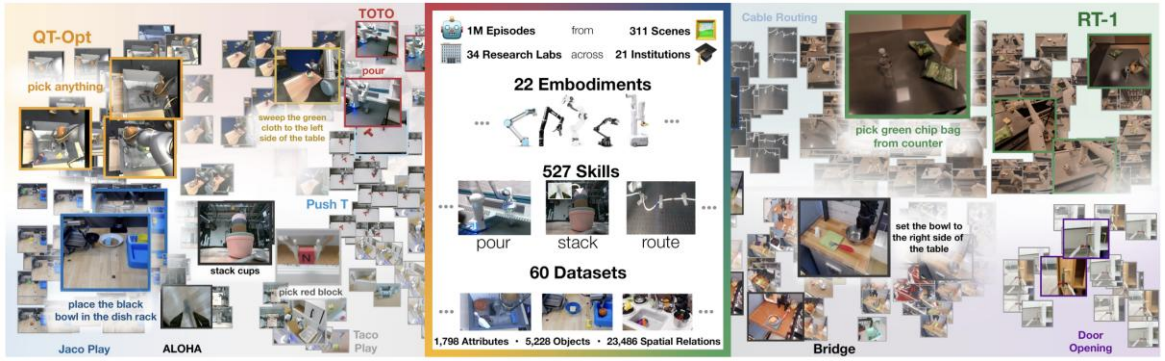
Diffusion and flow-matching policies jointly generate continuous action trajectories, capturing multimodal behaviors and temporal dependencies.

Georgia Tech

Preliminary: train across different robot bodies?

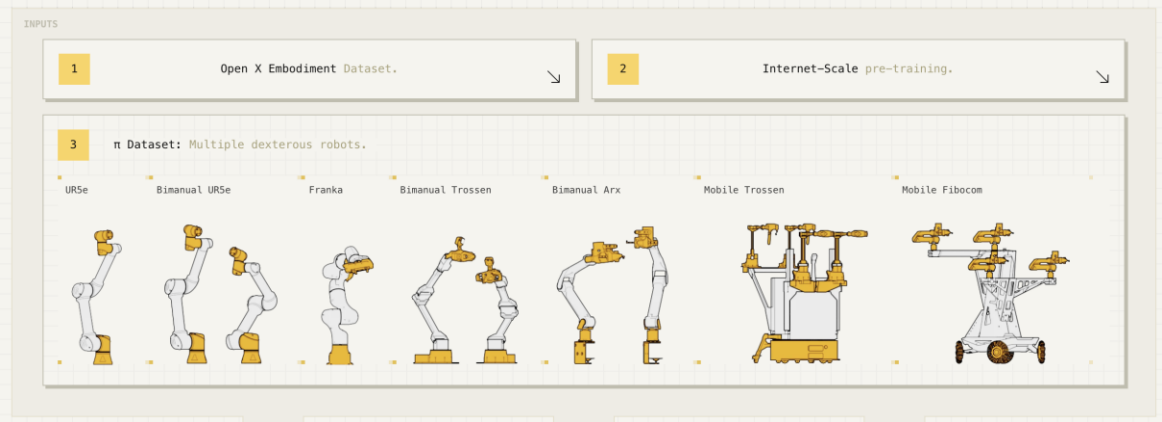
Single embodiment datasets can't provide large enough training scales.

Different embodiments have different action spaces.



Open X-Embodiment, O'Neill et al., 2024

Unlike π0, which zero-pads every robot to one max action dim, GR00T gives each embodiment its own MLP encoder/decoder into a shared token space.



π0, Physical Intelligence Team, 2024

Georgia Tech

Contribution: learn from videos that don't have action labels?

A · Latent actions (LAPA): learn an action code from video alone





Step 1 · train

Encoder → z_t continuous → **Codebook** snap to nearest code → **Decoder** also sees x_t → 

Trained to rebuild x_{t+h} from x_t plus one code, so the code has to capture the motion.

Step 2 · label

Encoder → z_t → action label of a separate "LAPA" embodiment
works on any video: human, robot, generated

B · Inverse dynamics (IDM): predict real actions

x_t, x_{t+h} → **IDM** trained on real robot demos → $a_t \dots a_{t+h}$ robot's own space → label for generated video
needs labelled robot data first

Who gets which label

Human video
latent (A)

Neural video
latent (A) + IDM (B)

Real robot
true actions + latent (A)

Either way, the policy trains on pseudo-actions with the same flow-matching loss.

Trade-off: latent labels fit any video but live in a new space; IDM labels are real actions, but only for robots that already have data.

Georgia Tech

LAPA, Ye et al., 2024 · GR00T N1, NVIDIA 2025 (Sec. 2.2, App. F)

Contribution: scale robot data beyond real demonstrations?

Physics sim · DexMimicGen

few dozen human demos → split into object-centric subtasks → re-target to new object poses → replay; keep only successes

780K trajectories in 11 h

≈ 6,500 h of demos · 540K used in pre-training (54 receptacle pairs)

Neural video · WAN2.1 I2V-14B

LoRA on 88 h of teleop → LLM writes new prompts → generate 81-frame clips → LLM judge filters

827 h · ~300K clips (~10× real)

≈ 105K L40 GPU-hours · labelled with latent + IDM actions



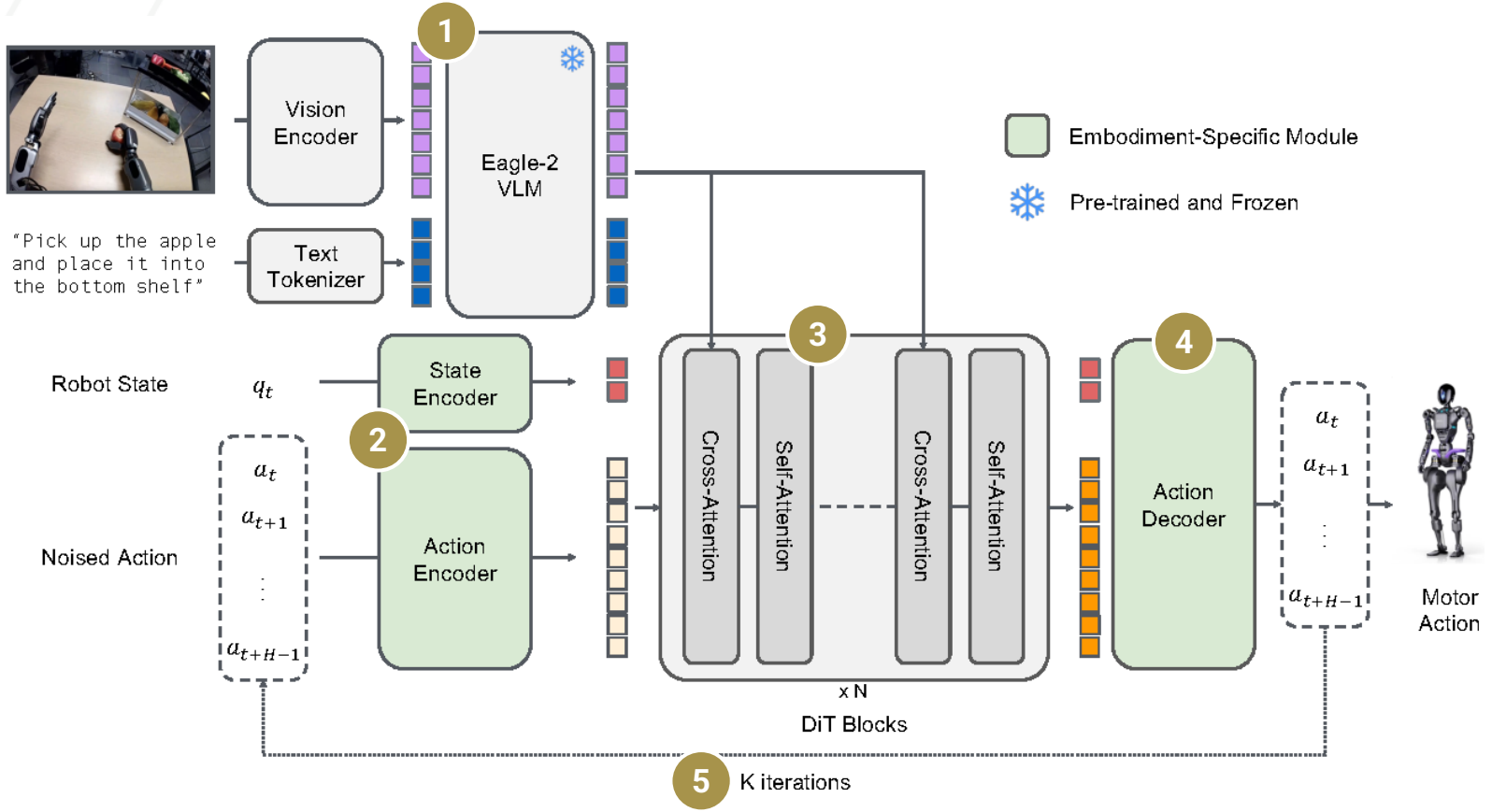
Generated, not recorded: "use the right hand to pick up cucumber to basket" (Fig. 5; red box = the real first frame)

Synthetic data turns 88 h of humanoid teleop into 2,658 h, 31% of all pre-training hours.

Georgia Tech

GR00T N1, NVIDIA 2025 (Sec. 2.2, 3.2, App. F)

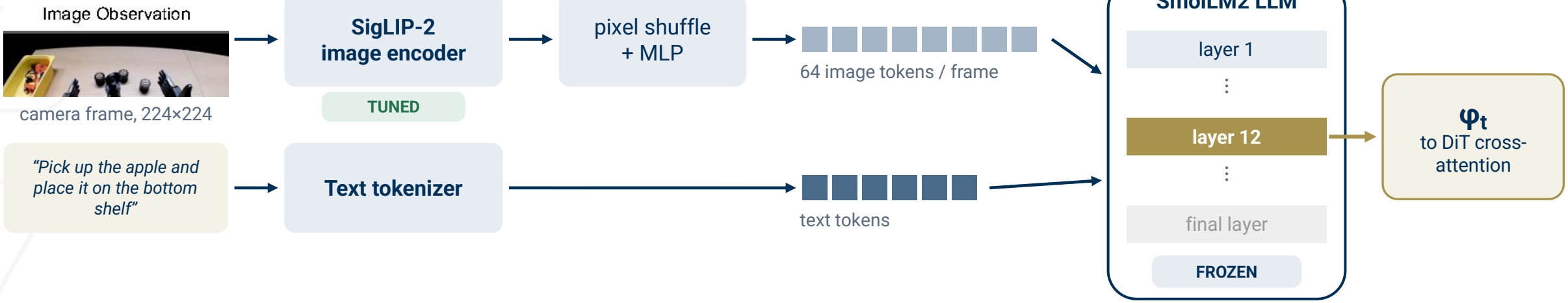
GR00T N1: Dual-System Architecture



1. **System 2 · Eagle-2, ~10 Hz**
   layer-12 tokens; LLM frozen, vision encoder tuned
2. **Per-embodiment encoders**
   state q_t and noised actions → width d
3. **System 1 · DiT × N, 120 Hz**
   self-attn on state + action tokens; cross-attn to VL tokens
4. **Per-embodiment decoder**
   velocity for H = 16 future actions
5. **K = 4 Euler steps**
   63.9 ms per 16-action chunk on one L40

Train: $A\tau = \tau A + (1 - \tau)\varepsilon$, minimise $\|V\theta(\varphi, A\tau, q) - (\varepsilon - A)\|^2$, $\tau \sim$ Beta (as in π0)

System 2: The Eagle-2 VLM Backbone



1.34B of 2.2B params | ~10 Hz on one L40 | Layer 12 > final layer: faster + better

How Eagle 2 is trained before GR00T sees it

Stage 1 · align
connector only · 1.2M caption samples

Stage 1.5 · pre-train
full model · 21.6M samples from 180+ sources

Stage 2 · instruction-tune
full model · 4.6M curated samples

In GR00T
LLM frozen · vision encoder tuned on robot data

GR00T uses a small SmolLM2 + SigLIP-2 variant trained with this general recipe; its exact data mix is not published.

A frozen LLM keeps web knowledge but caps grounding: N1 followed instructions in 46.6% of real GR-1 trials; N1.5's grounding-tuned Eagle 2.5 reached 93.3%.

Georgia Tech

GR00T N1, NVIDIA 2025 · Eagle 2, Li et al., 2025 (Sec. 2, Table 4) · GR00T N1.5 blog, NVIDIA 2025

Cross-Embodiment: One Shared Trunk, Per-Robot Adapters

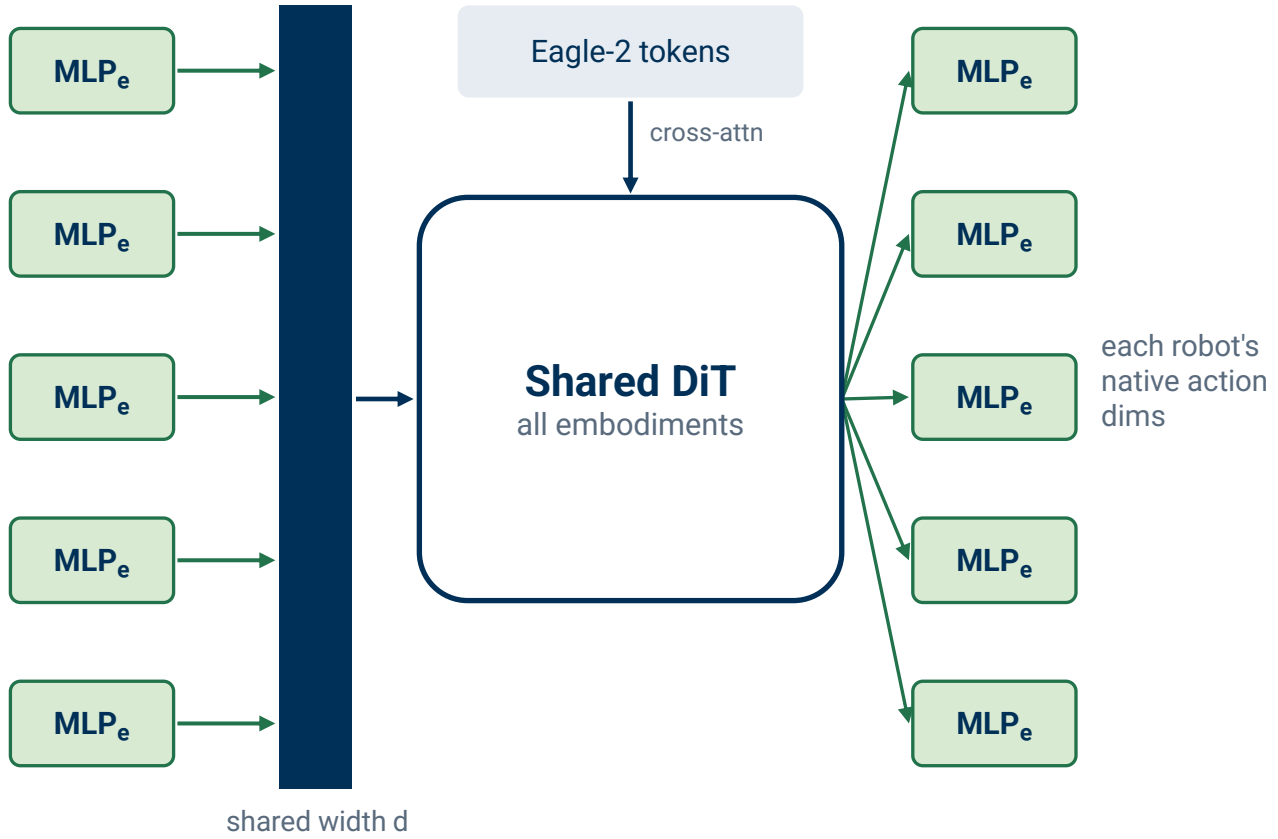
GR-1 humanoid
arm + hand joints, waist, neck

Franka (RoboCasa, DROID)
EEF pose + gripper

Bimanual Panda
2 × EEF pose + grippers / hands

Human video → "LAPA"
VQ-VAE latent action

Neural video → "IDM"
inverse-dynamics pseudo-actions



vs. π0

π0: pads every robot's state and action vectors with zeros up to 18 numbers, the size of its largest robot, then uses one shared projection

GR00T: one MLP per embodiment (like Octo), no padding

A new robot starts with untrained MLPs

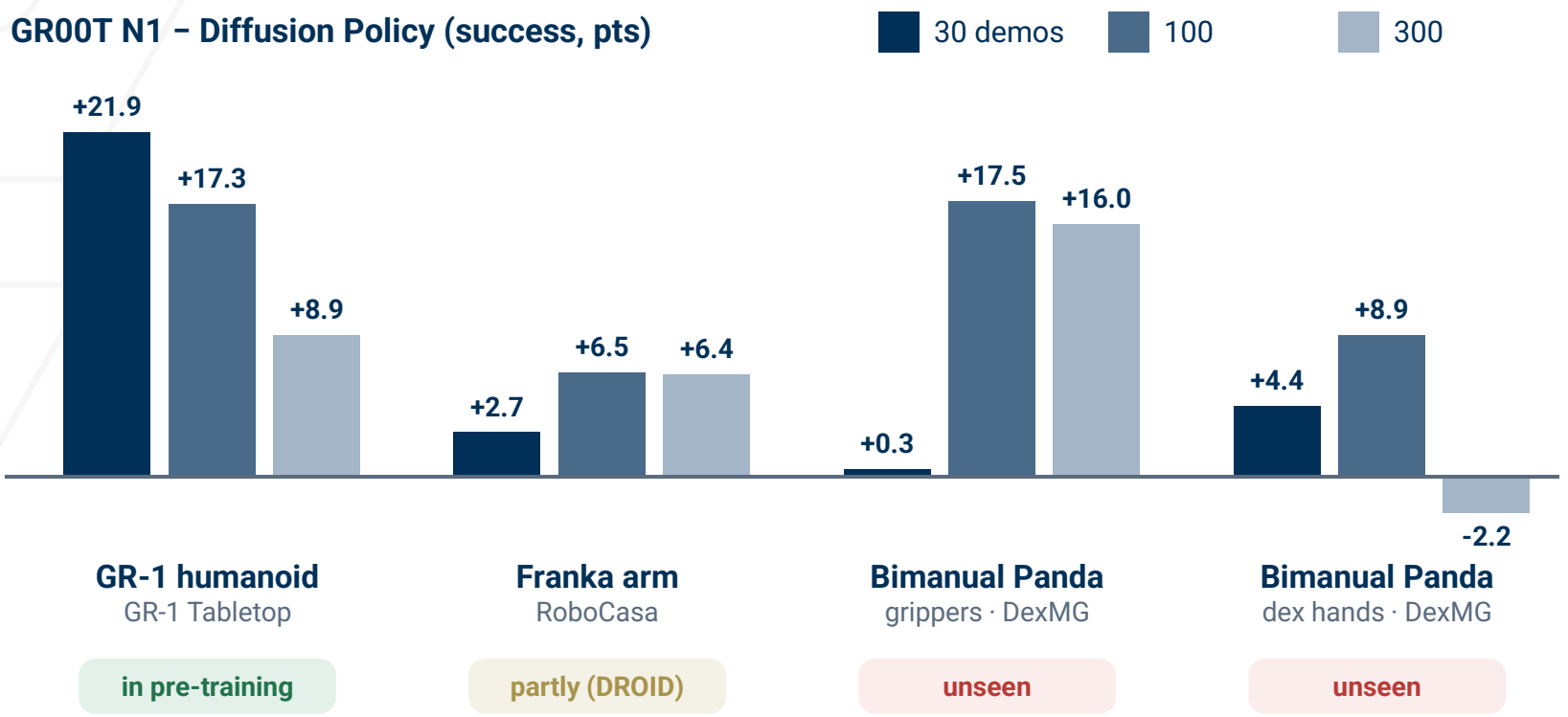
Pre-processing:

- rot. state → 6D
- rot. action → axis-angle
- min-max normalise
- order: rot → pos → grip, L → R

Georgia Tech

GR00T N1, NVIDIA 2025 (Sec. 2.1, App. E.2) · π0, Black et al., 2024 · Octo, 2024

Cross-Embodiment: Gains by Pre-training Exposure



Seen body: gains even at 30 demos
GR-1: +21.9 pts with 30 demos per task

Unseen bodies: fresh MLPs need data
Panda grippers: +0.3 at 30 → +17.5 at 100

Shared trunk still transfers
unseen bodies gain +9 to +18 pts at 100 demos

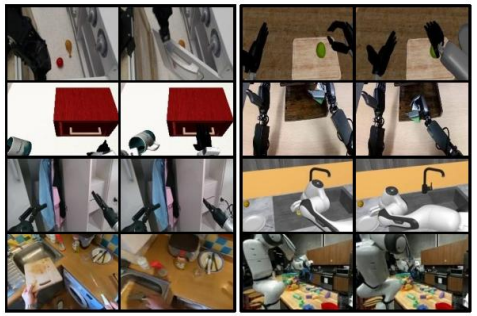
Shown: one fixed-size token space shared by every body, reusable by new bodies once their adapters train.

Georgia Tech

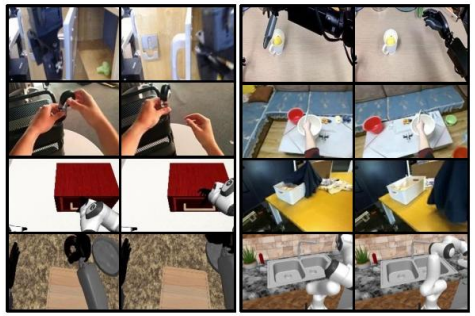
GR00T N1, NVIDIA 2025 (Table 4 per-task results, Table 7 pre-training mix)

Does Cross-Embodiment Alignment Actually Happen?

Claim (Sec. 2.2)
"…share the same learned latent action space, potentially improving cross-embodiment generalization."



Latent action 1: Move the right arm to the left



Latent action 2: Move the right arm to the right

The only evidence: latent-action retrievals that look alike across bodies (Fig. 4). No metric.

If bodies were aligned…	What the paper shows
a new body works zero-shot →	no zero-shot cross-embodiment test at all
a few demos suffice on a new body →	unseen Panda bodies: +0.3 / +4.4 pts at 30 demos
transfer comes from the shared space →	gains need ~100 demos to train fresh adapters (+17.5 / +8.9)
alignment is measurable →	no probe, retrieval accuracy, or ablation

Indicative: without zero-shot transfer, the shared space acts as shared trunk capacity, not an aligned action space. N1.7 later replaced it with one relative-EEF space.

Georgia Tech

Training Recipe & Pre-training Mix

0 VLM pre-training Eagle-2 on web image-text

1 Pre-training · all embodiments

200K	16,384	≤1,024	~50K
steps	batch	H100s	GPU-hours

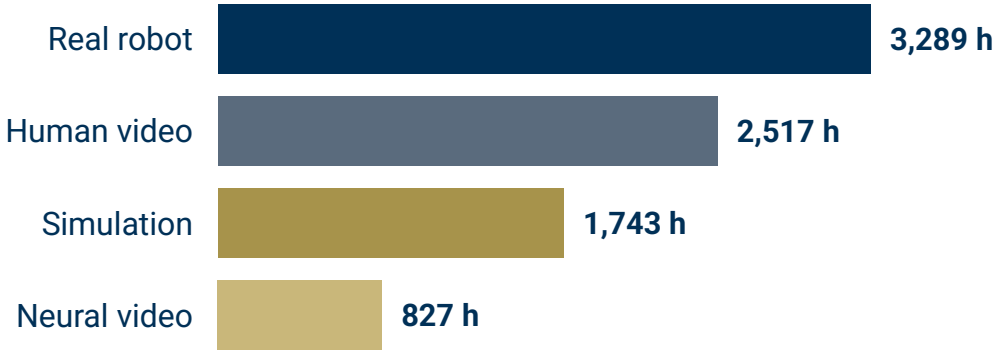
Targets: robot → true + latent actions · neural video → latent + IDM · human video → latent

2 Post-training · one embodiment at a time

20–60K	128–1,024	1× A6000	*optional: + neural trajectories*
steps	batch	is enough	

Vision enc: tuned | LLM: frozen | Adapters + DiT: tuned

Pre-training data: 8,376 hours



Target humanoid (GR-1): 2,658 h



↑ 88 h real

97% of the humanoid's data is synthetic.

Georgia Tech

GR00T N1, NVIDIA 2025 (Sec. 2.3, Tables 6–7)

What Does Sim Data Actually Buy?

	Physics sim DexMimicGen, 1,743 h	Neural video WAN2.1 I2V, 827 h	Real teleop GR-1, 88 h
Layouts, poses, distractors	yes	yes	costly
Rigid pick-and-place	yes	labels inferred	yes
Articulated: doors, drawers	benchmarks only	looks right	yes
Liquids, granular, deformables	absent	physics unchecked	yes
Force / torque signals	not used	none	not logged

N1 §4.6, in its own words
synthetic data struggles to stay diverse “while adhering to the laws of physics”

GR00T 2 (2026)
adding sim lifts G1 out-of-distribution success 6.7% → 63.3% (GR00T 1.7: 40%)

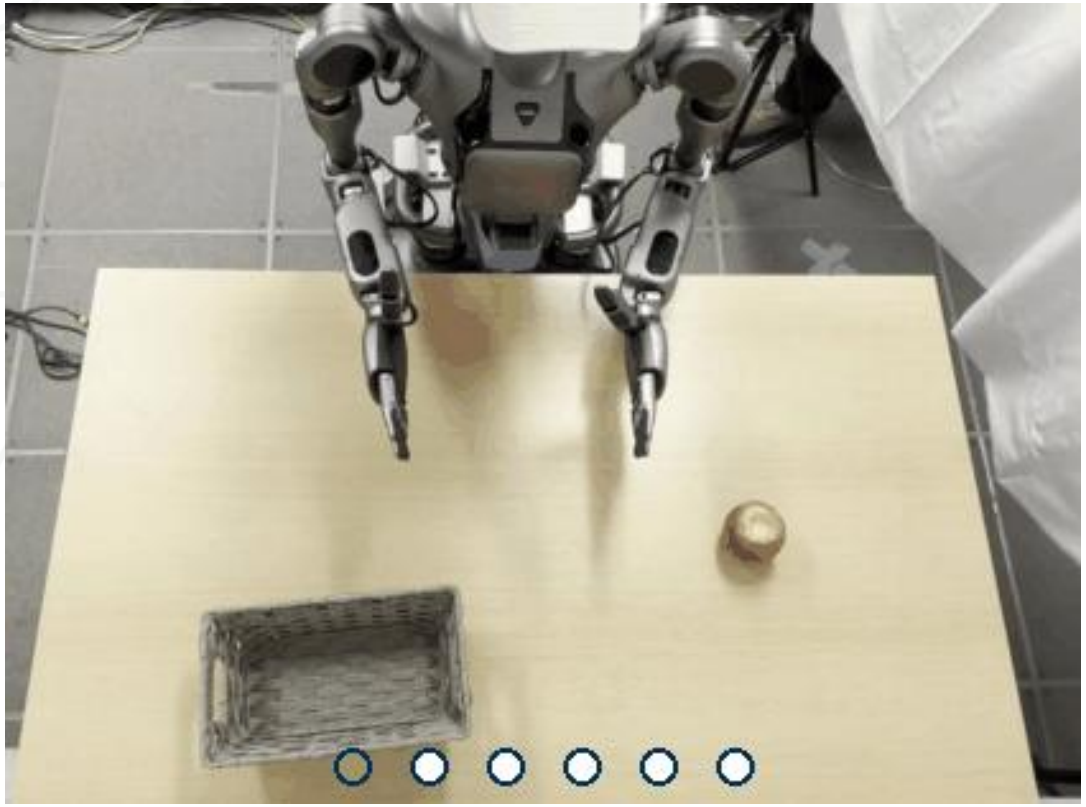
Real-to-sim twins (2026)
agent-built twins add articulated, deformable, particle tasks, used mainly to evaluate

Sim buys coverage of where things are; contact physics still has to come from real data.

Georgia Tech

GR00T N1, NVIDIA 2025 (Sec. 3.2, 4.6, Fig. 13) · GR00T 2, NVIDIA GEAR 2026

Pre-trained, Zero-Shot, and What Post-Training Erases



Pre-trained GR00T N1, zero-shot: "pick up the red apple and place it into the basket", apple placed out of the right hand's reach

76.6%
bimanual hand-over, then place on shelf (11.5 / 15)

73.3%
novel object → unseen container (11 / 15)

Pre-trained: hand-over ✓ → *fine-tune on right-hand-only demos* → **Post-trained: fails ✗**

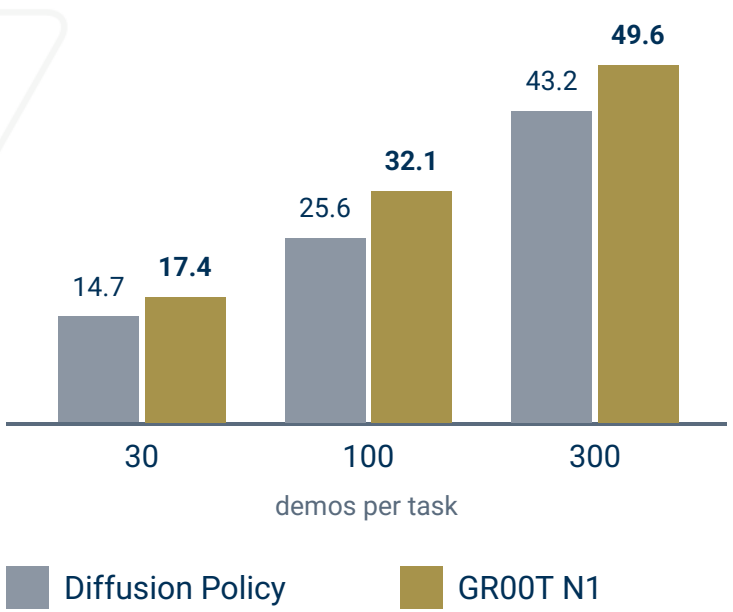
Post-training narrows behaviour to the fine-tuning data; generality is not kept by default.

Georgia Tech

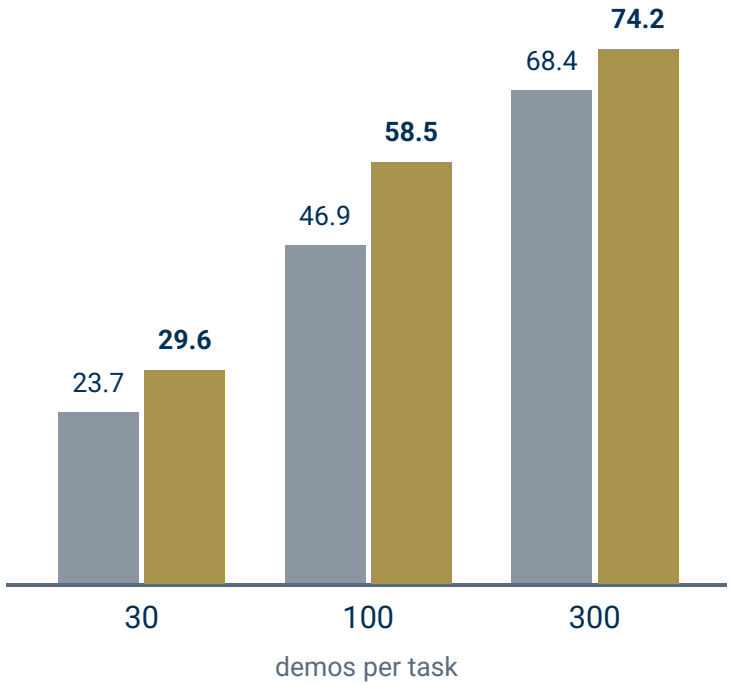
GR00T N1, NVIDIA 2025 (Sec. 4.4–4.5, Figs. 8, 11)

Simulation Results: Biggest Gains with Few Demos

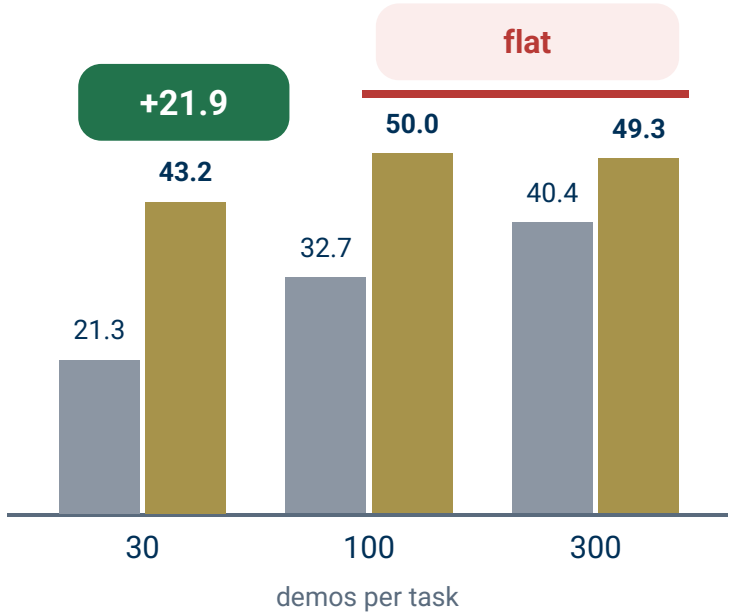
RoboCasa · 24 tasks · Franka



DexMG · 9 tasks · 3 bimanual bodies



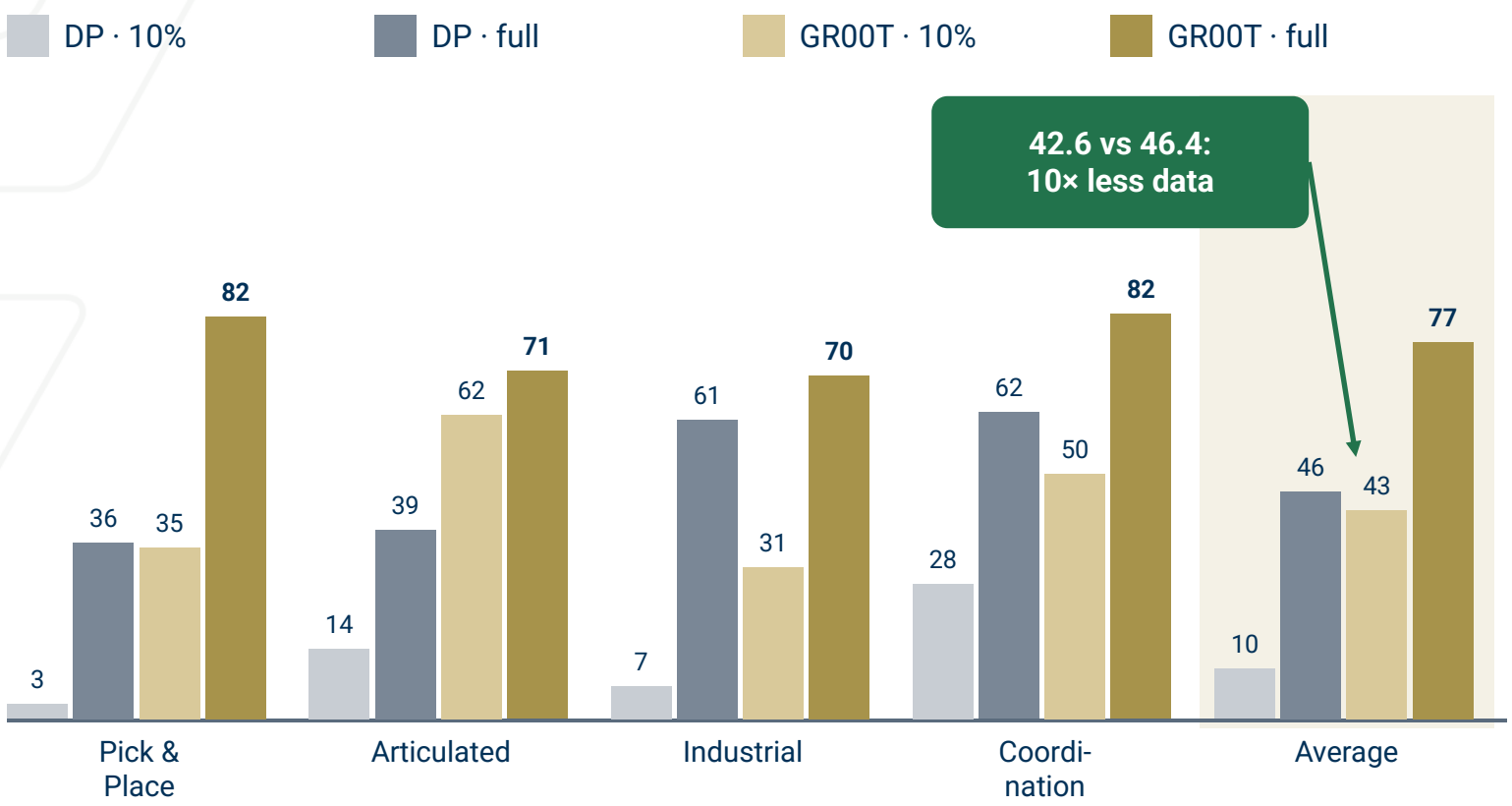
GR-1 · 24 tasks · humanoid

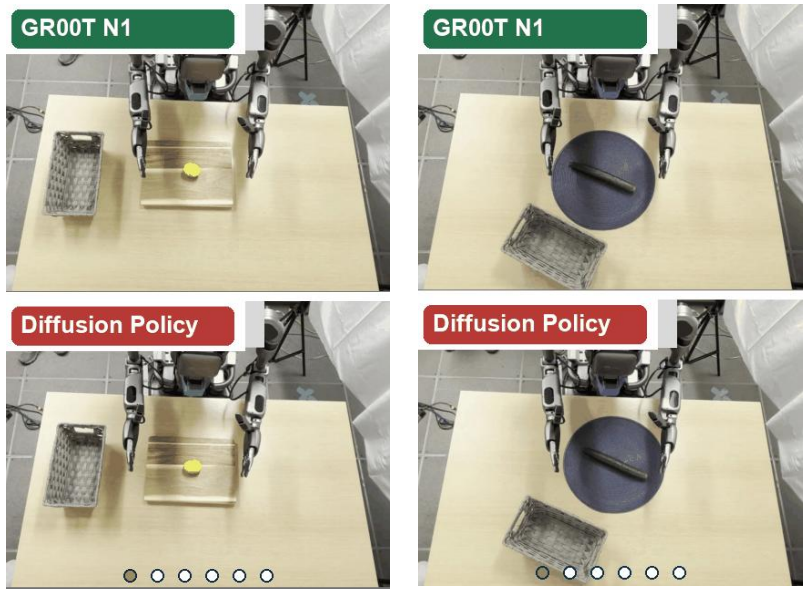


Diffusion Policy GR00T N1

GR00T N1, NVIDIA 2025 (Table 2, Table 4, Fig. 10)

Real GR-1: Pre-training Buys ~10× Data Efficiency





DP stalls at the start

DP mis-grasps

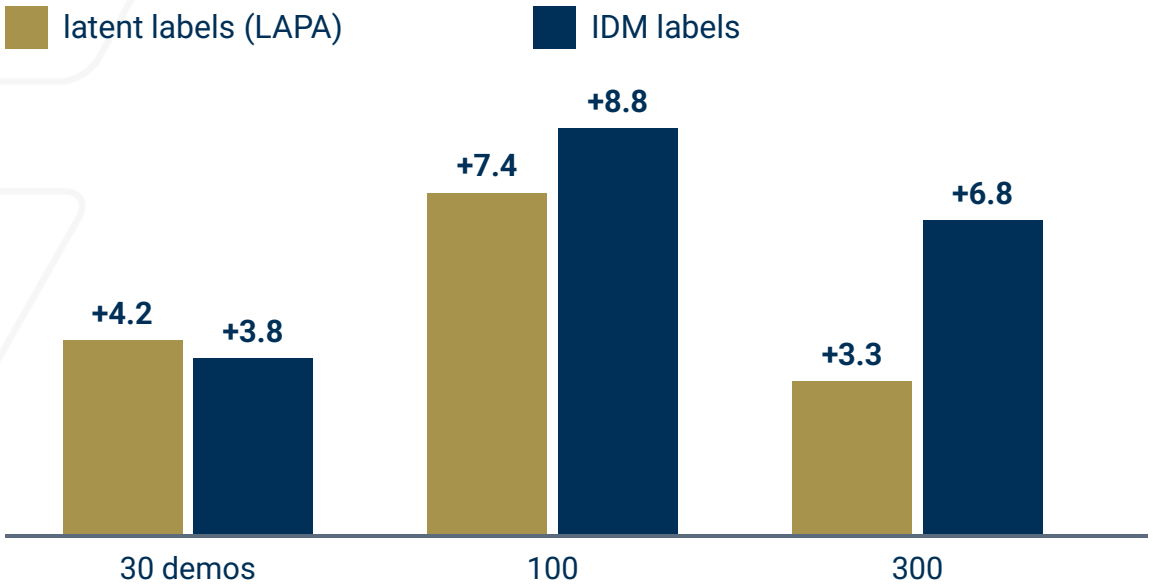
Pre-training pays off most where data is scarcest; with full data GR00T still leads by 30 points.

GR00T N1, NVIDIA 2025 (Table 3, Table 5, Figs. 9, 12)

Georgia Tech

Do Pseudo-Actions Work?

RoboCasa: gain from co-training with neural trajectories (pts over GR00T N1)



Latent labels win with 30 demos; IDM wins once it has real data to learn from.

Post-training: yes, modestly
+4–9 pts on RoboCasa; real GR-1 33.8% → 39.6% with 10% of the data

Pre-training: never tested
pseudo-actions on 2,517 h of human video are not ablated; N1.5 later added FLARE to make human video useful

Not free
827 h of generated video ≈ 105K L40 GPU-hours

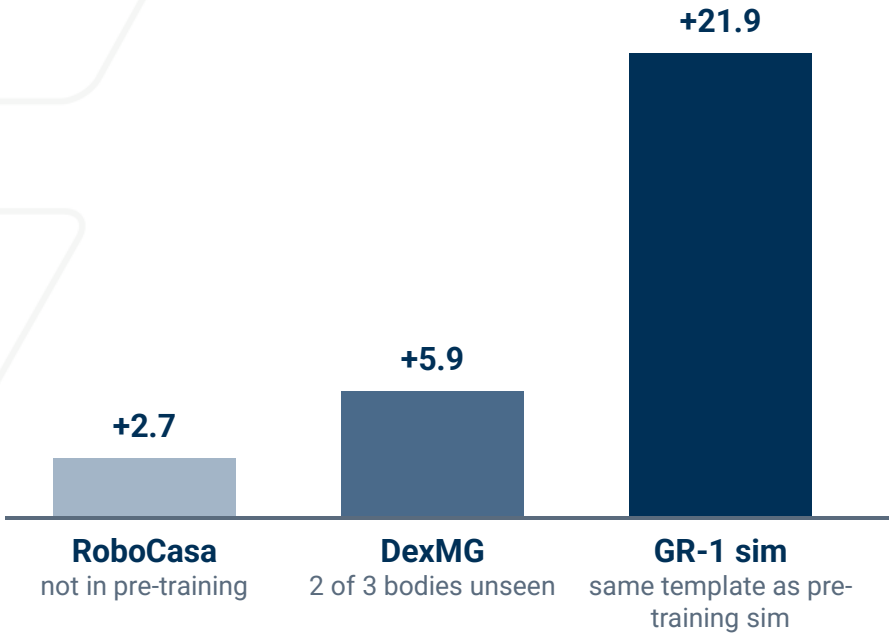
Indicative: pseudo-actions help as post-training augmentation; the pre-training bet the thesis rests on is never measured.

Georgia Tech

GR00T N1, NVIDIA 2025 (Sec. 4.4, Fig. 9) · GR00T N1.5 blog, NVIDIA 2025

Do the Benchmark Gains Hold Up?

Gain over Diffusion Policy at 30 demos vs. overlap with pre-training



Baselines
from-scratch BC-Transformer / Diffusion Policy only; no π0, Octo, OpenVLA, or GR00T without pre-training

Benchmark overlap
GR-1 sim tasks reuse the pre-training sim template (“rearrange A from B to C”)

Reporting
sim: best of the last 5 checkpoints · real: 10 trials per task, partial credit · zero-shot: 15 trials

Noise
Dark Cabinet 86.6% with 10% of the data vs 69.7% with all of it; Box Cleanup 29.4 vs DP 80.8 at 100 demos

Indicative: the gains are real, but they shrink as the test data moves away from the pre-training distribution.

Georgia Tech

GR00T N1, NVIDIA 2025 (Sec. 3.2, 4.3, Tables 2–5)

Lessons from GR00T N1

Strengths

Data pyramid
a concrete recipe for robots with almost no real data

Cheap synthetic multiplier
780K sim trajectories in 11 h; ~10× real data via video generation

Swappable System 2
the cross-attention seam let NVIDIA swap Eagle-2 → 2.5 → Cosmos-Reason2

Open
weights, pre-training data, and sim benchmarks are public

Open questions

How much is pre-training?
every comparison is against from-scratch policies

Which pyramid layer matters?
no layer is ablated; N1.5 later added FLARE to use human video

Do embodiments really align?
no zero-shot transfer to a new body is ever shown

Is neural video worth it?
827 h of video ≈ 105K L40 GPU-hours for +4–9 pts

Georgia Tech

GR00T N1, NVIDIA 2025 · GR00T N1.5 blog, NVIDIA 2025

GR00T N2: From VLA to World-Action Model

VLA · GR00T N1



"pick up the apple…"

VLM

Action head

Motor action ∿∿∿

"LVA": Language > Vision > Action

most parameters serve language; physics and verbs come last

WAM · GR00T 2 (N2)

ego | L wrist | R wrist

0.8 s history · 5 fps

Wan 2.1 VAE frozen

"Flip GR00T 1 to 2"

Cosmos-Reason1 frozen

robot state

MLP$_e$

One video DiT · 2B (Cosmos-Predict2.5)

video tokens: past → imagined future

noisy actions

action tokens attend to the imagined future

future frames, every camera

MLP$_e$

robot actions ∿∿∿

Vision ≈ Action

language joins through cross-attention

"If the video prediction works, the action works."

N1 · Mar '25: per-robot heads → N1.5 · Jun '25: Eagle 2.5, FLARE loss → N1.6 · Dec '25: Cosmos VLM, relative actions → N1.7 · Apr '26: relative EEF shared with humans → **N2 · Nov '26: world-action model**

Georgia Tech

Jim Fan, "Robotics' End Game", AI Ascent 2026 · GR00T 2 whitepaper, NVIDIA 2026 (Sec. 2, Fig. 1)

Xiaomi-Robotics-1

Scaling Vision-Language-Action Models with over 100K Hours of Real-World Trajectories

Xiaomi Robotics Team · 2026

Georgia Tech

Introduction:

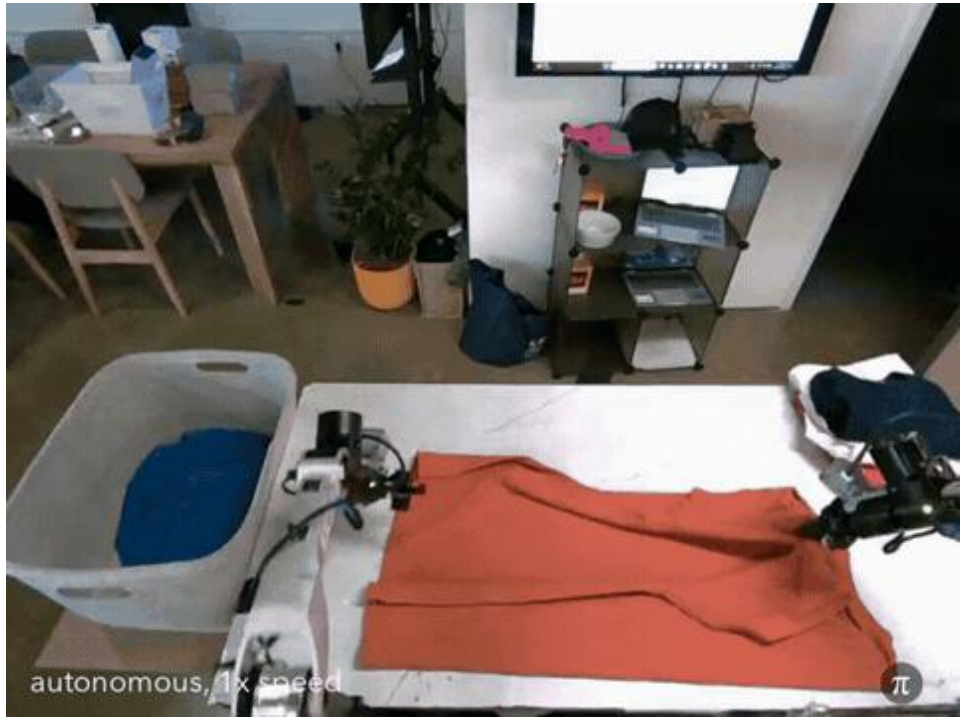
Generalization to unseen environments:
Perform familiar task types in unseen environments and with unfamiliar objects.
No additional task-specific fine-tuning at evaluation.

Adaptation to new tasks:
Learn new manipulation tasks from a small set of additional demonstrations.

Key challenge:
Collect diverse action data at scale and pair it with useful language annotations.

Related Works: Generalist VLAs

π0: a generalist robot policy



Benefit: Pretrained VLM knowledge + flow matching produce continuous robot actions.

Limitations: Complex tasks require additional fine-tuning, and performance remains uneven across tasks.

Georgia Tech

Related Works: Scalable Data Collection

UMI — Universal Manipulation Interface





Benefit: Portable demonstrations without a robot at each collection site.

Limitations: SLAM needs sufficient visual texture; Human labor required.

Georgia Tech

Related Works: World-Action Models

UWM — Unified World Models

Actual robot execution



Predicted dynamics



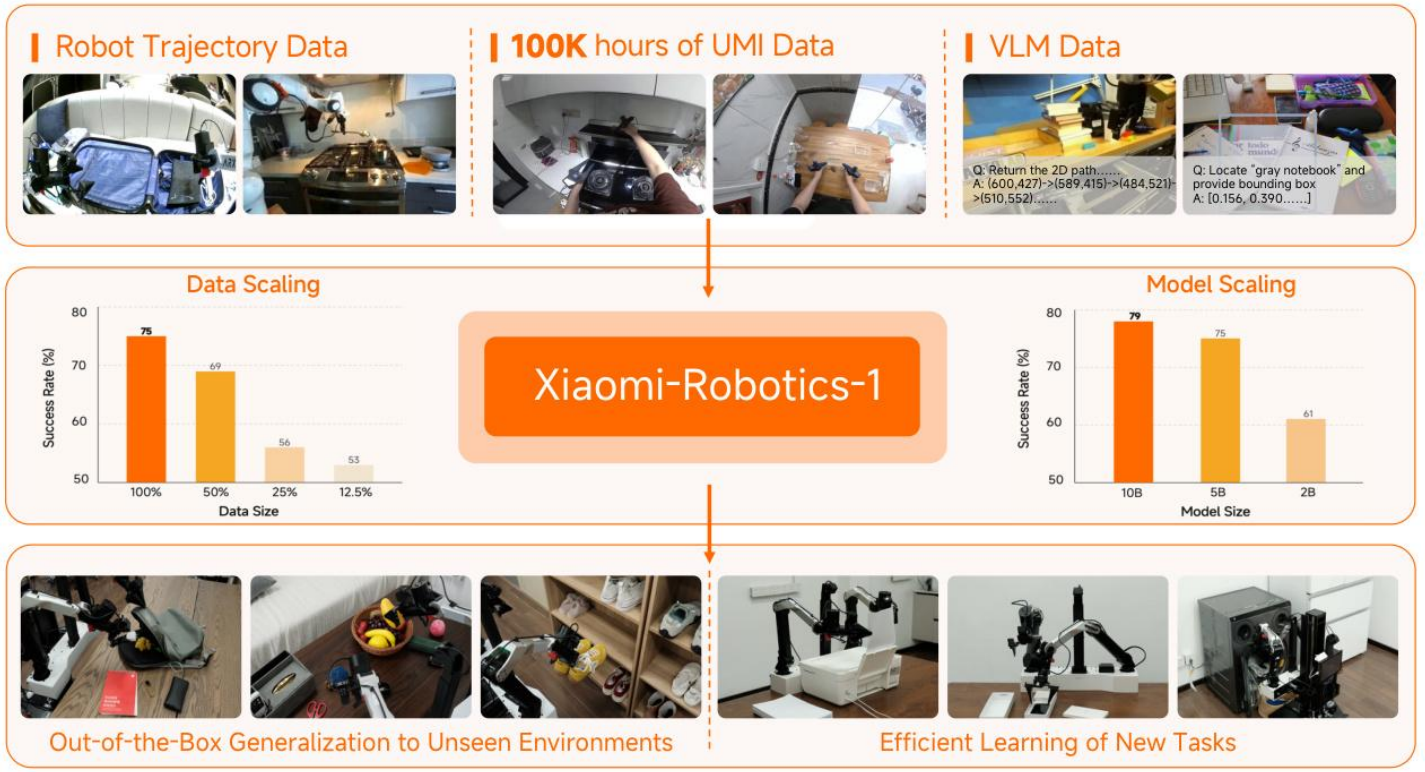
Benefit: Joint video–action diffusion learns from robot trajectories and action-free video.

Limitation: Errors in predicted dynamics can reduce their usefulness for planning.

Xiaomi focuses on data scale, automatic language annotation, and alignment to robot execution.

Georgia Tech

Xiaomi-Robotics-1: Overall Framework



1. Pre-training

100K+ hours of UMI data
State-transition captions

2. Post-training

~10K hours of mixed data
Human-style instructions

3. Deploy / adapt

Unseen environments
New-task fine-tuning

Georgia Tech

Policy Learning Formulation

Given an observation and a language instruction, predict an action chunk.

$$\max_{\theta} \mathbb{E}_{(\mathbf{o}_t, l, \mathbf{a}_{t:t+H}) \sim \mathcal{D}} \log \pi_\theta(\mathbf{a}_{t:t+H} \mid \mathbf{o}_t, l)$$

Inputs

o_t: current observation
l: language instruction

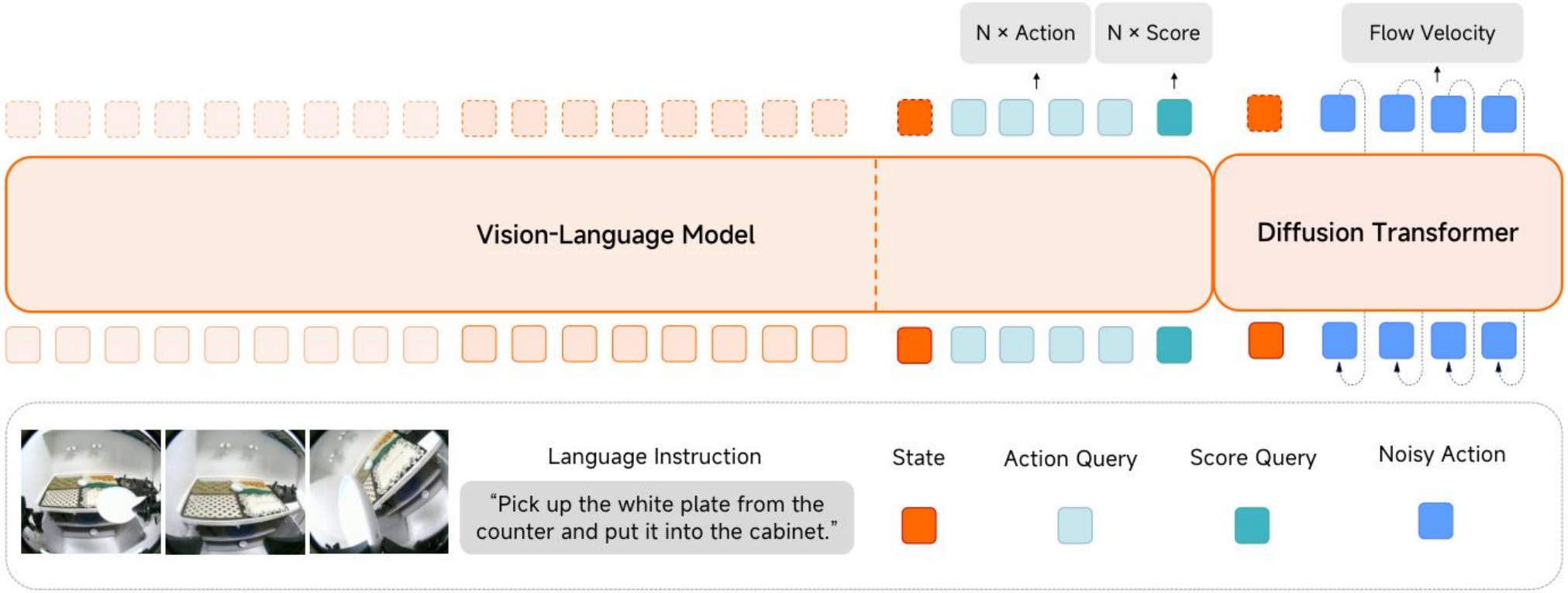
Prediction

A sequence of future actions
H: action-chunk horizon

Learn a policy that assigns higher likelihood to demonstrated action chunks.

Georgia Tech

Model Architecture



Qwen3-VL encodes images and language.
DiT generates action chunks conditioned on these features and robot state.

Georgia Tech

Action Generation and Auxiliary Supervision

DiT: flow-matching actions

$$L_{\text{Flow}}(\theta) = ||\mathbf{v}_\theta(\mathbf{o}_t, l, \mathbf{s}_t, \tilde{\mathbf{a}}^\tau_{t:t+H}, \tau) - \mathbf{u}(\tilde{\mathbf{a}}^\tau_{t:t+H}, \mathbf{a}_{t:t+H}, \tau)||_2^2$$

- Mix demonstrations with Gaussian noise.
- **Learn a flow velocity that moves the noisy action chunk toward the demonstrated action chunk.**
- At inference: 5 Euler steps, Δt = 0.2.

Five integration steps generate one action chunk.

VLM: Choice Policies

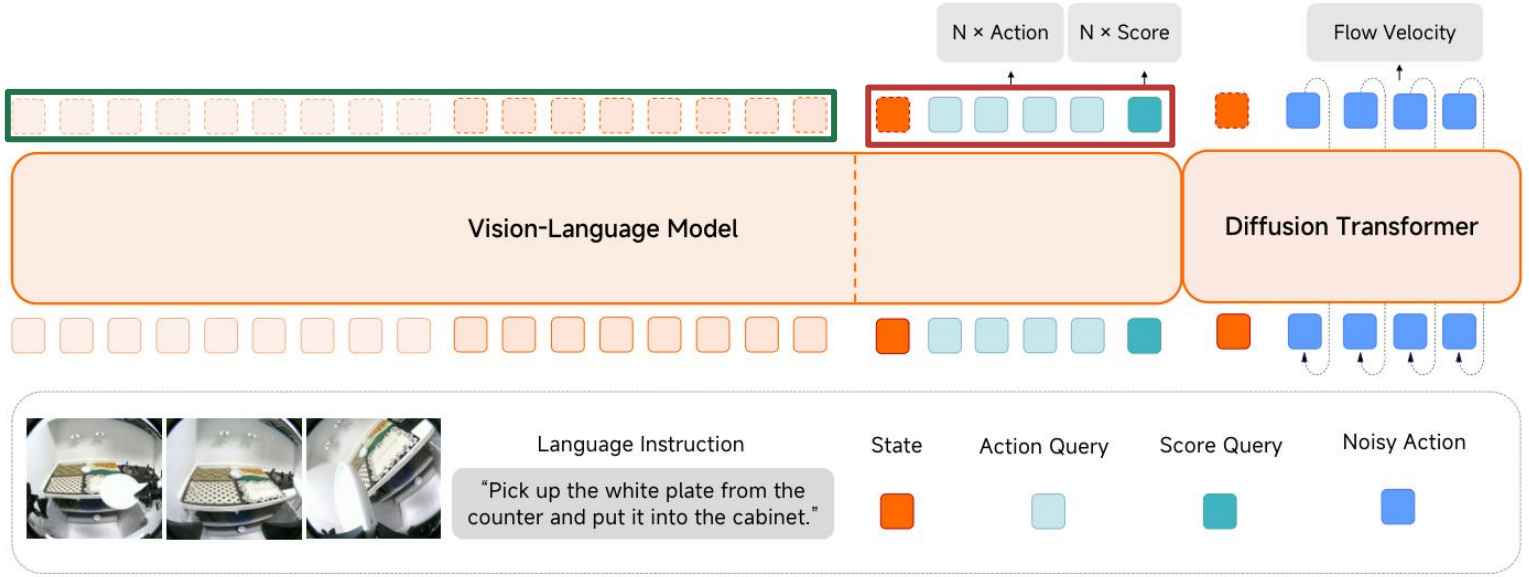
$$L_{\text{Regression}}(\theta) = ||\hat{\mathbf{a}}^*_{t:t+H} - \mathbf{a}_{t:t+H}||_1 + \sum_k^K ||\hat{s}_k - s_k||_2^2$$

- Predict K candidate action chunks and their scores.
- **Regress only the candidate closest to the demonstration in L1 distance (winner takes all).**
- Train every score to predict its candidate's L1 error.

Auxiliary VLM supervision improves action relevant representations and accelerates training convergence.

Georgia Tech

Preventing an Action-Copying Shortcut



Visual + language KV → **DiT**

Action-related VLM KV ✗ **DiT**

Observed: Attending to action-related VLM tokens degrades performance.

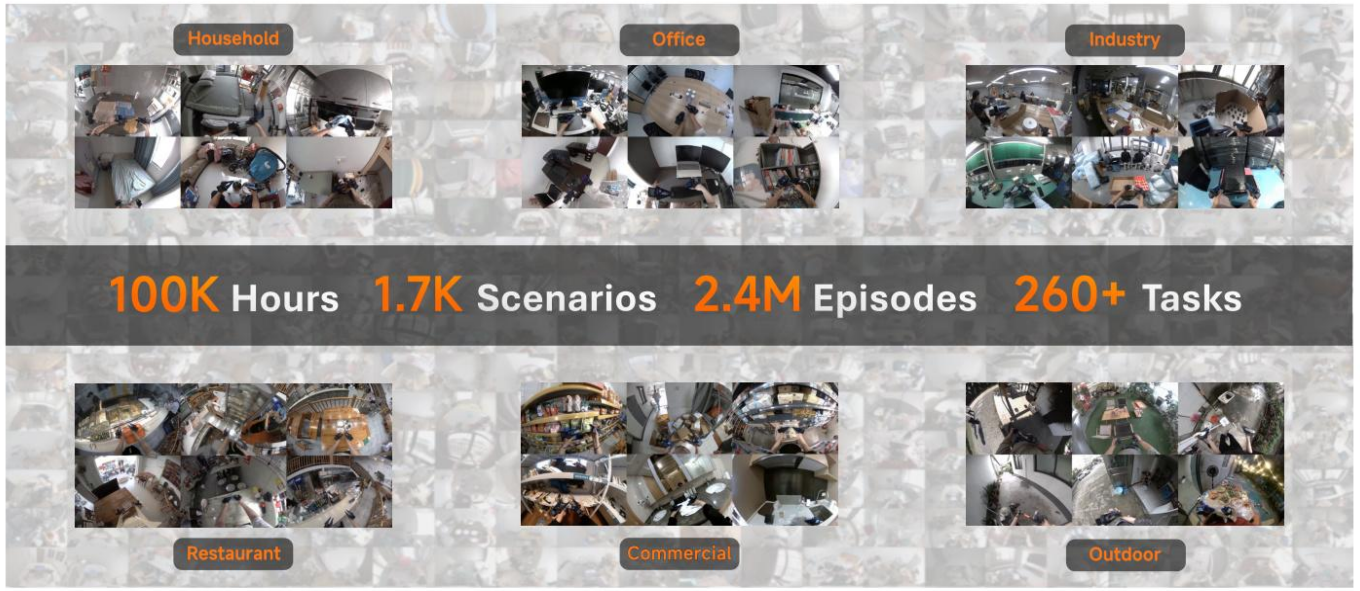
Authors' hypothesis: The DiT may copy VLM actions instead of grounding generation in images and language.

Attention restriction: Allow only visual and language representations from the VLM.

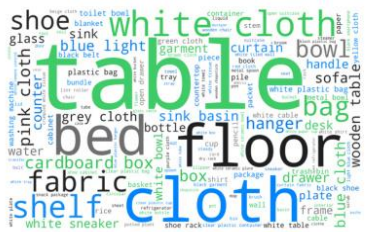
The DiT still receives robot state and noisy actions through its own inputs.

Georgia Tech

Pre-training Data: Scale and Diversity



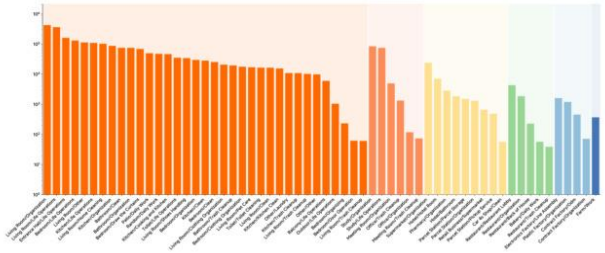
Word Cloud of Objects



Word Cloud of Verbs



Distribution of Environments & Tasks



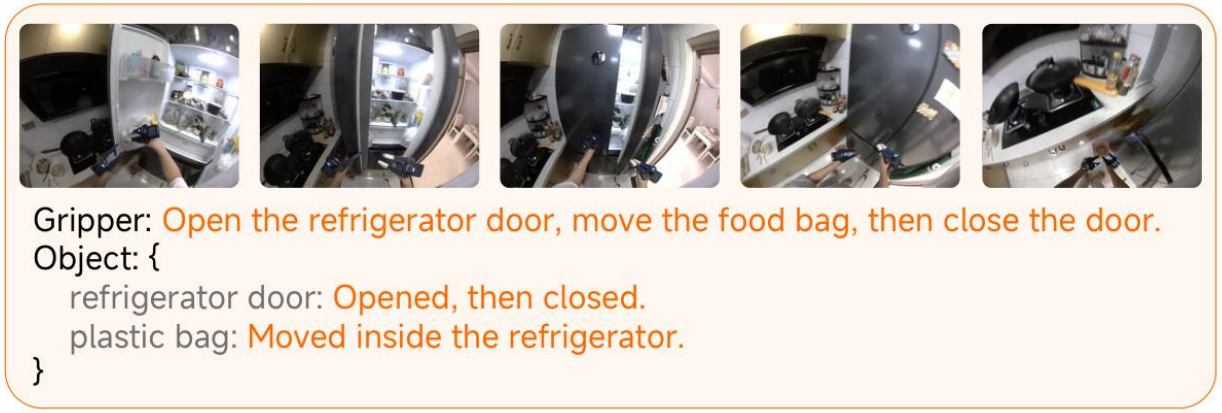
Pre-training Goal: Learn broadly transferable action-generation representations before adapting them to robot embodiments.

UMI enables large-scale collection across diverse real-world settings.

Georgia Tech

Automatic Annotation and Training Objective

Fixed-length trajectory clips → Qwen3.5-27B → state-transition captions



Captions describe gripper behavior and object changes.
Recorded clips are used for offline annotation.

Joint training objective

$$L = L_{\text{Flow}} + L_{\text{Regression}} + \lambda L_{\text{NTP}}$$

- Flow matching + Auxiliary regression + Vision–language next-token prediction
- L_{NTP}: Preserve the pretrained VLM's vision-language capabilities.
- λ = 0.1, VL : UMI sampling = 1 : 9

Parallel clip segmentation and captioning annotate over 100K hours of trajectories in roughly two weeks.

Georgia Tech

Post-training Data: Toward Robot Execution

~10K hours total: 7,200+ h robot data, 1,000+ h UMI with instruction labels, and Public robot datasets (BridgeV2, RT-1, DROID)



- Robot data: human-segmented clips with Qwen3.5-generated tasks instructions
- UMI data: human-annotated temporal segments and tasks instructions.

Transfer UMI manipulation skills to robot embodiments and align the policy with human task instructions.

Georgia Tech

Embodiment and Instruction Alignment

Embodiment alignment

All future end-effector targets are relative to the current EE pose at t.

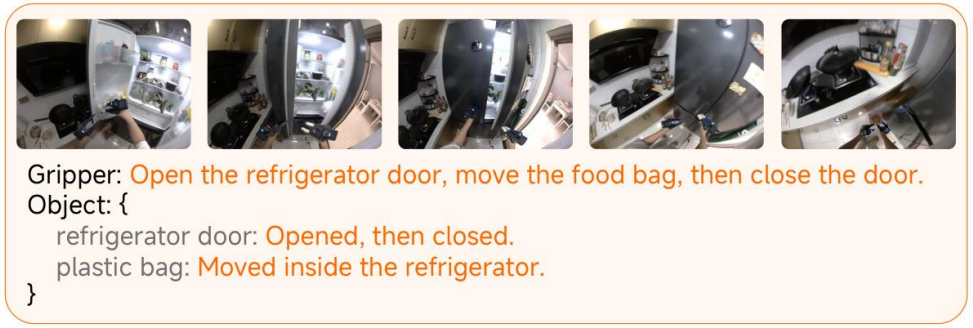
Aligned EE frame orientations give consistent motion values across UMI and robots in training stages.

Use base velocities and relative waist position changes.

Use a unified action vector and mask missing dimensions in the training loss.

Instruction alignment

Pre-training: condition on state-transition descriptions



Post-training: condition on imperative task instructions



Georgia Tech

Experiments: Four Questions

Q1: Does pre-training improve with more data and larger models?

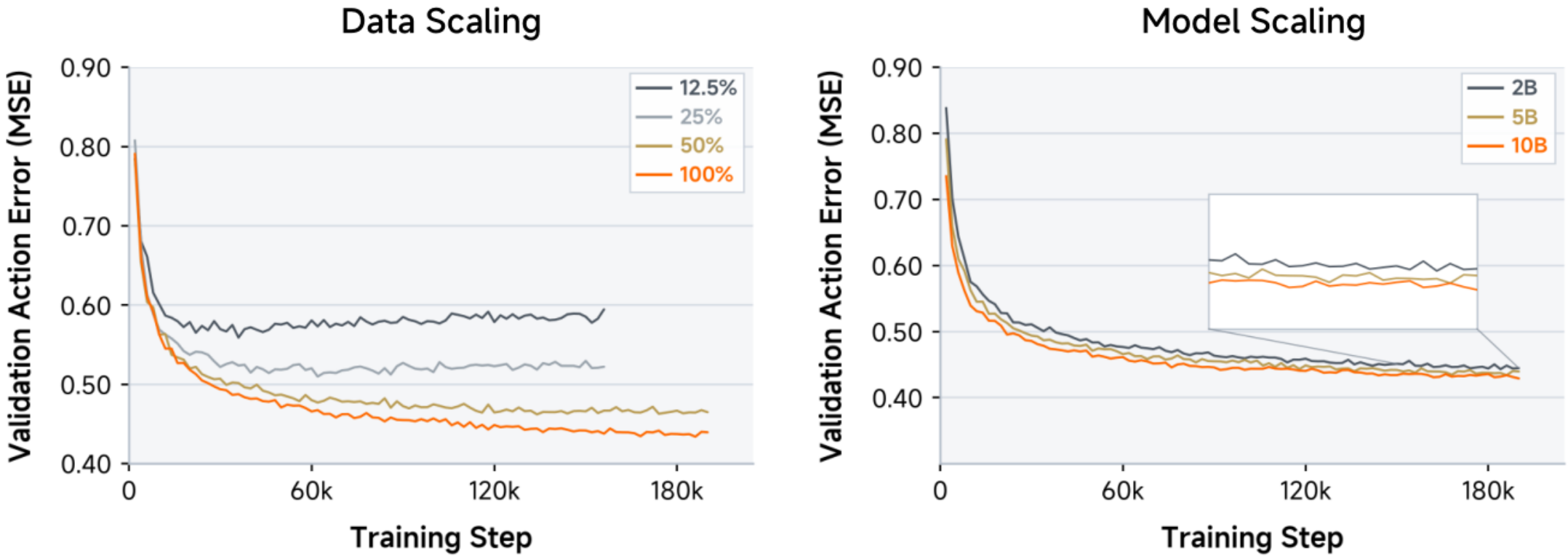
Q2: Does stronger pre-training improve out-of-the-box performance in unseen environments?

Q3: Can the model adapt to challenging new tasks with little additional data?

Q4: How does it compare with other robot foundation models on real robots and in simulation?

Georgia Tech

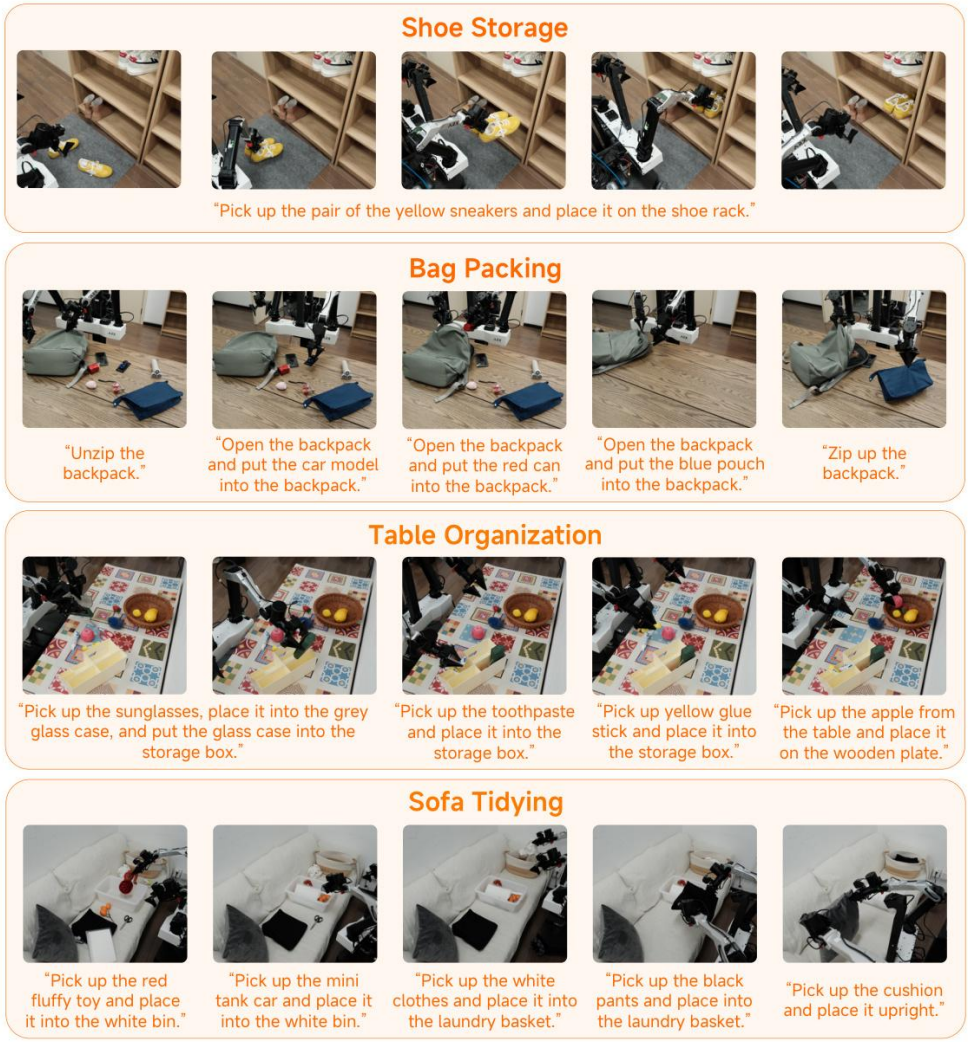
Q1: Pre-training Scales with Data and Model Size



Takeaway: Data availability appears to be a stronger bottleneck than model capacity.

Georgia Tech

Q2: Evaluation in Unseen Environments

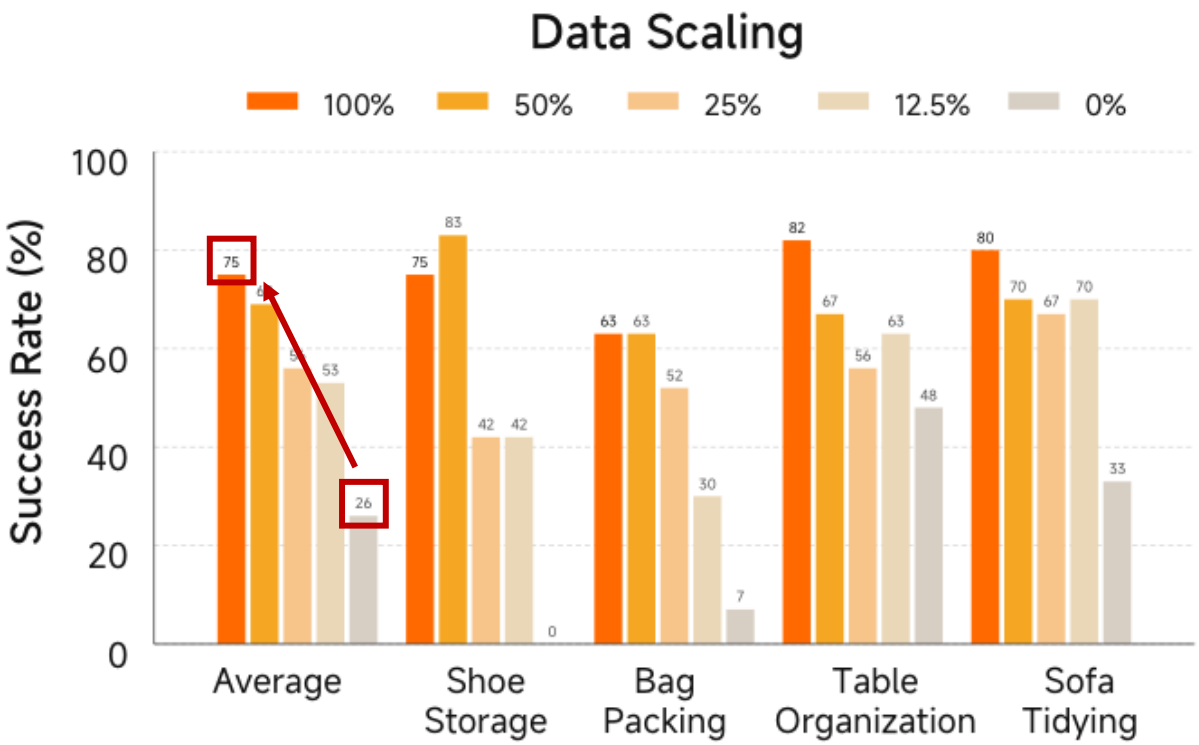


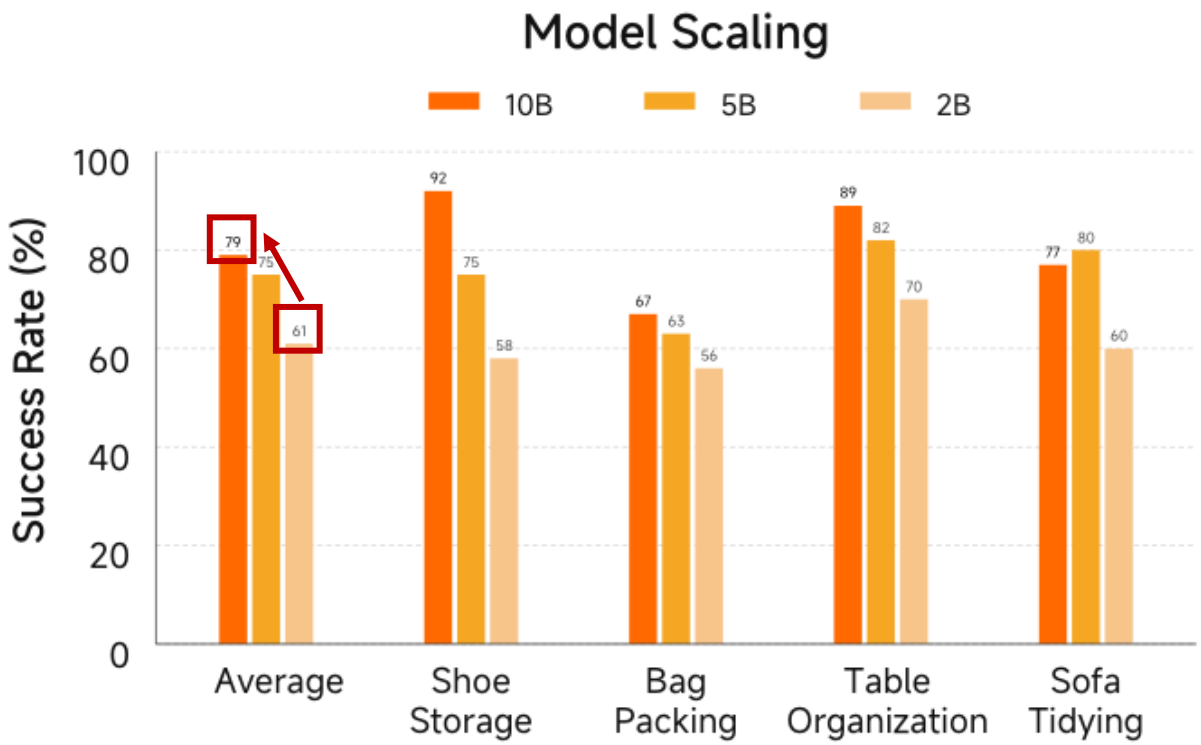


- Unseen environments and objects.
- Same post-training recipe.
- No additional task fine-tuning.

Georgia Tech

Q2 Results: Better Pre-training Transfers

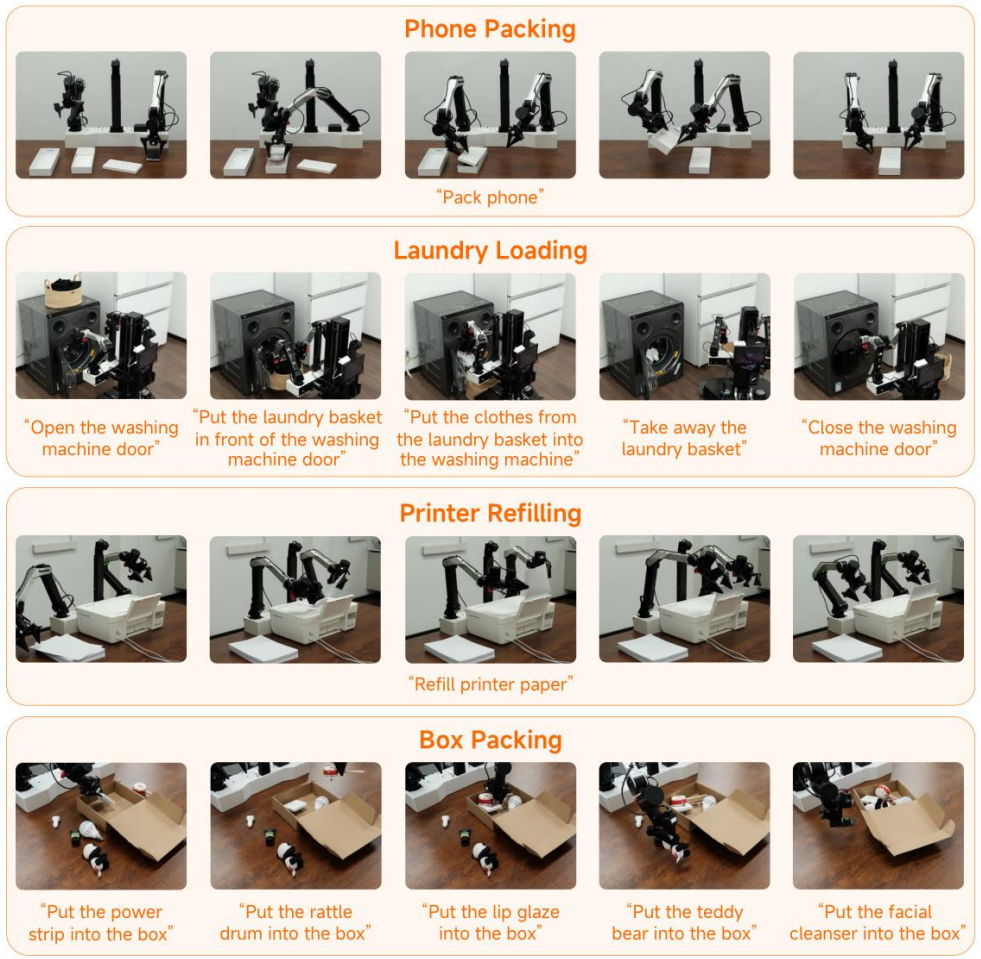




Takeaway: Gains from UMI pre-training survive the transition to robot execution.

Georgia Tech

Q3: Adaptation to Four New Tasks

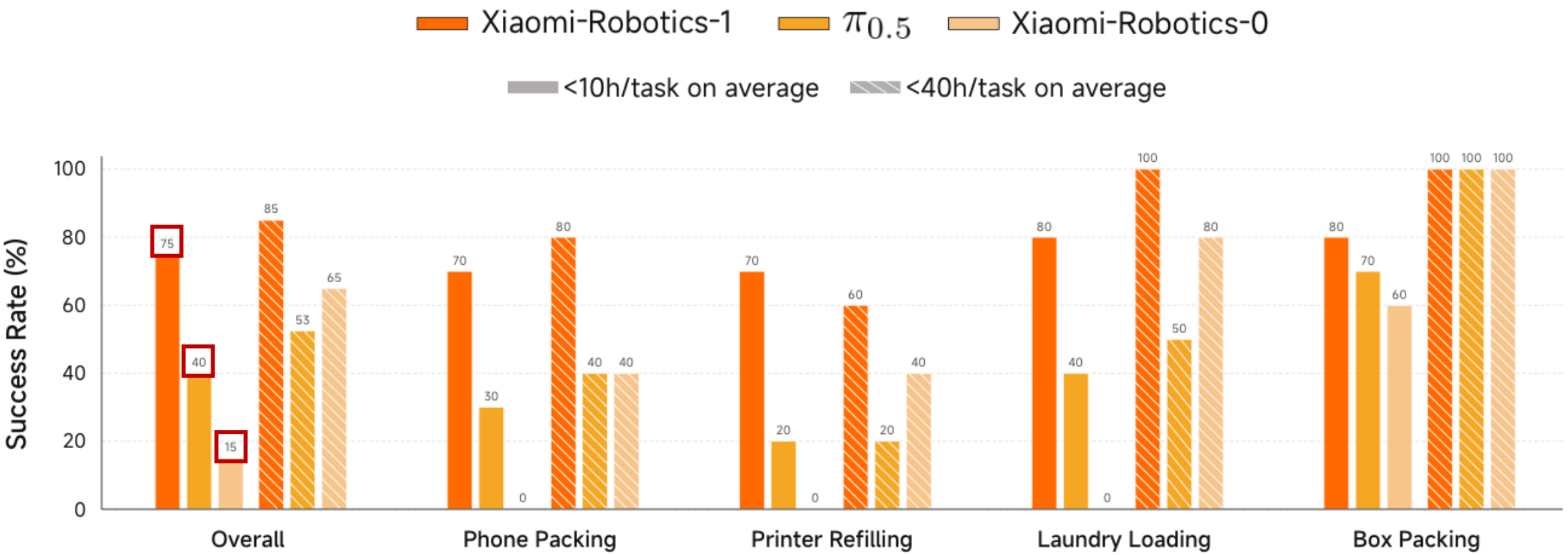




- Low: 36 h (~9 h/task) · High: 144 h
- Joint fine-tuning across four tasks
- 10 trials per task/model

Georgia Tech

Q3 Results: Adaptation with small amount of data



Takeaway: Strong pre-training supports adaptation to new tasks.

Georgia Tech

Q4: RoboCasa — Kitchen Manipulation

Method	Avg. Success (%)
UVA [41]	50.0
UWM [96]	60.8
$\pi_{0.5}$ [5]	62.1
π_0-FAST [58]	63.6
GR00T N1.6 [54]	66.2
Cosmos Policy [32]	67.1
RLDX-1 [29]	70.6
World2Act [73]	72.6
Xiaomi-Robotics-1 (Ours)	**74.5**

Evaluation:
24 tasks; 5 evaluation scenes

Unseen object instances

100 episodes per task

Takeaway: Xiaomi-Robotics-1 is competitive with leading VLA and WAM.

Georgia Tech

Q4: RoboCasa365 — Compositional Generalization

Method	Average	Atomic	Comp.-Seen	Comp.-Unseen
Diffusion Policy [16]	6.1	15.7	0.2	1.3
$\pi_{0.5}$ [5]	16.9	39.6	7.1	1.2
GigaWorld-Policy 0.1 [79]	20.7	44.4	11.8	2.9
GR00T-N1.6 [54]	21.9	51.1	9.4	1.7
WorldDreamer [75]	35.3	66.3	26.7	9.0
Qwen-RobotManip [71]	35.9	68.6	20.1	<u>14.9</u>
RLDX-1 [29]	36.0	67.6	27.9	8.5
ABot-M0.5 [11]	40.4	75.9	38.3	2.7
ABot-M0.6 [11]	<u>46.6</u>	<u>79.4</u>	<u>48.3</u>	7.9
Xiaomi-Robotics-1 (Ours)	**57.4**	**80.2**	**57.1**	**32.1**

50 evaluation tasks:
18 atomic

16 seen composite

16 unseen composite

Takeaway: The improvement on unseen task compositions suggests that Xiaomi-Robotics-1 can reuse learned skills in new task combinations.

Georgia Tech

Q4: VLABench — Multiple Generalization Tracks

ID, category, commonsense, instruction, and texture tracks · 2,500 rollouts

Method	In-dist.			Cross Category			Commonsense		
	SR ↑	PS ↑	IS ↑	SR ↑	PS ↑	IS ↑	SR ↑	PS ↑	IS ↑
ERVLA [61]	69.7	81.1	**84.2**	47.0	61.0	66.4	44.0	55.0	57.2
Xiaomi-Robotics-1 (Ours)	**75.6**	**85.0**	79.8	**53.0**	**66.6**	**66.4**	**48.4**	**58.3**	**58.2**

Method	Instruction			Texture			Avg.		
	SR ↑	PS ↑	IS ↑	SR ↑	PS ↑	IS ↑	SR ↑	PS ↑	IS ↑
ERVLA [61]	**58.0**	**70.2**	**73.8**	47.4	62.3	70.6	53.2	65.9	**70.4**
Xiaomi-Robotics-1 (Ours)	55.8	66.8	70.2	**62.6**	**74.9**	**74.8**	**59.1**	**70.3**	69.9

SR: success · PS: progress · IS: intention.
Benchmark-specific CoT supervision is added during training.

Takeaway: Category and texture gains suggest generalization to unfamiliar objects and visual conditions

Georgia Tech

Q4: RoboDojo — Five Capability Dimensions

Method	Generalization	Precision	Long-Horizon
GalaxeaVLA (G0) [26]	4.53 / 2.83%	8.10 / 3.83%	12.60 / 5.58%
GigaWorld-Policy [79]	5.34 / 2.89%	6.15 / 1.83%	15.51 / 8.92%
StarVLA-α [80]	3.93 / 2.33%	9.90 / 4.33%	14.15 / 6.50%
Xiaomi-Robotics-0 [8]	7.43 / 5.56%	8.42 / 4.58%	13.51 / 6.92%
X-WAM [21]	7.39 / 3.33%	6.72 / 1.83%	17.47 / 9.08%
X-VLA [92]	10.48 / 6.78%	18.32 / 12.00%	16.53 / 9.75%
$\pi_{0.5}$ [5]	13.37 / 8.17%	12.40 / 5.50%	23.54 / 14.67%
Spatial Forcing [35]	14.12 / 9.33%	17.33 / 10.58%	23.26 / 14.58%
Hy-Embodied-0.5-VLA [85]	11.77 / 8.39%	13.81 / 8.00%	25.74 / 14.92%
Xiaomi-Robotics-1 (Ours)	**23.55** / **17.00%**	**26.69** / **18.83%**	**38.39** / **23.67%**

Method	Memory	Open	Average
GalaxeaVLA (G0) [26]	3.17 / 1.89%	0.70 / 0.67%	5.82 / 2.96%
GigaWorld-Policy [79]	3.46 / 2.22%	0.54 / 0.50%	6.20 / 3.27%
StarVLA-α [80]	3.34 / 2.44%	0.68 / 0.58%	6.40 / 3.24%
Xiaomi-Robotics-0 [8]	5.07 / 3.67%	0.22 / 0.17%	6.93 / 4.18%
X-WAM [21]	6.32 / 4.67%	0.57 / 0.25%	7.69 / 3.83%
X-VLA [92]	4.76 / 3.56%	0.55 / 0.50%	10.13 / 6.52%
$\pi_{0.5}$ [5]	5.78 / 4.56%	1.98 / 1.67%	11.41 / 6.91%
Spatial Forcing [35]	5.43 / 4.11%	1.78 / 1.58%	12.38 / 8.04%
Hy-Embodied-0.5-VLA [85]	**13.37** / **12.11%**	0.65 / 0.58%	13.07 / 8.80%
Xiaomi-Robotics-1 (Ours)	7.81 / 6.56%	**3.94** / **3.58%**	**20.07** / **13.93%**

Entries report score / success rate.

Takeaway: Broad capability gains show versatility, while weaker memory suggests a need for observation history.

Georgia Tech

Conclusion



Pre-training gains appear in both generalization and less data adaptation.

Georgia Tech

What persisted and what changed?

Persisted: VLM + flow-matching DiT action head, action chunks, two-stage cross-embodiment training

	GR00T N1 (2025)	Xiaomi-Robotics-1 (2026)
Data	Data pyramid: web/human video, sim + neural trajectories, real teleop	100K+ h real-world handheld UMI trajectories
Action labels	Latent actions / IDM pseudo-labels for action-less video	Recorded by UMI; captions auto-labeled by Qwen3.5
Embodiments	Embodiment-specific state/action MLPs	One relative end-effector action space + masked dims
VLM–DiT link	DiT cross-attends to frozen LLM layer-12 features	Layer-wise KV sharing; VLM gets auxiliary action loss
Scope	2.2B; short-horizon tabletop humanoid tasks	2B–10B with scaling study; 10-min mobile manipulation

Why: Real robot data is costly, so GR00T N1 relied on simulation and on videos whose actions had to be inferred. Xiaomi records real actions with a cheap handheld gripper (no robot needed), so real data can scale to 100K+ hours.

Georgia Tech

Limitations & Societal Implications

Limitations

- **GR00T N1: short-horizon, tabletop only**
  Synthetic videos can violate physics
- **GR00T N1: post-training forgets skills**
  Lost the pre-trained bimanual hand-off skill
- **Xiaomi-R1: no observation history**
  Scores lower on memory tasks (RoboDojo)
- **Xiaomi-R1: hard to reproduce**
  Proprietary 100K h data; weights not yet out
- **Both: success rate is the main metric**
  Little on safety, force, or failure recovery

Societal Implications

- **Privacy**
  100K h of egocentric video from homes and offices
- **Labor**
  Automates home and factory work
- **Safety**
  General policies acting in homes need guarantees
- **Access**
  Open weights (GR00T) vs. data held by few firms

Discussion Questions

Q1 Real vs. synthetic data

Xiaomi scaled real UMI data while GR00T relied on synthetic and video data. Is synthetic data still needed for rare or unsafe cases?

Q2 Forgetting after fine-tuning

GR00T's post-trained policy lost a pre-trained bimanual skill. How do we adapt a robot model without erasing general behaviors?

Q3 Do scaling laws transfer?

Lower validation action error tracked better real-robot success here. Will offline loss stay a reliable proxy as models grow?