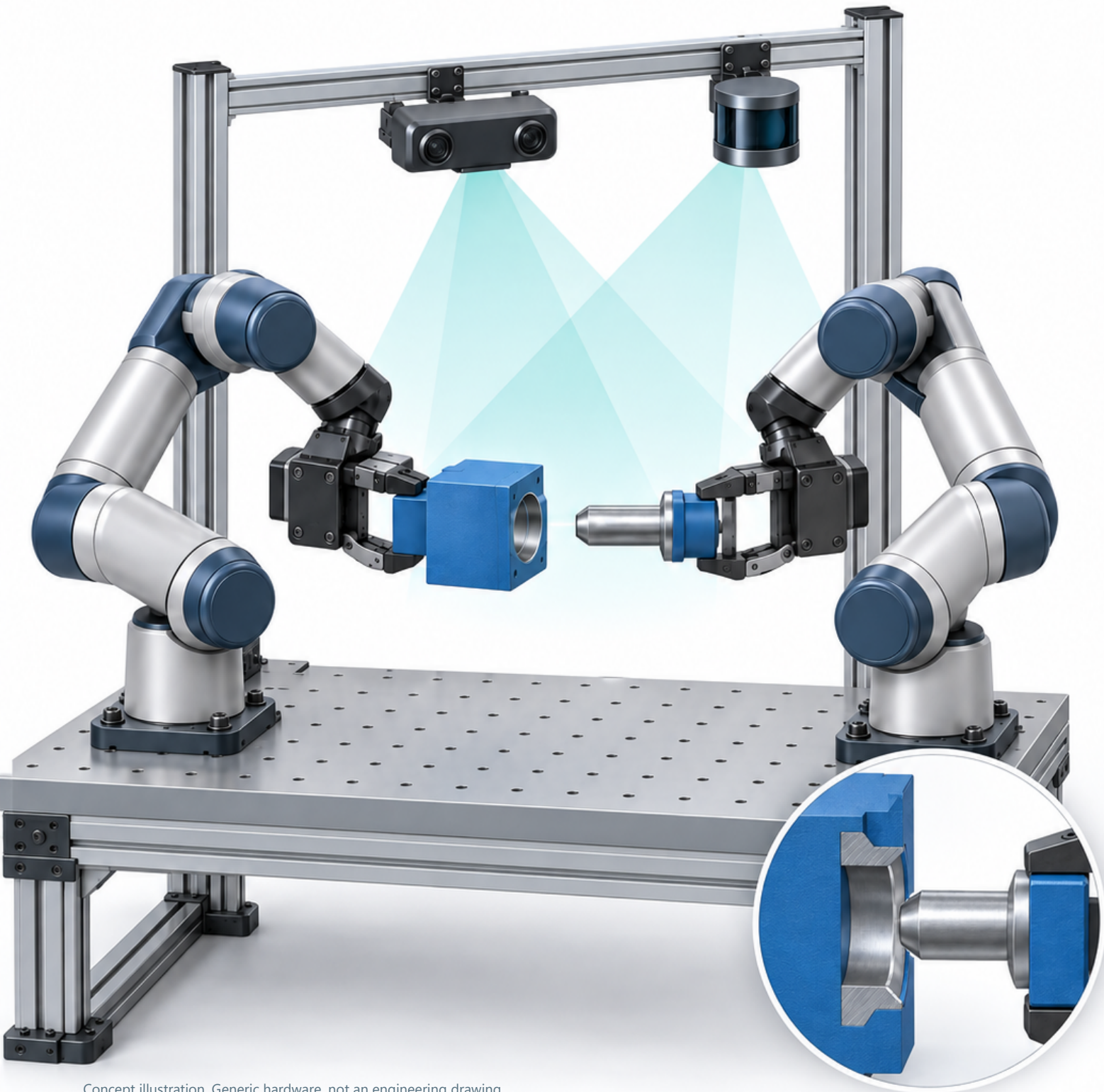


FMCW LiDAR for Dual-Arm Assembly

GR00T-based peg-in-hole research

Range, radial velocity and time for precise insertion



Concept illustration. Generic hardware, not an engineering drawing.

Project and research questions

The project will test whether FMCW LiDAR improves dual-arm peg insertion when it supports a pretrained vision-language-action model (VLA). A VLA uses images, instructions and robot state to generate actions.

Project status: This is a proposed system. The project has no completed model training or robot test results. All performance values below are targets or reference results.

Task and equipment

Two fixed-base arms will pick the parts and move them into a shared view. The left arm will hold the hole part. The right arm will align the peg and insert it under force control.

Component	Planned configuration
Robot arms	Two Unitree R1-7a arms, each with seven axes and a gripper.
Camera	Orbbec Gemini 335 [3]. Compare passive stereo with active stereo, which uses the projector.
FMCW LiDAR	100,000 points/s across the full field of view. Non-repeating Lissajous scan, 40° × 40° field, central work area.
Range specification	Nominal accuracy: 2 mm. The meaning of this specification and its error under task conditions need measurement.
Contact and teaching	Two wrist force/torque sensors, pending installation. PICO 4 Ultra controllers for demonstrations.
Compute	Two modified RTX 4090 GPUs, each with 48 GB of memory.

Three questions

Input	Benefit to test
Range	Can sparse range measurements improve alignment when stereo depth becomes unreliable?
Radial velocity	Can direct speed measurements reduce delay, overshoot or contact force during approach?
Time history	Can separate time stamps and motion correction improve coverage without excessive motion blur?

FMCW means frequency-modulated continuous-wave. This LiDAR uses coherent signals to measure range and radial velocity. Radial velocity measures motion along the laser line of sight. It does not measure full three-dimensional velocity.

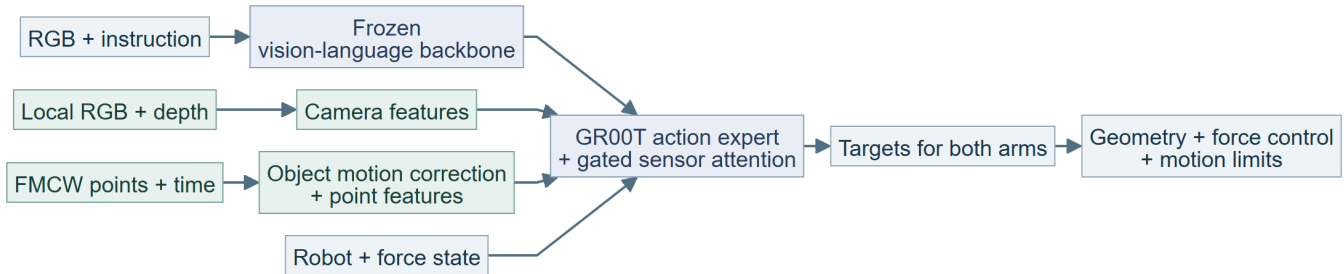
Expected value and physical limits

Range may help most when camera depth is unstable. Velocity may help most during approach and braking. Neither benefit is certain. Sideways motion can produce almost zero radial velocity [4]. Slow insertion gains depend on measured velocity noise.

A nominal 2 mm range specification does not establish submillimetre assembly accuracy. Calibration, stable grips, image features and compliant contact control must work together. LiDAR cannot replace force feedback inside the hole.

Model and sensor fusion

The main model is GR00T N1.7-3B, with about 3.14 billion parameters in the selected release [2]. The plan freezes its vision-language backbone. Training updates the full action head, about 1.62 billion parameters. New sensor modules have a combined budget of 30 million parameters.



Proposed data flow. Separate sensor paths let the experiments measure the value of each input.

The action expert uses flow matching. It learns to turn a noisy action sequence into a demonstrated sequence. Gated cross-attention lets it read camera and LiDAR features. These sensor paths are project additions to GR00T.

How sparse points enter the model

1. Retain range, ray direction, radial velocity, measurement time, arrival time and valid-data flags for each point.
2. Use only completed measurements that the system decodes before the decision time.
3. Correct historical points for each object's motion, using robot feedback and calibrated grip geometry.
4. Project each point into the matching image to sample local visual features.
5. Compress the two time windows into 64 compact summaries. Pass these summaries to the action expert.

The initial windows cover 0-20 ms and 20-100 ms before the decision. Each window accepts at most 1,024 points. Missing measurements retain explicit masks. Uncertain object tracking reduces the use of older points.

Pixel anchors remain part of the design. The model also retains physical values in a separate point path. A simpler sparse-image input will serve as a comparison. The model does not treat inferred depth as a measured return.

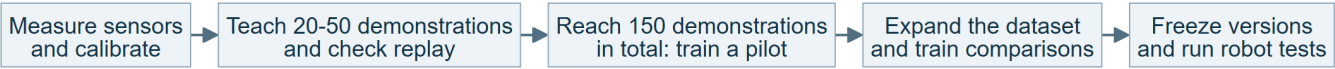
From model actions to robot motion

One policy controls both arms. Each step contains 14 values: position, rotation and gripper opening for each arm. Each action block covers 40 steps at 50 Hz, or 0.8 seconds. The controller executes only a short, current section.

The controller applies bounded geometry corrections, force-based compliance and motion limits. It rejects expired targets and handles faults without waiting for the model. Model updates at 10 Hz and external control at 100 Hz are targets that require hardware tests.

Public reference: NVIDIA reports 62.3% for Bridge and 72.5% for Fractal in SimplerEnv, after task-specific fine-tuning [1]. These simulation results do not establish dual-arm insertion performance.

Data, training and deployment



Development sequence. Data and hardware checks come before larger training runs.

Demonstration plan

An operator will teach both arms with PICO controllers. Start with 20-50 trajectories to check recording and replay. Then increase the total to about 150 trajectories for the first model. These trajectories count toward the full dataset if they meet its quality rules.

Purpose	Per camera mode	Two modes
Training: normal task	480	960
Training: correction and recovery	120	240
Validation	130	260
Offline test	130	260
Total accepted trajectories	860	1,720

The two modes are passive and active stereo. All sensors record together during each demonstration. LiDAR comparisons reuse the passive-mode trajectories. Active-mode recording uses matched task distributions in separate batches.

The 150-trajectory stage is a pilot, not the full manual-teaching budget. At a 75% acceptance rate, the plan needs about 2,300 attempts. At 3-6 minutes per attempt, collection requires about 115-230 hours, including resets and labels.

Separate training, validation and test data by session, collection day and physical part. Keep related trajectories in one partition. Recovery data need valid corrective actions. Keep sensor gaps with valid-data masks.

Training on two GPUs

Start with an Action Chunking Transformer (ACT) to check the data and action interface. Then train camera-only GR00T models. Add the range path before velocity and motion history. Each formal comparison starts from the same pretrained model with the same training budget.

The plan uses fine-tuning rather than pretraining from zero. Low-rank adaptation (LoRA) requires its own implementation and tests. It is not the initial method. Start with data-parallel training, one sample per GPU and gradient accumulation for an effective batch of 32.

Each GPU holds a full model copy. The two cards do not form one 96 GB memory pool. Run 200-500 trial updates to measure peak memory and training speed before fixing the final recipe.

Deployment and field work

Use Ubuntu 24.04 and ROS 2 Jazzy on the control host. Run training and inference in separate GPU containers. Local C++ control must continue without waiting for inference or file writes. Stop training during robot tests.

AI will support software development, training and analysis. On-site staff must install devices, calibrate the station, teach trajectories and reset parts. Measured data throughput and available field hours determine the schedule.

Experiments, targets and outputs

What the tests will compare

Group	External sensing
RGB baseline	RGB images only.
Passive stereo	RGB and passive stereo depth, including useful history.
Active stereo	RGB and projector-assisted stereo depth.
FMCW range	Passive stereo plus LiDAR range, time history and motion correction. No velocity input.
FMCW full	The same inputs plus radial velocity.
Geometry baseline	A conventional geometry method with the same controller.

All groups share robot feedback, force sensors, action definitions and control rules. Stereo baselines retain dense depth and motion estimates. Separate tests will remove velocity, explicit time features or motion correction. Each formal sensor comparison requires retraining.

Task targets and test budget

The initial peg diameter is 10 mm, with a 20 mm insertion depth. Start with 1.0 mm diametral clearance. The main task uses 0.5 mm clearance and a 0.5 mm × 45° entrance chamfer. Diametral clearance means hole diameter minus peg diameter.

The research target is at least 90% first-attempt success in the main task. A 0.2 mm clearance tests the capability limit. Neither value is a measured result. Success requires independent depth evidence, completion within the time limit and no force or torque limit breach, dropped part or jam.

The core plan contains 3,900 robot trials across static, low-texture and dynamic conditions. It uses three training seeds for the main models. These trials start with both parts already in the grippers. Tests of the full pick-and-insert sequence need separate results.

The 3,900 trials are autonomous evaluations, not extra manual demonstrations. Resets and records still require field work. The estimated station time is 130-325 hours, excluding preparation and repairs.

Measures and expected outputs

Report first-attempt success, recovery success, cycle time, alignment error, peak contact force, latency and GPU use. Keep aborted trials in the denominator. Report confidence intervals, seed variation and failures. Compare learning with 100, 300 and 600 training trajectories per camera mode.

Outputs will include the calibrated station, traceable datasets, trained baselines, sensor comparisons and a reproducible report. Range may improve difficult depth conditions. Velocity may improve dynamic approach. Tests must establish whether either gain exceeds the added compute and sensor cost.

Selected sources

[1] NVIDIA GR00T: SimplerEnv results and training context

[2] GR00T N1.7-3B: selected weights and parameter index

[3] Orbbec Gemini 335: camera specifications

[4] Need for Speed: limits of Doppler motion measurements

Source: Project Execution Manual V4.6, dated 3 October 2026. This brief describes the proposed research scope. Equipment error, training memory and task performance still require measurement.