MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

FurnitureVLA: Learning Long-Horizon Bimanual Furniture Assembly with Vision-Language-Action Model

Ma, Chenyang; Yang, Yue; Corcodel, Radu; Jain, Siddarth; Wu, Andrew; Hori, Chiori; Romeres, Diego

TR2026-174 November 13, 2026

Abstract

Current work on robot furniture assembly mostly focuses on toy-scale settings or single-arm manipulation. We introduce FurnitureVLA, the first systematic study of real-scale bimanual furniture assembly using Vision-Language-Action models (VLAs). We formalize the task, develop a scalable simulation pipeline for expert data generation and evaluation, and build a VR teleoperation system for single-operator bimanual control to collect high-quality real-world demonstrations. To address extreme long-horizon assembly with up to 7 subtasks and 1550 control steps, we propose a progress-enhanced VLA that jointly predicts actions and a continuous progress signal, enabling automatic subtask transitions and reducing compounding errors. FurnitureVLA improves average simulation success from 48% to 80% over baselines across three furniture types, with a further 21% gain from our study of perception and control design factors, and transfers to a real Kinova Gen3 platform with only 16% drop on the hardest task.

Conference on Robot Learning (CoRL) Workshop on The Action Gap: Large Generative Models Learn Rich Knowledge 2026

© 2026 MERL. This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

FurnitureVLA: Learning Long-Horizon Bimanual Furniture Assembly with Vision-Language-Action Model

Chenyang Ma[1,2] **Yue Yang**[1,3] **Radu Corcodel**[1] **Siddarth Jain**[1] **Andrew Wu**[1]
Chiori Hori[1†] **Diego Romeres**[4‡]
[1]Mitsubishi Electric Research Laboratories, Cambridge, MA, USA
[2]University of Oxford, Oxford, UK
[3]University of North Carolina at Chapel Hill, Chapel Hill, NC, USA
[4]Amazon, Boston, MA, USA
[†]Corresponding author. [‡]Work done while at MERL.

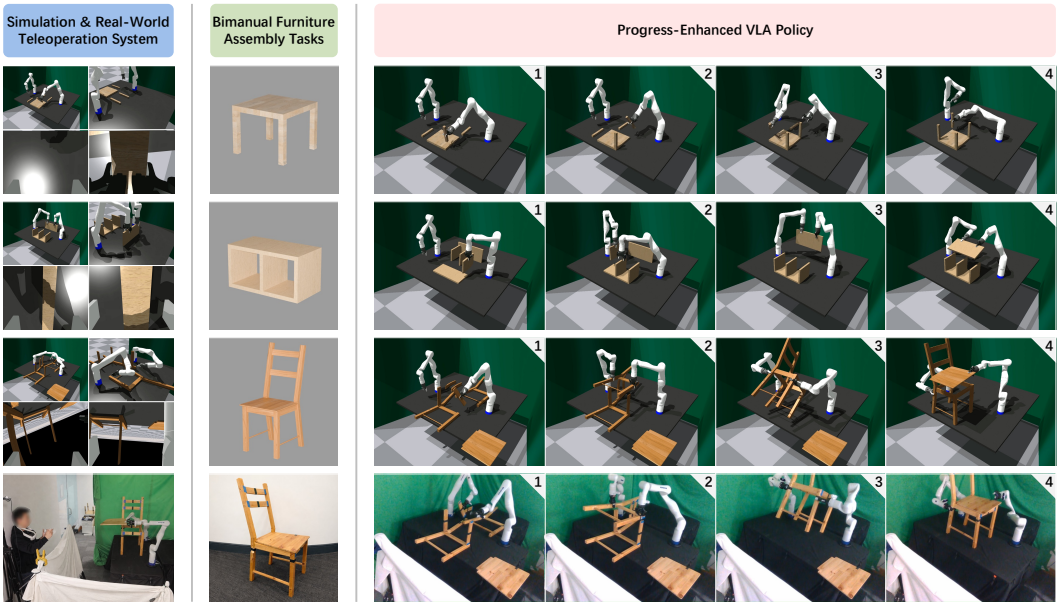


Figure 1: **Real-scale bimanual furniture assembly with Vision-Language-Action models.** We introduce **FurnitureVLA**, the first systematic study of this challenging setting. It comprises a scalable simulation pipeline for data generation and evaluation, and a tailored VR teleoperation system for high-quality demonstration collection. We propose a progress-enhanced VLA to tackle long-horizon assembly.

Abstract: Current work on robot furniture assembly mostly focuses on toy-scale settings or single-arm manipulation. We introduce **FurnitureVLA**, the first systematic study of real-scale bimanual furniture assembly using Vision-Language-Action models (VLAs). We formalize the task, develop a scalable simulation pipeline for expert data generation and evaluation, and build a VR teleoperation system for single-operator bimanual control to collect high-quality real-world demonstrations. To address extreme long-horizon assembly with up to 7 subtasks and 1550 control steps, we propose a progress-enhanced VLA that jointly predicts actions and a continuous progress signal, enabling automatic subtask transitions and reducing compounding errors. FurnitureVLA improves average simulation success from 48% to 80% over baselines across three furniture types, with a further 21% gain from our study of perception and control design factors, and transfers to a real Kinova Gen3 platform with only 16% drop on the hardest task. Project Page.

Keywords: VLAs, Bimanual Manipulation, Furniture Assembly

10th Conference on Robot Learning (CoRL 2026), Austin TX, USA.

1 Introduction

Furniture assembly is a long-standing challenge in robotics due to its long-horizon and high-precision nature. Assembling a piece of furniture requires manipulating multiple interdependent parts in strict order, where early mistakes can propagate and cause cascading failures [1–4]. Successful assembly also demands tight geometric alignment between mating parts [5–8]. These challenges are further amplified in real-scale settings: large, heavy components require sustained bimanual coordination, where two arms must jointly manipulate parts while respecting reachability limits and avoiding kinematic singularities throughout the assembly process [9, 10].

The goal of this work is to learn a generalist robot policy for real-scale bimanual furniture assembly using Vision-Language-Action models (VLAs), as shown in Fig. 1. Prior work focuses on toy-scale or scaled-down furniture with single-arm manipulation [1–8]. In this paper, we formalize real-scale bimanual furniture assembly tasks and build a system with two complementary components. First, we develop a scalable bimanual simulation pipeline that generates expert demonstrations via motion planning across diverse furniture types and task horizons. This controlled environment enables verification of task feasibility under reachability and kinematic constraints, and serves as a testbed for large-scale evaluation. Second, to support real-world deployment, we develop a VR teleoperation system with design principles tailored for real-scale bimanual assembly, enabling coordinated dual-arm control by a single teleoperator on a Kinova Gen3 platform for high-quality demonstration collection.

With this system in place, directly finetuning a monolithic VLA policy on full-length bimanual demonstrations remains suboptimal, as current VLA models are most effective on short-horizon tasks with relaxed geometric precision requirements [11, 12]. Long-horizon trajectories induce distribution drift, where small deviations can drive the policy into out-of-distribution states and lead to compounding errors [13, 14]. To address this, we decompose each assembly into semantically grounded subtasks and finetune the VLA on shorter, language-conditioned segments. This narrows the within-subtask state distribution and provides cues for task stages. Importantly, we define subtask boundaries at stable, contact-free post-retreat arm states rather than at the contact-rich states immediately after part assembly. Since post-retreat states are free of contact and force constraints, small execution errors are less likely to amplify into large deviations, yielding a more consistent initial state distribution for each subsequent subtask and reducing cross-subtask distribution shift. However, this requires determining when to transition between subtasks during execution. We address this by introducing a progress-enhanced VLA that jointly predicts actions and an explicit subtask progress signal to trigger transitions during inference.

Finally, to improve precision in real-scale assembly, we conduct a focused study of key perception and control design factors, including action horizon, temporal ensembling [15], image resolution, and camera viewpoints, and analyze how these choices affect assembly success. In summary, we introduce **FurnitureVLA**, with the following contributions:

- We present the first systematic study of VLAs for real-scale bimanual furniture assembly and develop a system comprising a scalable simulation pipeline for expert demonstration generation via motion planning and large-scale evaluation, together with a VR teleoperation system with design tailored for real-scale bimanual assembly to enable high-quality real-world data collection.
- We propose a progress-enhanced VLA that jointly predicts actions and a subtask progress signal, enabling stable long-horizon execution via automatic subtask transitions, along with a focused study of design factors for improving precision.
- We demonstrate improved long-horizon stability and assembly precision in simulation, improving average success from 48% to 80% compared to baselines across three furniture types with an additional 21% gain from our design factor study, and validate on a real Kinova Gen3 platform with only 16% drop on the hardest task.

2 Related Work

Robot Furniture Assembly. Robot furniture assembly couples long-horizon planning with tight geometric constraints. Early work relies on engineered perception and task-and-motion planning [16], while later work scales this by translating assembly manuals into structured plans with learned contact corrections and vision-language alignment [4, 17, 18]. Building on simulation benchmarks [1, 19], learning-based methods study large-scale imitation learning [3], residual policies for precision [6], sim-to-real transfer with limited real data [5, 8, 20], and human-robot collaboration [21]. However, existing work largely targets single-arm, task-specific policies in toy-scale settings, leaving open how to train a generalist policy for real-scale bimanual assembly.

VLAs for Long-Horizon Manipulation. Long-horizon execution remains brittle due to error accumulation and covariate shift [13, 14, 22, 23]. A common strategy is to decompose long demonstrations into shorter phases. Some works learn modular skills (e.g., manipulation, movement) with external switching [24–26], while others structure VLAs into reusable modules that replan and recover under a failure detector [27, 28]. A complementary line equips VLAs with reasoning to output the next subtask during execution [29–31], but at higher computational cost. Our progress-enhanced VLA enables automatic subtask transitions during inference without an external stage estimator.

VLAs for Precise Manipulation. Generalist robot policies (e.g., VLAs [11, 12, 30, 32]) often remain precision-limited: cross-embodiment standardization can coarsen action representations [33], while discretized action tokenization can fail on dexterous, high-frequency data [34], as observed in large-scale real-robot evaluations [35]. ACT [15] uses action chunking and temporal ensembling to reduce compounding error and smooth control, while Openpi Comet [36] shows that perception and control-loop design choices materially affect manipulation accuracy. We study these factors in real-scale bimanual assembly and provide insights for geometric alignment and assembly success.

3 Tasks and System Setup

Our goal is to study real-scale bimanual furniture assembly. To this end, we first formalize the task setting and introduce key assumptions (Sec. 3.1). We then describe our system, which comprises a scalable simulation pipeline for data generation and evaluation (Sec. 3.2), and a VR teleoperation system for high-quality real-world data collection (Sec. 3.3).

3.1 Bimanual Furniture Assembly Tasks

Hardware Setup. Assembly is performed on a tabletop using two Kinova Gen3 7-DoF robot arms, with a Robotiq Hand-E gripper on the left arm and a Robotiq 2F-85 gripper on the right. The tabletop schematic is shown in Fig. 3. In addition to the front and wrist-mounted cameras commonly used in prior VLA work [30], we add a rear camera to provide complementary views, as large furniture parts and frequent bimanual interactions often occlude frontal observations.

Assumptions. We make two key assumptions. First, we use magnets to attach furniture parts once they are aligned, bypassing the screwing process, which is an orthogonal and more challenging problem [37, 38] and difficult to simulate in physics-based simulators. Prior assembly benchmarks similarly avoid screwing [1, 19]. Second, each furniture part is randomly initialized within 3 cm in position and 5° in orientation from its nominal pose, consistent with common settings in VLAs [11, 30, 32].

Task Properties. We study real-scale bimanual furniture assembly using three IKEA items of increasing difficulty: LACK side table, KALLAX shelf, and IVAR chair. We define subtask decompositions with corresponding language instructions that are concise, descriptive, and semantically grounded, as shown in Fig. 2. Our tasks exhibit the following properties that make them particularly challenging:

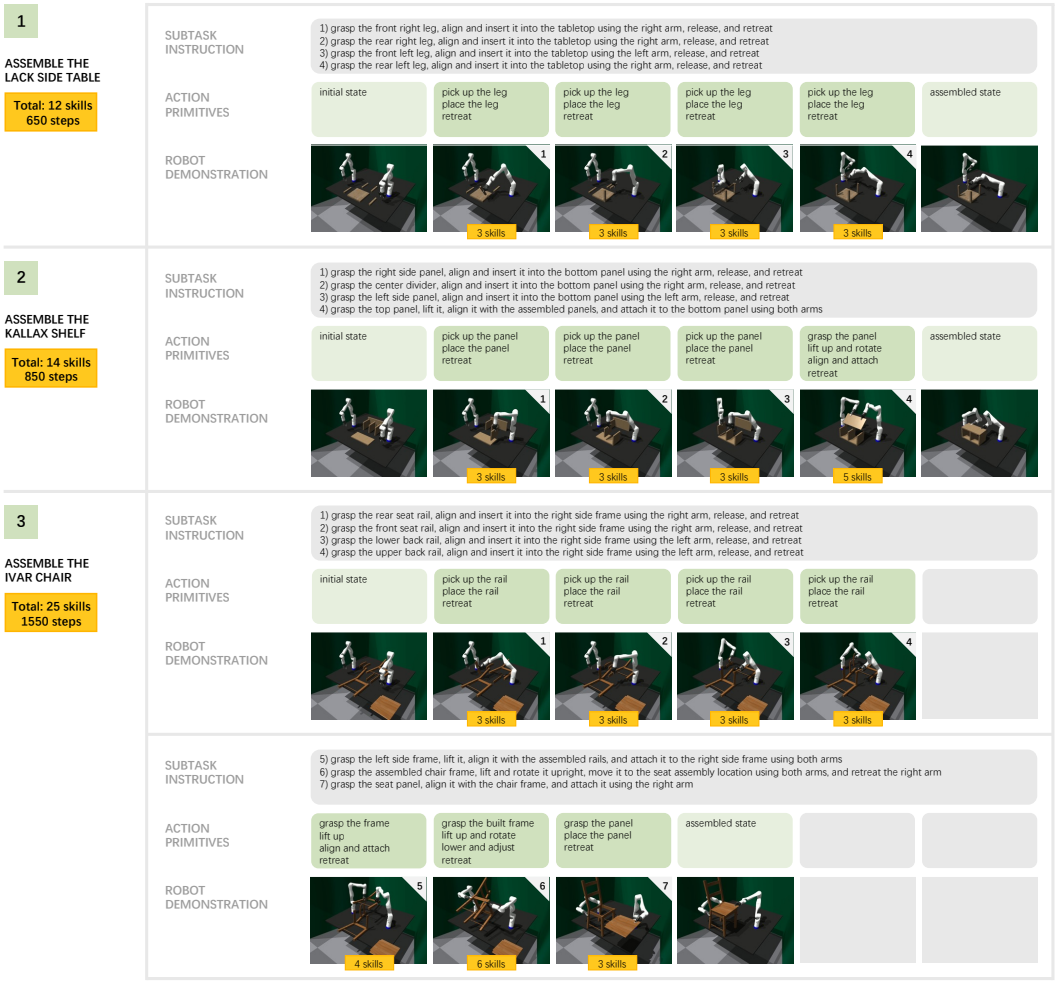


Figure 2: **Bimanual furniture assembly tasks.** These tasks require executing diverse manipulation skills over extremely long horizons. A simple item (LACK side table) requires 12 skill executions (650 steps), while a complex assembly (IVAR chair) requires 25 (1550 steps).

1) Long-horizon execution. Each assembly requires completing a strict sequence of subtasks, with horizons ranging from 4 (LACK, KALLAX) to 7 (IVAR), and 650–1550 control steps (65–155 seconds), exceeding prior benchmarks [39–41].

2) High-precision insertion. Each subtask requires precise alignment between mating parts under tight translation and rotation thresholds, stricter than prior toy-scale assembly benchmarks [1, 19] (see Sec. 3.2).

3) Diverse manipulation skills. Each subtask is specified by a structured language instruction (e.g., "grasp, align, and insert a part") and requires composing multiple manipulation skills, such as grasping, alignment, insertion, lifting, and rotation. As a result, a simple assembly (e.g., LACK with 4 subtasks) involves roughly 12 skill executions, while more complex assemblies, such as IVAR, can involve 25 skills due to longer horizons and additional bimanual operations such as coordinated lifting and rotation of assembled structures. These skills must be performed under varying conditions, including occlusion, contact-rich insertion, and collision-aware maneuvering around partially assembled structures.

4) Bimanual coordination. Several subtasks require both arms to act in concert over large, heavy parts. Attaching the KALLAX top panel requires simultaneously lifting, rotating, and aligning a full-width panel with both arms; assembling IVAR requires lifting and rotating the entire partially-assembled chair frame upright before attaching the seat panel. These bimanual subtasks involve up to four sequential action primitives and must maintain reachability and avoid singularities throughout.

3.2 Simulation Data Generation and Success Criteria

Data Generation via Motion Planning. We use motion planning to generate expert demonstrations in simulation. For single-arm actions, we plan end-effector pose trajectories directly. For bimanual actions, we disable physics simulation and instead plan trajectories for the manipulated object, constraining both arms to follow as rigidly attached end-effectors. All generated trajectories are validated by replaying them in simulation and verifying successful assembly. Initial part placements are carefully designed to ensure feasibility: all parts remain within robot reachability, avoid inter-object collisions during manipulation, and prevent configurations that lead to kinematic singularities.

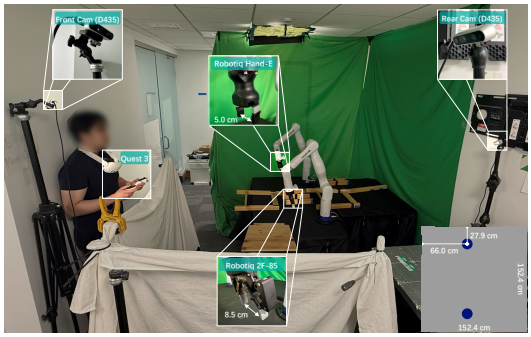


Figure 3: **System setup.** The teleoperator wears a Meta Quest 3 headset at the neck to track hand poses. A green screen reduces visual noise; a white cover occludes the teleoperator. Bottom-right: tabletop schematic with symmetric dual Kinova placement.

Assembly Success Criteria. Following FurnitureBench [1], we designate one part in each furniture as the base part and evaluate assembly by measuring the relative pose error of every other part with respect to this base. Let $\mathbf{T}_b, \mathbf{T}_k \in SE(3)$ denote the poses of the base part and the k-th part, respectively. We compute the relative transformation:

$$\mathbf{T}_{\text{rel}} = \mathbf{T}_b^{-1}\mathbf{T}_k, \tag{1}$$

and compare it against the ground-truth assembled pose $\mathbf{T}^*_{\text{rel}}$. A part is considered assembled if:

$$|t_{\text{rel},i} - t^*_{\text{rel},i}| \le \epsilon,\ i \in \{x, y, z\}, \qquad \cos\big(\mathbf{r}_{\text{rel},j}, \mathbf{r}^*_{\text{rel},j}\big) \ge \delta,\ j \in \{u, v, w\}, \tag{2}$$

where $t_{\text{rel},i}$ denotes the i-th component of the translation vector and $\mathbf{r}_{\text{rel},j}$ denotes the j-th column of the rotation matrix of $\mathbf{T}_{\text{rel}}$.

We set $\epsilon = 1$ cm for small parts and $\epsilon = 2$ cm for large parts, and $\delta = 0.998$ ($\approx 4.0^\circ$ per rotation axis), aligning with FurnitureBench [1] but more demanding given the larger scale of real-world furniture. A full assembly is considered successful only when all parts satisfy both criteria.

3.3 Real-World Data Collection via VR Teleoperation

To support real-world deployment, we develop a VR teleoperation system for high-quality demonstration collection, enabling coordinated dual-arm control by a single teleoperator on a Kinova Gen3 platform. The detailed setup is shown in Fig. 3. Given the complexity of long-horizon, high-precision bimanual assembly, we adopt the following design principles to make teleoperation precise, efficient, and easy to operate:

1) Decoupled translation and rotation control. All three translational axes are controlled simultaneously for smooth end-effector motion, while rotation is handled independently to improve stability and precision. This is critical for precise insertion subtasks, such as aligning rails in IVAR, where small rotational errors can lead to failure.

2) Pre-defined grasp primitives. We define a set of grasp poses with discrete 90° orientation variants, triggered via button inputs. Upon activation, the end-effector orientation snaps to a preset while maintaining its current position, removing manual wrist rotation and enforcing consistent, task-specific configurations. These presets enable stable grasps of slender rails, flat seat panels, and large side frames, reducing operator burden and improving demonstration consistency.

3) Synchronized bimanual control. We introduce a synchronized mode in which both arms execute mirrored commands simultaneously, enabling efficient repositioning, alignment, and rotation of large and heavy furniture components. This is critical for subtasks such as jointly lifting and rotating the IVAR chair frame, where coordinated motion maintains stability and enables accurate placement.

4 FurnitureVLA: Progress-Enhanced VLA

We propose a progress-enhanced VLA finetuned on semantically grounded subtasks to mitigate distribution drift in long-horizon tasks, jointly predicting actions and a progress signal to trigger subtask transitions (Fig. 4). We first describe continuous progress assignment using action primitives with post-retreat subtask boundaries (Sec. 4.1), then how it enables subtask transitions during inference (Sec. 4.2).

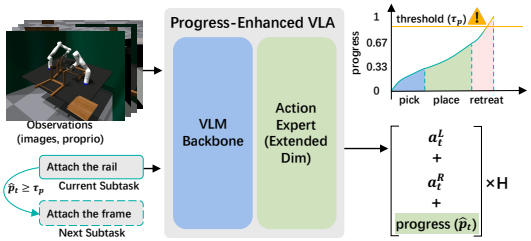


Figure 4: **FurnitureVLA: progress-enhanced VLA.** Each assembly is decomposed into subtasks with continuous progress labels derived from action primitives. The VLA is finetuned to jointly predict actions and a progress signal, which triggers automatic subtask transitions during inference.

Preliminaries. A VLA policy π_θ maps an observation o_t (RGB images and proprioception) and a language instruction g to robot actions. We represent actions in a continuous absolute end-effector space. For bimanual control, actions for the left and right arms are defined as $a_t^L, a_t^R \in \mathbb{R}^7$, where each action is defined as $[x_t, y_t, z_t, u_t, v_t, w_t, \gamma_t]^\top$, representing the target end-effector pose and gripper state. The full action is formed by concatenation, $a_t \in \mathbb{R}^{14}$. The policy uses flow matching [30] to decode a chunk of H future actions in a single forward pass: $\pi_\theta(a_{t:t+H-1} \mid o_t, g)$.

4.1 Progress-Enhanced Subtask Finetuning

Progress Assignment via Action Primitives. We decompose the task g into subtasks $G = (g_1, \ldots, g_K)$ and augment the 14-dim bimanual action a_t with a scalar progress signal p_t, giving $\tilde{a}_t = [a_t^\top, p_t]^\top \in \mathbb{R}^{15}$. Each subtask comprises N_k action primitives (Fig. 2), such as pick up, place, and retreat, which discretize progress into N_k fixed stages of completion (Fig. 4). Let s_i denote the start timestep of the i-th primitive, with s_{N_k} the end of the subtask. Segment $i \in \{0, \ldots, N_k - 1\}$ occupies the uniform interval $\left[\frac{i}{N_k}, \frac{i+1}{N_k}\right]$, and progress is computed as:

$$p_t = \frac{i}{N_k} + \frac{1}{N_k} \cdot \frac{t - s_i}{s_{i+1} - s_i}, \quad s_i \le t < s_{i+1}. \tag{3}$$

This yields a monotonic signal from 0 to 1 over each subtask. Each segment corresponds to a fixed control objective, while the motion evolves smoothly within the segment, making linear interpolation a natural time-parameterization of progress. We finetune the policy via flow matching on augmented action chunks, $\pi_\theta(\tilde{a}_{t:t+H-1} \mid o_t, g_k)$, to jointly predict actions and progress.

Post-Retreat Subtask Boundaries. We define subtask boundaries after retreat rather than immediately after assembly. Transitioning at assembly completion is fragile: small deviations can cause incomplete insertion or unstable contact, leading to failure. Contact-rich post-assembly states are also highly sensitive to execution errors, so small deviations produce large state variations across rollouts, widening the initial-state distribution of the next subtask. By contrast, post-retreat states are free of contact and force constraints. Small errors are less likely to amplify, yielding a narrower, more consistent initial-state distribution and reducing cross-subtask distribution shift.

4.2 Inference and Subtask Transition

At inference time, the policy predicts $\tilde{a}_{t:t+H-1}$ conditioned on the current subtask g_k. The predicted progress $\hat{p}_t$ triggers subtask transitions when exceeding a threshold. To suppress spurious predictions, we apply a lightweight filter. With a high signal $\hat{p}_t \ge \tau_p$, a transition is triggered if:

$$\text{TRANSIT} = \begin{cases} 1, & h_t \wedge h_{t-1}, \\ 1, & h_t \wedge \neg h_{t-1} \wedge \neg h_{t-2} \wedge \exists\, \Delta \ge 3 :\ h_{t-\Delta}, \\ 0, & \text{otherwise}, \end{cases} \qquad h_t = (\hat{p}_t \ge \tau_p). \tag{4}$$

This filters isolated spikes while remaining responsive to persistent signals. Upon confirmation, the system advances to g_{k+1}, resets progress, and clears action buffers.

5 Simulation Experiments

We examine: 1) how the progress-enhanced VLA improves long-horizon execution; 2) how key perception and control design factors affect assembly success and 3) ablations of design choices.

Implementation. For simulation, we use Isaac Gym [42], extending FurnitureBench [1], with furniture models from 3D Warehouse [43], textures from ambientCG [44], and assets processed in Blender 5.0 [45]. We use $\pi_{0.5}$ [30] as the VLA backbone, finetuned for 40,000 steps on 8 NVIDIA L40S GPUs with global batch size 64. Inference runs on a single L40S GPU with $\tau_p = 0.95$.

5.1 Long-Horizon Assembly Performance

Setup. We generate 500 demonstrations per furniture for finetuning and evaluate on 100 rollouts each, training a single VLA model across all furniture types. Unless otherwise specified, we use a default configuration for all design factors (see Sec. 5.2 for details).

Baselines. We compare against two baselines: $\pi_{0.5}$ (zero-shot), and $\pi_{0.5}$ (monolithic finetuned), which uses the unmodified $\pi_{0.5}$ backbone finetuned on full demonstrations with a single global instruction (e.g., "assemble the IVAR chair" as in Fig. 2) without subtask decomposition or progress modeling.

Table 1: **Assembly performance** (success rates ↑).

Method	LACK	KALLAX	IVAR	Average
$\pi_{0.5}$ (zero-shot)	0.00	0.00	0.00	0.00
$\pi_{0.5}$ (monolithic finetuned)	0.91	0.11	0.41	0.48
FurnitureVLA	**0.98**	**0.85**	**0.56**	**0.80**

Evaluation Metrics. We report the success rate of full assembly (i.e., completing all subtasks). If any preceding subtask fails, the rollout terminates, resulting in a monotonically decreasing success rate across subtasks.

Analysis of Task Performance. From Table 1, $\pi_{0.5}$ (zero-shot) achieves zero success across all furniture, as our tasks are highly challenging and out-of-distribution from its pretraining data. FurnitureVLA consistently outperforms $\pi_{0.5}$ (monolithic finetuned), with the largest gain on KALLAX. We attribute this to its large, heavy parts, which make the policy particularly sensitive to singularities induced by long-horizon drift and compounding errors. Among the three furniture, LACK is the easiest (0.98 success rate), while IVAR is the most challenging (0.56). We think this is reasonable, as more complex assembly tasks can lead to a large drop in success, as observed in prior work [1]. Per-subtask success rates in Fig. 5 further reveal where failures concentrate: for IVAR, the fifth subtask is the hardest, requiring both arms to grasp, lift, and attach the chair frame to the partially assembled frame.

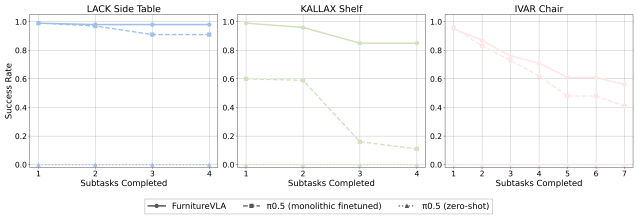


Figure 5: **Subtask success rates** (monotonically decreasing).

5.2 Design Factors for Assembly Precision

Setup. We vary the following perception and control design factors. **1) Action horizon:** $\pi_{0.5}$ predicts chunks of H=50 actions; we execute only the first 5, 10, or 25 actions before replanning, where fewer steps enable more frequent replanning and better error recovery at the cost of higher overhead and reduced trajectory smoothness. **2) Temporal ensembling:** We maintain a rolling buffer of B overlapping action chunks and compute the executed actions as a weighted average of all action predictions (excluding the progress dimension) for the current timestep [15], where $a_t^{[t-i]} \in \mathbb{R}^{14}$ is the prediction for timestep t made i steps ago:

$$\hat{a}_t = \frac{\sum_{i=0}^{B-1} w_i \, a_t^{[t-i]}}{\sum_{i=0}^{B-1} w_i}, \quad w_i = e^{\lambda i}. \tag{5}$$

Table 2: **Impact of perception and control design factors** on assembly precision (success rates ↑).

Furniture	Temporal Ensembling						Action Horizon			Viewpoint			Resolution		
	n/a	-0.25	-0.1	0.0	0.1	0.25	5	10	25	full	depth	w/o rear	224	300	448
LACK	0.76	0.90	**0.98**	0.91	0.91	0.92	0.92	**0.98**	0.75	**0.98**	0.69	0.45	0.59	0.97	**0.98**
KALLAX	0.83	0.81	**0.85**	0.44	0.84	0.80	0.43	0.67	**0.85**	**0.85**	0.45	0.57	0.77	0.68	**0.85**
IVAR	0.36	0.53	**0.56**	0.52	0.55	0.54	0.44	**0.56**	0.41	**0.56**	0.48	0.34	0.43	0.50	**0.56**
Average	0.65	0.75	**0.80**	0.62	0.77	0.75	0.60	**0.74**	0.67	**0.80**	0.54	0.45	0.60	0.72	**0.80**

Table 3: **Ablation studies on demonstration quantity and progress signal design** (success rates ↑).

Method	LACK	KALLAX	IVAR	Average
Discrete Progress	0.00	0.00	0.00	0.00
FurnitureVLA (25% demos)	0.53	0.63	0.33	0.50
FurnitureVLA (50% demos)	0.82	0.78	0.44	0.68
FurnitureVLA (100% demos)	**0.98**	**0.85**	**0.56**	**0.80**

Table 4: **Real-world IVAR chair assembly performance** (success rates ↑) over sequential subtasks S1–S7. Full-assembly SR monotonically decreases, while per-part SR evaluates each subtask independently,

Metric	S1	S2	S3	S4	S5	S6	S7
Full Assembly SR	0.80	0.73	0.60	0.53	0.47	0.47	0.40
Per-Part SR	0.80	0.80	0.73	0.80	0.67	0.87	0.80

The progress signal $\hat{p}_t$ always uses the most recent prediction, independent of the ensembling buffer. More negative λ emphasizes recent predictions, while $\lambda{=}0$ gives uniform averaging. **3) Observation viewpoint:** We evaluate removing the rear camera or replacing it with front-view depth. **4) Observation resolution:** We vary input resolution across $224{\times}224$, $300{\times}300$, and $448{\times}448$.

Impact of Design Factors. From Table 2, temporal ensembling consistently improves assembly success, with $\lambda = -0.1$ performing best across all furniture. We attribute this to the need for stable control in real-scale bimanual assembly: large, heavy parts and coordinated bimanual motion benefit from smoothing. At $\lambda = -0.1$, roughly 70% of the weight is assigned to the most recent prediction, with the remaining distributed across prior predictions, providing a good balance between responsiveness and stability. Action horizons of 10 and 25 perform best depending on the furniture; notably, KALLAX favors a longer horizon (25), likely because its largest and heaviest parts amplify instabilities and require stronger smoothing. For perception, the rear camera outperforms front-view depth and removing the rear view, indicating its effectiveness in mitigating occlusions. Finally, higher image resolution consistently improves performance, highlighting the importance of visual precision.

5.3 Ablation Studies

We study the effects of demonstration scale and an alternative discrete progress supervision scheme. In this variant, each subtask $k \in \{1, \ldots, K\}$ is assigned a fixed scalar value $(2k-1)/(2K)$, so all observations within a subtask share the same constant progress. At inference, the subtask is determined by discretizing progress into K bins. From Table 3, performance improves consistently with more data, with the largest gain from 25% to 50%. The smaller improvement from 50% to 100% suggests our demonstration budget achieves a good balance between data collection cost and performance. Discrete progress fails completely (zero success across all furniture). We observe that after subtask completion, the model fails to advance the progress signal and becomes stuck. We attribute this to the visual similarity between states where a part is nearly assembled and assembled, making discrete transitions difficult to detect. This supports our continuous progress design, which provides a smooth and unambiguous supervision signal throughout each subtask.

6 Real-World Experiments

We validate on a real bimanual Kinova Gen3 platform. We test on the most challenging IVAR chair.

Implementation. We use Quest2ROS [46] to map VR controller poses and inputs to robot end-effector commands, and place colored markers (e.g., black and blue tape) on parts to aid the teleoperator's visual alignment during demonstration collection. We filter no-op actions following DROID [47], but keep the initial and trailing frames of each non-idle segment, as they carry critical signals of subtask completion. Training, inference, and parameters follow simulation.

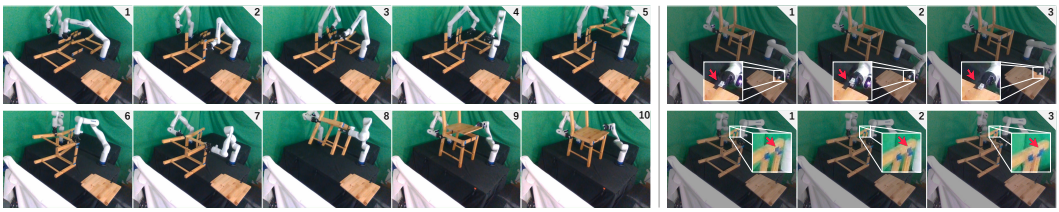


Figure 6: **Qualitative real-robot rollouts.** (Left) successful long-horizon assembly of the IVAR chair. (Right) emergent self-correction behaviors: regrasping for improved contact and fine-grained magnet alignment.

6.1 Full Assembly Performance

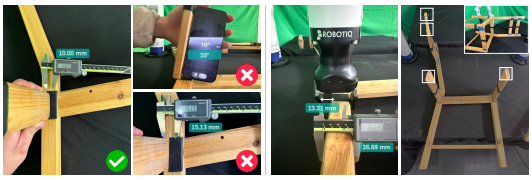


Figure 7: **Real-world precision requirements.** (Left) assembly tolerance with magnets. (Right) gripper clearance and multi-magnet alignment challenges.

Setup and Evaluation. We collect 100 demonstrations for the IVAR chair and evaluate on 15 rollouts. Magnets are installed in screwing holes. Assembly requires high precision even with magnets (Fig. 7 left): a 1 cm deviation can snap successfully, but 1.5 cm or a 10° tilt causes parts to fall apart. Additional precision demands (Fig. 7 right) include limited gripper clearance (5 cm Robotiq Hand-E gripper opening vs. up to 3.7 cm part thickness) and multi-point alignment when engaging up to 8 magnets. Following Sec. 5.1, we report full-assembly success rate with monotonically decreasing per-subtask success.

Results. Table 4 (row 1) shows that real-world performance is slightly lower than in simulation (Fig. 5), as expected due to increased difficulty and noise in real settings. Subtasks 3–5 are the main bottlenecks: subtasks 3 and 4 suffer from reduced visual detail as the left arm operates farther from the camera, while subtask 5 requires the highest spatial precision due to multi-point alignment.

Emergent Behavior and Qualitative Results. Fig. 6 (left) shows qualitative examples of successful real-robot assembly. We also observe emergent corrective behaviors (Fig. 6 right): in several rollouts, the robot self-corrects when parts are initially misaligned. For example, after grasping the seat panel with insufficient contact, the robot reopens the gripper, adjusts its pose, and regrasps for a more stable hold; when attaching the left chair frame, it performs small corrective motions to align the magnets before insertion. We attribute these behaviors to similar corrections in the teleoperated demonstrations, which the policy successfully learned.

6.2 Per-Part Assembly Performance

We evaluate each subtask independently. We initialize robot and scene at the corresponding intermediate assembly state and run 15 rollouts per subtask. From Table 4 (row 2), per-part success rates are consistently higher than full assembly rates, confirming that failures accumulate across subtasks rather than stemming from any single catastrophic failure mode. Subtask 5 remains the most challenging because of its high precision demands.

7 Conclusion and Discussion

We introduce **FurnitureVLA**, the first systematic study of real-scale bimanual furniture assembly with VLAs. We build a scalable simulation pipeline and VR teleoperation system for large-scale testbed and data collection, propose a progress-enhanced VLA for improved long-horizon execution, and conduct a focused study of VLA design factors for assembly precision. We see this work as a step toward deploying VLAs in challenging, real-world manipulation tasks.

Limitations. Our fixed-base setup limits assembly to furniture within the robot's kinematic workspace; mobile bimanual platforms could handle larger items. We also bypass screwing with magnets, extending to tool use remains an open challenge requiring significantly higher precision.

References

[1] M. Heo, Y. Lee, D. Lee, and J. J. Lim. Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. In *Robotics: Science and Systems*, 2023.

[2] H. Lin, R. Corcodel, and D. Zhao. Generalize by touching: Tactile ensemble skill transfer for robotic furniture assembly. In *International Conference on Robotics and Automation*, 2024.

[3] L. Ankile, A. Simeonov, I. Shenfeld, and P. Agrawal. JUICER: data-efficient imitation learning for robotic assembly. In *International Conference on Intelligent Robots and Systems*, 2024.

[4] Y. Tian, J. Jacob, Y. Huang, J. Zhao, E. Gu, P. Ma, A. Zhang, F. Javid, B. Romero, S. Chitta, et al. Fabrica: Dual-arm assembly of general multi-part objects via integrated planning and learning. In *Conference on Robot Learning*, 2025.

[5] X. Zhang, C. Wang, L. Sun, Z. Wu, X. Zhu, and M. Tomizuka. Efficient sim-to-real transfer of contact-rich manipulation skills with online admittance residual learning. In *Conference on Robot Learning*, 2023.

[6] L. Ankile, A. Simeonov, I. Shenfeld, M. Torne, and P. Agrawal. From imitation to refinement - residual rl for precise assembly. In *International Conference on Robotics and Automation*, 2025.

[7] M. Noseworthy, B. Tang, B. Wen, A. Handa, C. C. Kessens, N. Roy, D. Fox, F. Ramos, Y. Narang, and I. Akinola. FORGE: force-guided exploration for robust contact-rich manipulation under uncertainty. *Robotics and Automation Letters*, 2025.

[8] B. Huang, J. Xu, I. Akinola, W. Yang, B. Sundaralingam, R. O'Flaherty, D. Fox, X. Wang, A. Mousavian, Y. Chao, and Y. Li. Vt-refine: Learning bimanual assembly with visuo-tactile feedback via simulation fine-tuning. In *Conference on Robot Learning*, 2025.

[9] Z. Fu, T. Z. Zhao, and C. Finn. Mobile ALOHA: learning bimanual mobile manipulation using low-cost whole-body teleoperation. In *Conference on Robot Learning*, 2024.

[10] T. Li, J. Truong, J. Yang, A. Clegg, A. Rai, S. Ha, and X. Puig. Robotmover: Learning to move large objects from human demonstrations. *arXiv preprint arXiv:2502.05271*, 2025.

[11] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, 2024.

[12] Generalist AI Team. Gen-0: Embodied foundation models that scale with physical interaction. https://generalistai.com/blog/nov-04-2025-GEN-0, 2025.

[13] S. Ross, G. J. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*, 2011.

[14] Y. Yang, L. Zhao, M. Ding, G. Bertasius, and D. Szafir. BOSS: benchmark for observation space shift in long-horizon task. *Robotics and Automation Letters*, 2025.

[15] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems*, 2023.

[16] X. Zhou and Q. Pham. Can robots assemble an IKEA chair? *Science Robotics*, 2018.

[17] C. Tie, S. Sun, J. Zhu, Y. Liu, J. Guo, Y. Hu, H. Chen, J. Chen, R. Wu, and L. Shao. Manual2skill: Learning to read manuals and acquire robotic skills for furniture assembly using vision-language models. In *Robotics: Science and Systems*, 2025.

[18] C. Ma, K. Lu, T.-Y. Cheng, N. Trigoni, and A. Markham. Spatialpin: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3d priors. In *Neural Information Processing Systems*, 2024.

[19] Y. Lee, E. S. Hu, and J. J. Lim. IKEA furniture assembly environment for long-horizon complex manipulation tasks. In *International Conference on Robotics and Automation*, 2021.

[20] B. Tang, M. A. Lin, I. Akinola, A. Handa, G. S. Sukhatme, F. Ramos, D. Fox, and Y. Narang. Industreal: Transferring contact-rich assembly tasks from simulation to reality. In *Robotics: Science and Systems*, 2023.

[21] G. Giacomuzzo, M. Terreran, S. Jain, and D. Romeres. DECAF: a discrete-event based collaborative human-robot framework for furniture assembly. In *International Conference on Intelligent Robots and Systems*, 2024.

[22] J. Sun, A. Curtis, Y. You, Y. Xu, M. Koehle, Q. Chen, S. Huang, L. Guibas, S. Chitta, M. Schwager, et al. Arch: Hierarchical hybrid learning for long-horizon contact-rich robotic assembly. In *Conference on Robot Learning*, 2025.

[23] C. Ma, K. Lu, R. Desai, X. Puig, A. Markham, and N. Trigoni. Coopera: Continual open-ended human-robot assistance. In *Advances in Neural Information Processing Systems*, 2025.

[24] Y. Fan, P. Ding, S. Bai, X. Tong, Y. Zhu, H. Lu, F. Dai, W. Zhao, Y. Liu, S. Huang, Z. Fan, B. Chen, and D. Wang. Long-vla: Unleashing long-horizon capability of vision language action model for robot manipulation. In *Conference on Robot Learning*, 2025.

[25] R. Yang, Z. An, L. Zhou, and Y. Feng. Seqvla: Sequential task execution for long-horizon manipulation with completion-aware vision-language-action model. *arXiv preprint arXiv:2509.14138*, 2025.

[26] Y. Yang, S. Cheng, Y. Fang, H. Bharadhwaj, M. Ding, G. Bertasius, and D. Szafir. Lilo-vla: Compositional long-horizon manipulation via linked object-centric policies. *arXiv preprint arXiv:2602.21531*, 2026.

[27] K. Lu, C. Ma, C. Hori, and D. Romeres. KitchenVLA: Iterative vision-language corrections for robotic execution of human tasks. In *International Conference on Robotics and Automation Workshop Safe-VLM*, 2025.

[28] C. Ma, G. Yang, K. Lu, S. Xu, B. Byrne, N. Trigoni, and A. Markham. Cyclevla: Proactive self-correcting vision-language-action models via subtask backtracking and minimum bayes risk decoding. *arXiv preprint arXiv:2601.02295*, 2026.

[29] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. In *Conference on Robot Learning*, 2024.

[30] Physical Intelligence. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[31] L. X. Shi, B. Ichter, M. R. Equi, L. Ke, K. Pertsch, Q. Vuong, J. Tanner, A. Walling, H. Wang, N. Fusai, A. Li-Bell, D. Driess, L. Groom, S. Levine, and C. Finn. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. In *International Conference on Machine Learning*, 2025.

[32] GR00T Team. GR00T N1: an open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

[33] Open X-Embodiment Collaboration. Open x-embodiment: Robotic learning datasets and RT-X models : Open x-embodiment collaboration. In *International Conference on Robotics and Automation*, 2024.

[34] Physical Intelligence. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

[35] P. Atreya, K. Pertsch, T. Lee, M. J. Kim, A. Jain, A. Kuramshin, C. Eppner, C. Neary, E. Hu, F. Ramos, et al. Roboarena: Distributed real-world evaluation of generalist robot policies. In *Conference on Robot Learning*, 2025.

[36] J. Bai, Y.-W. Chao, Q. Chen, J. Gu, M. J. Kim, Z. Li, X. Li, T.-Y. Lin, M.-Y. Liu, N. Ma, et al. Openpi comet: Competition solution for 2025 behavior challenge. *arXiv preprint arXiv:2512.10071*, 2025.

[37] S. Han, M. Choi, Y. Shin, G. Jang, D. Lee, J. Cho, J. Park, and J. Bae. Screwdriving gripper that mimics human two-handed assembly tasks. *Robotics*, 2022.

[38] L. Tang and Y. Jia. Robotic fastening with a manual screwdriver. In *International Conference on Robotics and Automation*, 2023.

[39] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. CALVIN: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *Robotics and Automation Letters*, 2022.

[40] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. LIBERO: benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, 2023.

[41] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems*, 2024.

[42] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State. Isaac gym: High performance GPU based physics simulation for robot learning. In *NeurIPS Datasets and Benchmarks*, 2021.

[43] 3D Warehouse. 3d warehouse. https://3dwarehouse.sketchup.com/, 2026.

[44] ambientCG. ambientcg. https://ambientcg.com/, 2026.

[45] Blender Foundation. Blender. https://www.blender.org, 2025.

[46] M. C. Welle, N. Ingelhag, M. Lippi, M. Wozniak, A. Gasparri, and D. Kragic. Quest2ros: An app to facilitate teleoperating robots. In *Workshop on VAM-HRI*, 2024.

[47] DROID Team. DROID: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems*, 2024.

Appendix for *FurnitureVLA*

A Overview

This appendix includes: 1) additional details on constructing the simulation environment, 2) details of the real-world system, and 3) more experimental details on training and results breakdown.

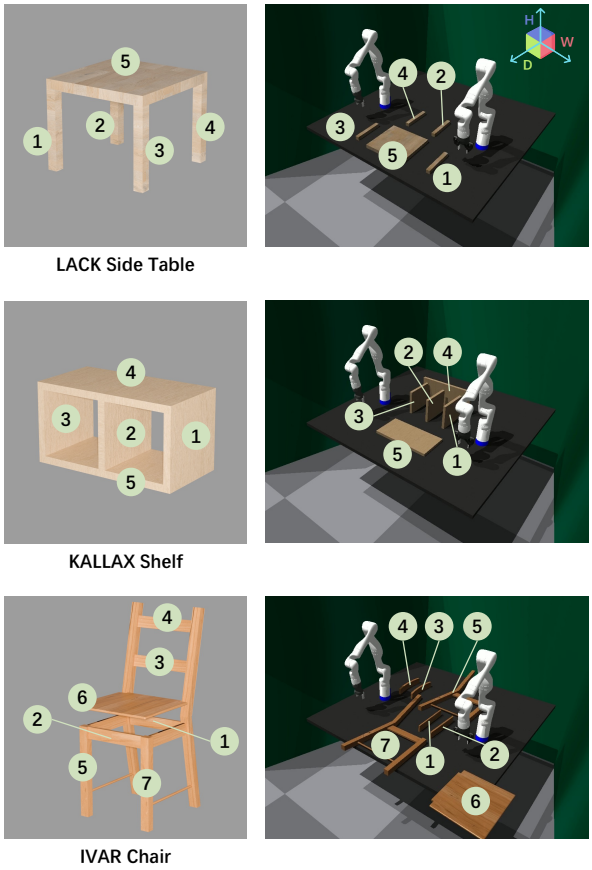


Figure 8: **Furniture part taxonomy and geometric properties.** Illustration of part naming, dimensions, and base part definition used in assembly. For the IVAR chair stage 1 definition, see Sec. B.2.

B Additional Details of Simulation

B.1 Furniture Models

The IKEA furniture used can be purchased online. 3D models and textures are also publicly available and can be used for reproducibility. IKEA serial IDs and links are below:

- **LACK side table**: [30449908] [3D model] [textures]
- **KALLAX shelf**:[90301555] [3D model] [textures]
- **IVAR chair**: [90263902] [3D model] [textures]

We illustrate the naming of each furniture part, their dimensions, and the definition of the base part in Fig. 8. Part names are manually defined based on their spatial location relative to the base part in the assembled configuration, rather than their positions on the tabletop. In simulation, all parts are assigned a density of pine wood ($450\,\mathrm{kg/m^3}$), consistent with real IKEA components.

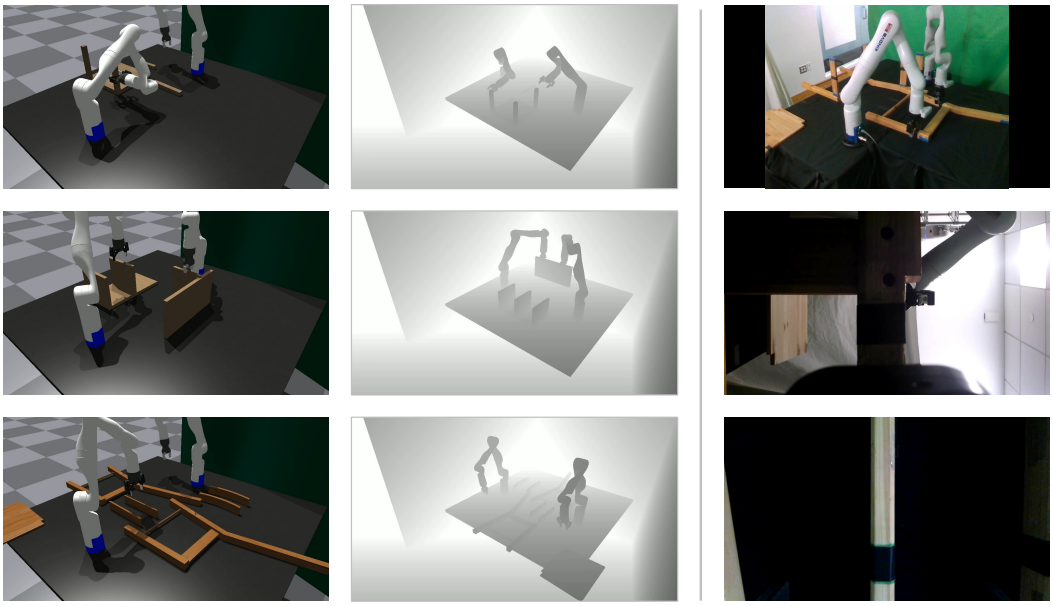


Figure 9: **Simulation and real-world observation examples.** (Left) simulation rear-view RGB and depth observations. (Right) real-world observations, including a rear-view image captured by an Intel RealSense D435 and two wrist-mounted images from the Kinova arms. All images are raw and unprocessed.

B.2 Simulation Implementation Details

Simulation of Magnets. We implement the simulator in Isaac Gym on top of FurnitureBench. Because Isaac Gym does not support runtime weld constraints, we simulate magnetic attachment by pose-resetting: at every `env.step()`, once the relative pose between a moving part and its target falls within ϵ translational and δ rotational tolerance, the moving part is kinematically teleported to maintain its rigid relative pose to the base part on every subsequent tick, with linear and angular velocities synchronized.

IVAR Chair Two-Stage Simulation. Isaac Gym does not support runtime weld constraints, so once the left side frame, rails, and right side frame are held together by the pose-reset magnets, the simulator cannot treat the resulting sub-structure as a single rigid body, preventing the final rotation required to make the partially assembled chair frame upright and attach the seat panel. We therefore split the IVAR chair episode into two stages. Stage 1 uses the right side frame as the base part and assembles the rails and left side frame against it. Stage 2 inherits the recorded terminal poses of the seat panel and partially assembled chair frame from Stage 1, then respawns the partially assembled chair frame as a single rigid mesh. In stage 2, the world frame is treated as the base reference. This allows us to evaluate whether lifting, rotating, and lowering the chair (subtask 6 in Fig. 2) is successful based on the relative pose error with respect to the world frame. Conceptually the task remains a single continuous assembly; the staging is purely an engineering workaround for the absence of weld constraints.

Control and Execution. The dual-arm Kinova Gen3 is controlled via an operational-space PD controller in the end-effector frame at $10\,\text{Hz}$.

Observations. Each environment provides RGB-D observations from four cameras: wrist-mounted cameras rigidly attached to the left and right end-effectors, along with a front and a rear camera. All observations are captured at a native resolution of 1280×720. Depth is measured using clipping planes at $0.001\,\text{m}$ and $3.5\,\text{m}$; values are clipped to $[0.001, 3.5]\,\text{m}$ and linearly normalized to the $[0, 255]$ range. Fig. 9 shows examples of rear-view RGB and depth observations in simulation.

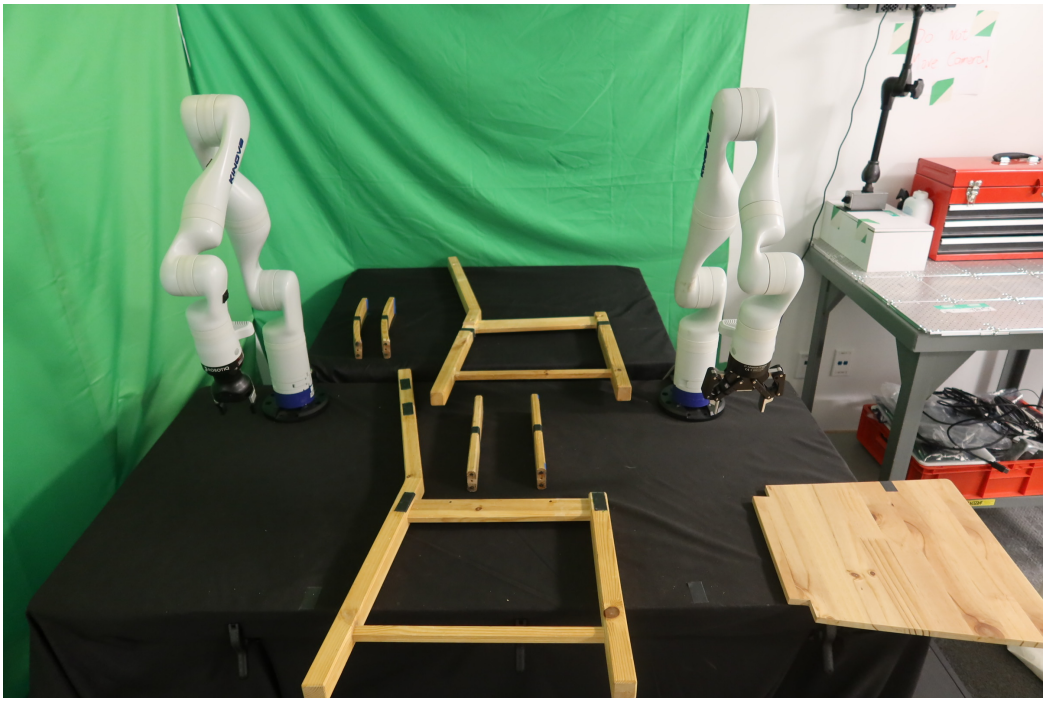


Figure 10: **IVAR chair assembly initialization.**

Data Generation Thresholds and Budgets. Scripted demonstrations are generated via per-part motion planning. The relative pose threshold for successful assembly is set to $\epsilon = 5$ mm for translation and $\delta = 0.999$ ($\approx 3.0^\circ$ per rotation axis) for rotation. Each episode is capped at a furniture-specific step budget: 900 steps for IVAR chair stage 1, 650 for IVAR chair stage 2, 650 for the LACK side table, and 850 for the KALLAX shelf. Episodes are terminated and discarded if not all parts are assembled within this budget.

VLA Policy Evaluation. As detailed in Sec. 3.2, for VLA policy evaluation we use relaxed thresholds than data generation: $\epsilon = 1$ cm for small parts and $\epsilon = 2$ cm for large parts, with $\delta = 0.998$ ($\approx 4.0^\circ$ per rotation axis). The part classification is as follows:

- **Small parts:** LACK (front right leg, rear right leg, front left leg, rear left leg); KALLAX (right side panel, center divider, left side panel); IVAR (rear seat rail, front seat rail, lower back rail, upper back rail)
- **Large parts:** KALLAX (top panel); IVAR (left side frame, partially assembled chair frame, seat panel)

For evaluation, we extend the data generation budgets by 100 steps to 1000 (IVAR chair stage 1), 750 (IVAR chair stage 2), 750 (LACK side table), and 950 (KALLAX shelf).

C Additional Details of Real System

Control and Execution. We operate both arms in end-effector space. At each 10 Hz control step, the system computes a target end-effector pose delta for each arm, converts it into a Cartesian twist, and executes it via the Kinova Cartesian velocity controller, which performs low-level joint-velocity tracking on-board. Data collection and VLA inference share this same end-effector-delta $\rightarrow$ velocity-controller interface, differing only in the source of the end-effector pose deltas.

We teleoperate the dual Kinova Gen3 arms using a Meta Quest 3 headset with Quest2ROS, which streams 6-DoF controller poses at approximately 72 Hz. For each arm, the teleoperation node

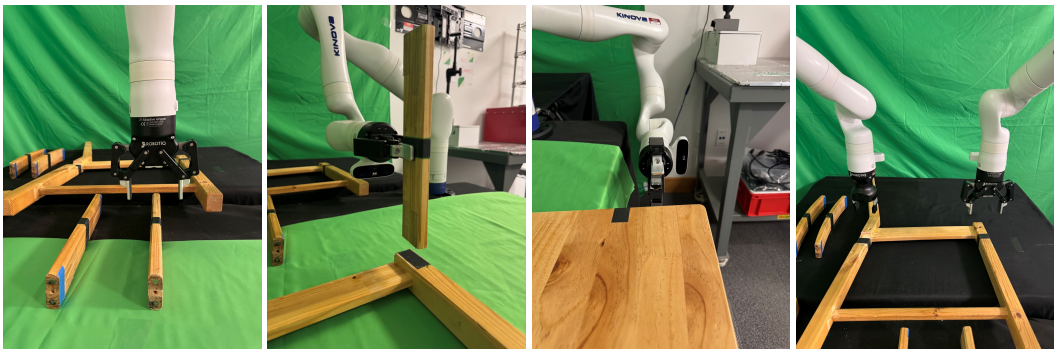


Figure 11: **Predefined orientation presets.** Left to right: grasp poses for rail grasping, rail attachment, seat panel grasping, and dual-arm coordinated grasping of the left side frame.

differentiates consecutive VR poses to obtain end-effector pose deltas, transforms them from the VR frame to the arm base frame, and forwards the resulting twists directly to the velocity controller.

At deployment, the VLA policy outputs sequences of absolute end-effector pose targets. For each arm, we compute the delta to the current pose, convert it into a Cartesian twist via a simple proportional (P) gain, and execute it through the same Kinova velocity controller.

Teleoperation Control Mapping. Since the teleoperator stands opposite the dual-arm system, control is defined in the teleoperator's frame rather than the robot's egocentric frame. Motions are mirrored such that directional commands remain intuitive (e.g., moving left/right or forward/backward results in the robot moving in the same perceived direction from the teleoperator's viewpoint). Controller button mappings are as follows (identical on both hands unless noted):

- **Index trigger** — engage translation-only control (orientation frozen); releasing freezes the arm (clutching).
- **Middle trigger** — engage rotation-only control (position frozen).
- **Both triggers held** — full 6-DoF velocity control.
- **Lower face button** (A on right, X on left) — toggle the corresponding gripper open/closed.
- **B button** (right) — emergency stop on both arms; **Y button** (left) — trigger a dual-arm coordinated bring-to-pose preset (Fig. 11).
- **Left thumbstick up** — home both arms to the simulator-matched reset pose (interruptible only by B/Y).
- **Left/right/down thumbstick** — snap the end-effector to a predefined orientation preset while holding its current position (Fig. 11).
- **Both index triggers held simultaneously** — translation sync mode: the left arm mirrors the right arm's translation.
- **Both middle triggers held simultaneously** — rotation sync mode: the left arm mirrors the right arm's rotation (combining both yields fully mirrored 6-DoF motion).

Observations. Fig. 9 shows rear and wrist camera images captured by the system.

Furniture Setup and Initialization. We use small neodymium (NdFeB) magnets. The magnets are cylindrical, with a diameter of approximately 6 mm and thickness of 2 mm, and provide sufficient holding force to stabilize parts upon contact. Magnets are attached inside screwing holes using epoxy glue. Fig. 10 shows an example initialization of the IVAR chair assembly.

Table 5: **Finetuning hyperparameters of FurnitureVLA.**

Hyperparameter	Value
pretrained checkpoint	`pi05_base`
# GPUs	8 × NVIDIA L40S (48GB VRAM)
learning rate (LR)	2.5×10^{-5} (cosine decay to 2.5×10^{-6} over 30K steps, with 1K-step linear warmup)
optimizer	AdamW (β_1=0.9, β_2=0.95, ϵ=10^{-8}, weight decay 10^{-10}, grad-norm clip 1.0)
effective batch size	64 (8 per GPU, no gradient accumulation)
# train steps	40K
# denoising steps (inference)	10 (flow-matching)
input images	4 RGB views: 1 front, 2 wrist-mounted, 1 rear
input image size	448 × 448
use observation history	no (single-step inputs)
action chunk size (H)	50 steps
action dimensions	15 (14 robot + 1 progress)
use proprio (robot state)	yes (discretized and inlined into the language prompt)
use LoRA	no (full finetune)
# trainable parameters	∼2.6B total (2.3B PaliGemma-2B backbone + 311M Gemma-300M action expert + <1M action/state projection heads)
image augmentations	resize-with-pad

Demonstration Processing. Raw demonstrations are recorded at 10 Hz. After applying a DROID-style no-op filter, the average duration per demonstration is reduced from 3 min to 2.5 min for the IVAR chair.

D Additional Experiments and Details

Training Hyperparameters. Table 5 lists the hyperparameters used to finetune FurnitureVLA on the $\pi_{0.5}$ backbone. We perform full finetuning.

Resolution Upscaling. To retain fine-grained visual cues for contact-rich bimanual assembly, we increase the input resolution of the PaliGemma–SigLIP vision backbone from its pretrained 224×224 to 448×448. The target resolution aligns with the SigLIP So400m/14 patch size, yielding a 32×32 token grid ($448/14 = 32$) without truncation. Input images are resized with aspect-ratio-preserving interpolation and zero-padded to 448×448. The pretrained positional embeddings (24×24) are resized to match the new 32×32 token grid via bicubic interpolation. All other pretrained parameters are kept unchanged, and the backbone is finetuned end-to-end with the action expert.

Simulation Results Breakdown. We visualize subtask success rates from Table 2 and Table 3 in Fig. 12.

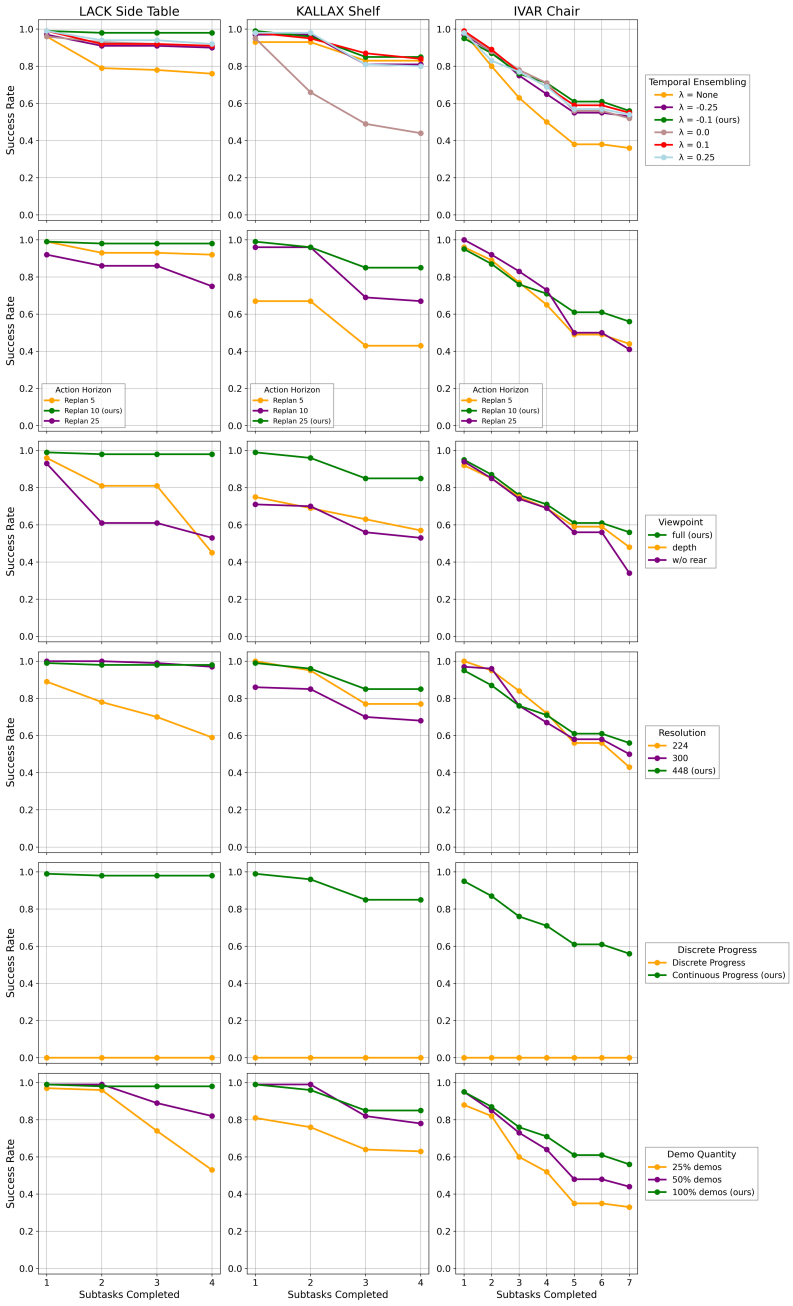


Figure 12: **Subtask success rates under different design choices.** Results correspond to Table 2 and Table 3.