# Few-shot SOP-Grounded Vision–Language–Action Policy for Proactive Human–Robot Collaborative Construction Assembly via Structured Semantic Prompts

**Jiabao Liao[1], Qiao Zheng[2], and Xinyu Tao[3]***

[1]Department of Real Estate and Construction, The University of Hong Kong, Hong Kong
[2]Zhejiang Construction Investment Group Co., Ltd., China
[3]*Department of Building and Real Estate, The Hong Kong Polytechnic University, Hong Kong

liaojiabao58@connect.hku.hk, zhengqiao@u.nus.edu, xytao@polyu.edu.hk

**Abstract -**

**Persistent labor shortages and safety exposure motivate close-contact human–robot collaboration (HRC) for construction assembly. Training proactive collaborative robots for assembly, however, commonly depends on large-scale demonstrations and extensive sensing, which increases data-collection cost and limits portability across variable workspaces. This paper develops a standard operating procedure (SOP)-grounded vision–language–action (VLA) policy. This policy uses structured semantic prompts and few-shot, parameter-efficient fine-tuning to improve the reliability of proactive assistance in physical construction assembly workflows. Two contributions are made: (i) a structured semantic SOP prompt that externalizes precedence constraints, role allocation, and proactive trigger semantics in a machine-consumable form; (ii) a few-shot, parameter-efficient fine-tuning procedure that adapts a flow-based VLA foundation policy under the structured prompt context to support intent-aware proactive assistance. On a physical 7-piece window frame assembly benchmark, few-shot tuning with 50 teleoperated demonstrations achieves 65% success (13/20) with a 102 s average completion time for successful trials, compared with 5% (1/20) for a hand-engineered visual-servoing baseline(without VLA) and 0% (0/20) for VLA baseline(without parameter-efficient fine-tuning procedure); in a prompt-free ablation(without structured semantic SOP prompt), 0/5 successes are obtained. These results support an SOP-grounded few-shot training paradigm that reduces task-specific demonstration requirements and associated training cost while improving the reliability of proactive assistance in physical assembly workflows.**

**Keywords -**

**human–robot collaboration, construction robotics, vision–language–action, few-shot adaptation, intent recognition, SOP-grounded prompting.**

## 1 Introduction

Construction projects increasingly rely on prefabrication and on-site assembly of components that demand both high throughput and tight geometric tolerance. At the same time, the construction sector faces persistent labor shortages and elevated safety exposure, motivating sustained interest in Construction 4.0 approaches that integrate robotics and AI into production and assembly workflows [1, 2]. Despite this momentum, large-scale deployment of construction automation remains constrained by fragmented processes and heterogeneous jobsite conditions that can undermine brittle, preprogrammed autonomy [3, 4]. As a result, human–robot collaboration is often viewed as a pragmatic pathway in which humans retain dexterous manipulation and safety-critical judgment while robots contribute repeatability, reach, and logistical support for repetitive subtasks [5, 6].

For collaborative assembly, however, simply colocating a robot with a worker is not enough. Effective assistance requires the robot to interpret intent-relevant cues from partial human motion, contact dynamics, and tool-use context, and then coordinate actions proactively under standard operating procedure (SOP) and safety constraints. Existing research on construction human–robot collaborative (HRC) assembly can be broadly grouped into two complementary streams. One stream improves collaboration quality through task- and interaction-level coordination designs, such as physical-state-aware handovers, to reduce idle time and support close-proximity material transfer [7]. Construction-oriented reviews further show that interaction design, safety management, and deployment robustness remain key barriers when HRC moves from controlled laboratory settings to broader on-site adoption [5]. However, even with well-designed interaction protocols, collaborative intent is often implicit and time-sensitive, so proactive assistance can still become premature or mistimed when the robot cannot reliably infer the worker's near-term goal from partial observations [8]. The other stream learns collaborative assembly be-

haviors from demonstrations, aiming to reduce manual programming by mapping observations directly to assistance actions [9]. Yet reviews of learning-based HRC show that these methods often depend on large numbers of demonstrations and/or extensive sensing and instrumentation, which increases data-collection cost and reduces portability across sites and layouts [10]. Related reviews on machine-learning-driven HRC also emphasize that sensing choices and data pipelines strongly affect reliability and generalization, and can therefore become deployment bottlenecks in heterogeneous workspaces [11]. These limitations motivate the use of policies that can leverage strong pretrained priors for intent-relevant reasoning while reducing task-specific data and instrumentation requirements, which in turn motivates the vision–language–action (VLA) foundation policy adopted in this study.

Recent progress in VLA foundation policies provides an opportunity to reuse web-scale visual and linguistic priors for robotic control, potentially reducing task-specific data requirements [12, 13]. In safety-critical assembly settings, however, specifying tasks through open-ended natural language alone can be under-constrained, and reliability concerns such as mis-grounding and hallucinated constraints have been widely discussed for large vision–language models [14]. A practical mitigation is to externalize process knowledge into structured representations (e.g., task graphs or behavior-tree-like state machines) so that action selection is conditioned on explicit prerequisites and interpretable state transitions rather than implicit assumptions [15].

Motivated by these gaps, SOPs are represented as a structured semantic state machine so that precedence constraints, role allocation, and proactive trigger semantics are made explicit for close-contact construction assembly. The resulting prompt, $P_{\text{struct}}$, is used as static global context to condition a flow-based VLA policy for continuous low-level support actions. Building on this formulation, we propose (i) a structured semantic SOP prompt that externalizes workflow constraints in a machine-consumable form and (ii) a few-shot, parameter-efficient fine-tuning procedure that adapts a flow-based VLA foundation policy under the structured prompt context to support intent-aware proactive assistance. The backbone policy is adapted via parameter-efficient fine-tuning on 50 teleoperated demonstrations and is evaluated on a physical 7-piece window frame assembly benchmark with randomized initial placements, compared against a hand-engineered visual-servoing baseline and a zero-shot VLA baseline.

## 2 Methodology

Figure 1 summarizes a closed-loop policy for proactive human–robot collaboration in component assembly, where robotic assistance is required to respect explicit SOP dependencies and close-contact constraints [5]. In the policy, workflow knowledge is externalized into a structured prompt and provided as global context to a vision–language–action policy, so that the current scene can be mapped to low-level support actions with SOP-consistent timing. Intent recognition is operationalized as inferring the worker's current SOP state and near-term need from intent-relevant visual cues and using that inference to trigger the appropriate supportive transition [8].

The proposed framework combines a structured prompt, denoted as $P_{\text{struct}}$, with a flow-based VLA policy. The pretrained backbone is denoted as $\pi_{0.5}$, and the few-shot adapted policy obtained through parameter-efficient tuning is denoted as $\pi_\theta$ [13]. At each control step, $\pi_\theta$ takes synchronized multi-view RGB observations $I_{\text{obs}}$ together with $P_{\text{struct}}$ as input and outputs continuous low-level control targets for proactive assistance. In this formulation, $P_{\text{struct}}$ provides workflow-level semantic constraints, while $I_{\text{obs}}$ provides the current scene evidence required for action generation. The concrete hardware setup, sensing configuration, and task instantiation used to obtain $I_{\text{obs}}$ are described in Section 3.

### 2.1 Structured semantic prompts

Free-form language is not used as the sole task specification. Instead, the SOP is represented as a structured semantic state machine and provided as a static global context $P_{\text{struct}}$. The structured prompt is manually authored following a fixed schema and is intended to make permissible transitions explicit, similar in spirit to structured task representations used for reliable sequencing in robotics [15].

The first part of $P_{\text{struct}}$ encodes topological dependency logic by specifying precedence constraints among subtasks. This representation captures which assembly states must be stabilized before dependent states are initiated, so that supportive actions are conditioned on prerequisite completion rather than inferred implicitly.

The second part encodes affordance-based task allocation. Dexterous, judgment-intensive operations are assigned to the human role, while repeatable logistical support is assigned to the robot role. Encoding this role split inside $P_{\text{struct}}$ avoids treating role coordination as an emergent property of the policy and instead makes it an explicit constraint on action selection.

The third part encodes spatio-temporal orchestration through proactive triggers. Rather than waiting for an explicit "task complete" signal, the prompt allows transitions

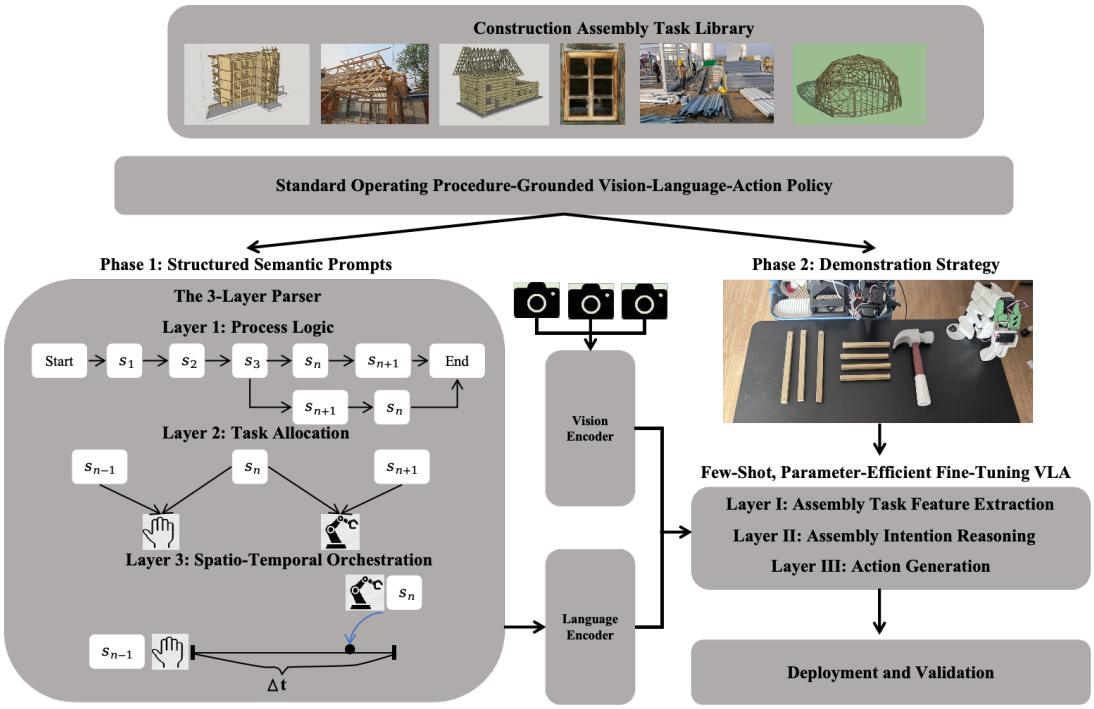


Figure 1. Overview of the SOP-grounded VLA policy. $P_{\text{struct}}$ is constructed in Section 2.1, demonstrations are collected in Section 2.2, and $\pi_\theta$ is obtained by few-shot fine-tuning from $\pi_{0.5}$ in Section 2.3 with three functional layers: feature extraction, intention reasoning, and action generation.

when intent-relevant cues indicate stabilization or imminent need. These cues are expressed at the level of prompt semantics (including nominal temporal indicators) and are not implemented as hard-coded numerical tests; instead, the tuned policy learns which visual patterns correspond to appropriate and timely transitions from the demonstration data (Section 3).

Table 1. Schema of the structured semantic state-machine prompt $P_{\text{struct}}$.

| Component | Description |
|---|---|
| Role definition | Encodes role assignments based on affordances (e.g., Human=Dexterous, Robot=Logistical). |
| Topological logic | Defines a dependency graph $G(V, E)$ where $T_i$ must precede $T_j$. |
| Proactive trigger | Encodes transition semantics linking intent cues to the next assistance action. |

### 2.2 Few-shot, parameter-efficient fine-tuning VLA

Starting from the pretrained backbone $\pi_{0.5}$ [13], parameter-efficient fine-tuning is applied on the collected demonstrations to obtain $\pi_\theta$ while keeping the prompt schema unchanged for deployment. The fine-tuned policy is organized into three layers consistent with Fig. 1:

The first part of the fine-tuned policy corresponds to assembly task feature extraction (Layer I). At each control step, $\pi_\theta$ conditions on synchronized multi-view RGB observations $I_{\text{obs}}$ together with the structured prompt $P_{\text{struct}}$. This multimodal fusion allows workflow constraints in $P_{\text{struct}}$ to modulate visual features, which is necessary because visually similar scenes can require different actions depending on the workflow state.

The second part corresponds to assembly intention reasoning (Layer II). Within this formulation, intent recognition is operationalized as inferring the worker's current SOP state and near-term need from intent-relevant visual cues, and using that inference to select a workflow-consistent supportive transition permitted by $P_{\text{struct}}$ .

The third part corresponds to action generation (Layer III). Given the fused observation–prompt context and the inferred workflow progression, the policy outputs smooth continuous low-level control targets for proactive assistance. To generate smooth continuous control targets, the decoder is expressed as a flow process:

$$\frac{d\phi_t(x)}{dt} = v_t\big(\phi_t(x); I_{\text{obs}}, P_{\text{struct}}\big), \tag{1}$$

where $t$ denotes the continuous flow time, $x$ is the variable transported by the decoder, $\phi_t(x)$ is the intermediate decoded representation at time $t$, and $v_t(\cdot)$ is a learned

time-dependent vector field [12]. The conditioning variables $I_{\text{obs}}$ and $P_{\text{struct}}$ denote the synchronized multi-view RGB observations and the structured workflow prompt, respectively. The decoded outputs are issued as continuous low-level targets that are tracked by standard robot controllers, while the specific action interface used in the physical system is provided in Section 3.

## 3 System Implementation

The methodology is instantiated in a physical setting that emulates a prefabrication workstation. The implementation follows the closed-loop structure in Fig. 1: synchronized multi-view observations are paired with a task-specific $P_{\text{struct}}$ during demonstration collection and are reused at deployment time for closed-loop inference.

### 3.1 Hardware and compute configuration

A dual-arm leader–follower teleoperation configuration is used with two LeRobot SO-101 manipulators (6-DoF). During data collection, the leader arm is operated by the human to provide reference motions while remaining passive with respect to the environment, and the follower arm tracks the leader joint trajectories; during autonomous inference, the follower arm is driven directly by the VLA policy outputs. Perception is provided by a synchronized tri-camera RGB setup comprising a global workspace view (Intel RealSense D415, RGB stream only), a fine-grained view for joinery details (DJI Pocket 3), and an ego-centric wrist-mounted view for grasping precision. Model tuning is conducted on a cloud node (AutoDL) with an NVIDIA A800 GPU (80GB VRAM), while real-time inference runs locally on an NVIDIA RTX 4070 GPU (12GB VRAM). The workstation layout and the three synchronized camera views are shown in Fig. 2.

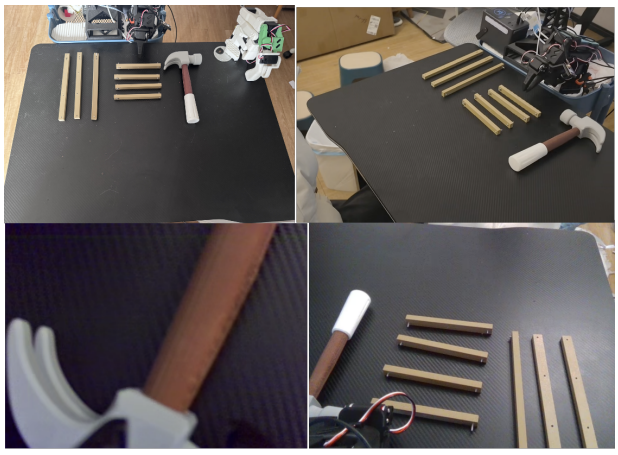

Figure 2. Desktop platform settings and the views from the three cameras.

### 3.2 Benchmark task: 7-piece window assembly

The benchmark is a 7-piece timber window frame assembly consisting of two vertical stiles (long), four horizontal rails (short), and one central mullion. The task requires coordinated positioning, stabilization, tool handover, and percussive fastening (nailing), with the human performing dexterous alignment and fastening while the robot provides logistical support and timely handovers consistent with $P_{\text{struct}}$.

### 3.3 Prompt instantiation for the benchmark

The schema in Section 2 is instantiated into a task-specific structured prompt used as static context during both training and inference. Table 2 summarizes the task definition, roles, safety semantics, and the intended trigger-to-action mapping for proactive assistance.

In the current prototype, "contact" is assessed visually as continuous hand–component contact, rather than via dedicated tactile or force sensing. The $> 0.5$ s duration is treated as prompt-level safety semantics learned from demonstrations, rather than being enforced by an explicit contact detector.

### 3.4 Demonstration collection protocol

To encode proactive behavior, 50 expert episodes are collected using the leader-follower setup. During collection, latency masking is demonstrated by initiating retrieval of the next component as soon as the worker's hand appears stabilized on the current component, encouraging trajectories that include concurrent assistance aligned with the structured prompt.

## 4 Experiments and Evaluation

The pipeline is evaluated on the physical 7-piece timber window frame assembly task described in Section 3. Task success, completion time for successful trials, and safety violations verified by post-hoc review of synchronized multi-view RGB recordings are reported, following common practice in close-contact construction human–robot collaboration (HRC) [5, 7]. The evaluation is designed to assess whether SOP-grounded prompting and few-shot adaptation yield reliable, timely assistance under moderate variation in initial scene conditions.

### 4.1 Experimental setup and baselines

All methods are tested under the same workspace configuration and task definition. For the SOP-grounded few-shot policy $\pi_\theta$, 20 independent trials are conducted. At the start of each trial, the initial placement of timber components is randomized within a 10 cm margin to probe sensitivity to initial conditions. Comparisons are made against

Table 2. The instantiated Structured Semantic State-Machine Prompt for the window assembly task.

| System configuration | Parameters |
|---|---|
| Task definition | Assemble 7-piece timber window frame (2 stiles, 4 rails, 1 mullion) |
| Agent roles | Human (Leader/Fastener); Robot (Follower/Provider) |
| Safety protocol | Release gripper only after visual confirmation of hand–component contact duration > 0.5 s |
| Proactive logic | Initiate next fetch when current installation is stabilized ($\approx$ 90% progress) |
| **Execution sequence** | **Action logic and triggers** |
| Phase 1: Initial framing | |
| Step 1: Vertical stiles | Cue: Human holds two long vertical stiles. Robot: Retrieve Rail 1 (bottom) and hand over. |
| Phase 2: Recursive rails | |
| Step 2: Rail 2 (lower-mid) | Cue: Human stabilizes Rail 1. Robot: Retrieve Rail 2 during securing. |
| Step 3: Rail 3 (upper-mid) | Cue: Human stabilizes Rail 2. Robot: Retrieve Rail 3 during securing. |
| Step 4: Rail 4 (top) | Cue: Human stabilizes Rail 3. Robot: Retrieve Rail 4 during securing. |
| Phase 3: Front fastening | |
| Step 5: Tool handover | Cue: Rail 4 placed; human hand empty. Robot: Retrieve hammer and hand over. |
| Step 6: Concurrent fetch | Cue: Post-handover state confirmed. Robot: Retrieve mullion and move to standby pose while nailing proceeds. |
| Phase 4: Structure flip | |
| Step 7: Instant handover | Cue: Frame rotated 180 degrees (backside visible). Robot: Extend from standby to hand over mullion. |
| Phase 5: Final fastening | |
| Step 8: Final tooling | Cue: Mullion stabilized. Robot: Retrieve hammer for final securing. |
| Step 9: Completion | Cue: Handover complete. Robot: Return to home position. |

a conventional vision-based pipeline that uses OpenCV-style perception with a hand-engineered state machine, and a zero-shot VLA baseline that deploys the pretrained foundation policy $\pi_{0.5}$ without construction-specific tuning [13]. The same policy family is used across the zero-shot and adapted conditions; in the adapted condition, the backbone is updated via few-shot, parameter-efficient fine-tuning on the 50-episode demonstration dataset collected under the structured prompt context (Section 3).

A safety violation is counted when the robot enters the shared workspace or initiates a handover-relevant motion before the human completes the corresponding prerequisite SOP state (e.g., initiating the post-flip handover while the frame is still rotating), as judged by post-hoc review of synchronized recordings; such cases correspond to premature or mistimed state transitions under ambiguous intent-relevant cues.

To isolate the role of structured prompting, a small qualitative ablation is conducted in five additional trials where the structured prompt $P_{\text{struct}}$ is removed and replaced with open-ended natural-language instruction. None of these trials completes the full workflow (0/5). The observed behaviors include inactivity or unstable motion, consistent with the general concern that under-constrained multimodal instructions can lead to mis-grounded action selection in safety-sensitive embodied settings [14].

### 4.2 Quantitative results

Table 3 summarizes performance across 20 trials per method.

The visual servoing baseline is sensitive to the 10 cm randomization; outside a narrow set of favorable initial alignments, stable grasp establishment frequently fails and recovery is limited. The zero-shot VLA baseline does not complete the full workflow, indicating that pretrained generalist priors alone do not satisfy the construction-specific sequencing and close-contact timing required by this task [13]. Under $P_{\text{struct}}$ conditioning, few-shot, parameter-efficient fine-tuning increases the success rate to 65% (13/20), with a 102 s mean completion time among successful trials. Two trials contain safety violations. The current success rate is mainly limited by two factors. First, although the adopted VLA policy benefits from pretrained priors, its action generation in our current system is still not consistently reliable enough for robust physical collaboration under occlusion, clutter, and RGB-only perception. Second, several failures were caused by premature proactive intervention, which indicates that the present framework does not yet verify human subtask completion sufficiently before triggering assistance.

Table 3. Performance comparison across 20 trials per method.

| Method | Success Rate | Avg. Time (Success) | Safety Violations |
|---|---|---|---|
| Visual Servoing | 5% (1/20) | N/A | 0 |
| Zero-shot VLA | 0% (0/20) | N/A | 0 |
| Few-shot Tuned (Proposed) | 65% (13/20) | 102 s | 2 |

### 4.3 Qualitative workflow evidence and failure modes

Figure 3 provides visual evidence of a successful continuous trial and clarifies how the policy couples intent-

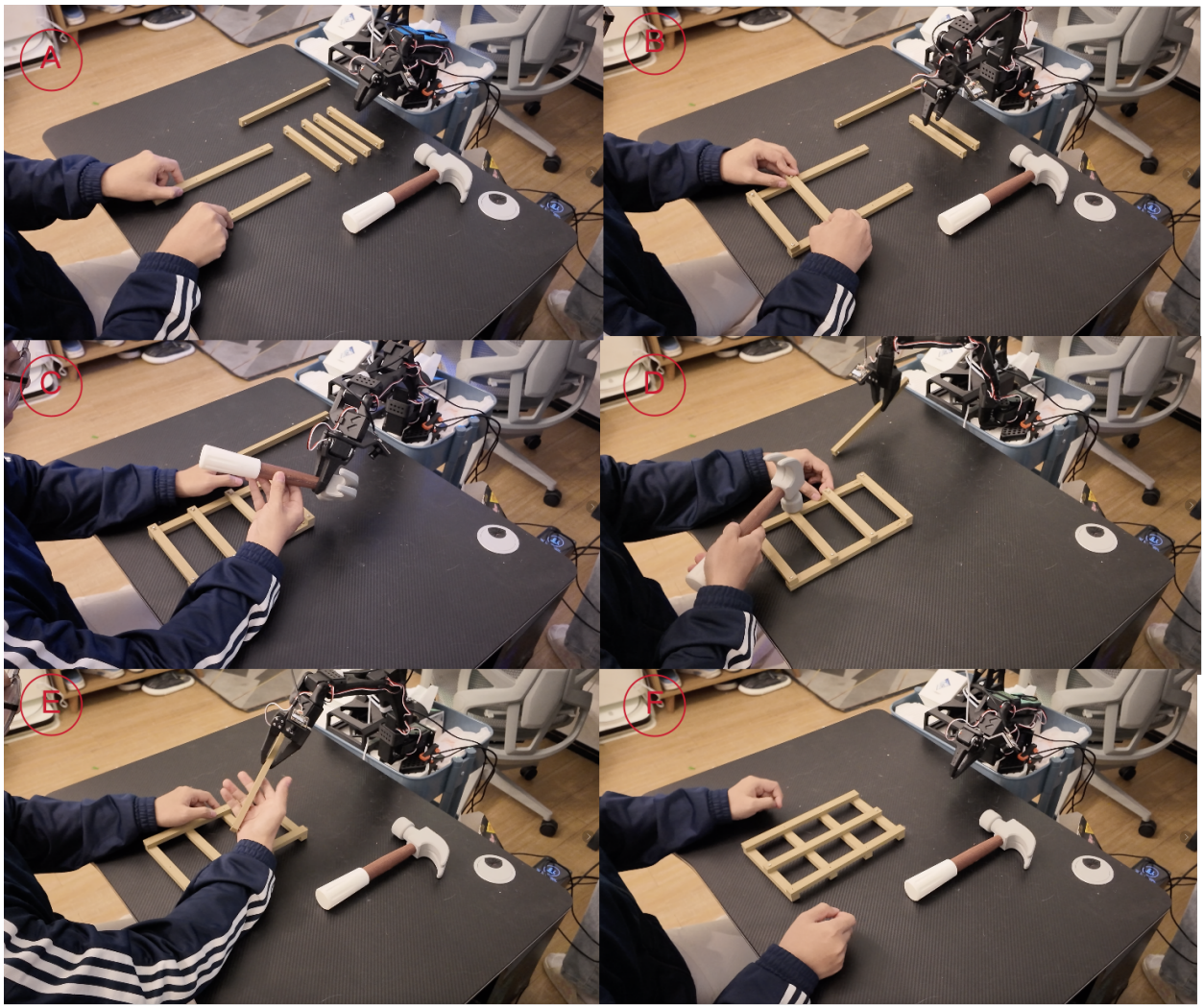


Figure 3. Keyframes of the collaborative workflow. (A) Preparation. (B) Delivery of short rails. (C) First tool handover. (D) Latency masking (robot holds mullion at standby). (E) Post-flip handover. (F) Completion.

relevant cues to SOP-consistent assistance. The sequence begins when the worker picks up two long stiles (A), which serves as the start cue for the assistance loop. The robot then delivers short rails during the worker's stabilization and placement phases (B). Once the framing reaches a fastening-ready phase, the robot switches to tool support and hands over the hammer (C). During nailing, the robot pre-fetches the mullion and holds it at a standby pose to mask latency (D). After the worker flips the frame, the robot immediately hands over the mullion (E), and the trial ends after the final securing steps are completed (F). These keyframes qualitatively illustrate SOP-consistent sequencing and intent-conditioned proactivity, complementing the quantitative metrics.

The seven failed trials fall into two recurring patterns based on post-hoc video inspection. Five failures occur during fetching when the robot does not secure a timber component, which is consistent with perceptual ambiguity under RGB-only sensing and occlusions in cluttered viewpoints. The remaining two failures are safety incidents associated with early intervention, where the handover motion is initiated before the human fully completes the frame flip. These incidents indicate that learned trigger timing can generalize prematurely; explicit safety gating is therefore required for conservative deployment in close-contact construction workflows [5, 7].

## 5 Conclusion and Future Work

An SOP-grounded, few-shot adapted VLA framework is presented for proactive human–robot collaboration in component assembly, targeting construction workflows

where explicit SOP dependencies and close-contact coordination must be respected. A structured semantic state-machine prompt is used to externalize process logic, role allocation, and proactive timing semantics, and parameter-efficient fine-tuning is applied to condition policy learning on this prompt context. On the physical 7-piece window frame assembly benchmark, proactive logistics consistent with fluency-oriented collaboration are observed, while robustness under randomized initial placements is improved relative to a hand-engineered visual-servoing baseline and a zero-shot VLA baseline, suggesting that SOP-grounded conditioning can reduce workflow and semantic gaps that often limit generalist visuomotor policies in construction-like settings. Although validated on a timber window-frame assembly task, the proposed pipeline is task-extensible because workflow knowledge is encoded through a structured semantic prompt rather than a task-specific controller. This design can be re-instantiated for other human-robot collaborative assembly tasks with different parts, role allocations, and precedence constraints.

Several limitations remain in the current prototype. First, RGB-only perception without contact-aware sensing is sensitive to lighting variation, shadows, and occlusions, which can degrade grasp reliability and propagate to mistimed proactive transitions. Second, proactive trigger timing is learned end-to-end and therefore does not provide deterministic safety assurance when intent cues are ambiguous during close-contact handovers. Future work will integrate complementary sensing modalities and lightweight contact/force feedback to improve robustness of grasping and contact-state estimation in cluttered construction workspaces. Future work will also layer explicit safety gating and state verification on top of policy outputs so that conservative, SOP-consistent transitions can be enforced even when learned triggers generalize aggressively. Together, these directions would enable more robust and conservatively deployable proactive assistance across diverse construction assembly workflows.

## Acknowledgment

This research is supported by the Hong Kong PolyU (UGC) Start-up Fund for New Recruits (P0056204) and Zhejiang Provincial "Pioneer" and "Leading Goose" Science and Technology Plan Project (2026C04040)

## References

[1] Eric Forcael, Isabella Ferrari, Alexander Opazo-Vega, et al. Construction 4.0: A literature review. *Sustainability*, 12(22):9755, 2020. doi:10.3390/su12229755.

[2] Jeroen van der Heijden, Raquel G. Santibanez Gonzalez, Sergio Vega, et al. Construction 4.0 in a narrow and broad sense: A systematic and comprehensive literature review. *Building and Environment*, 244: 110788, 2023. doi:10.1016/j.buildenv.2023.110788.

[3] Stefana Parascho. Construction robotics: From automation to collaboration. *Annual Review of Control, Robotics, and Autonomous Systems*, 6(1):183–204, 2023. doi:10.1146/annurev-control-080122-090049.

[4] Gonzalo Garcés. Future perspectives of construction robotics: An analysis of emerging trends and evolving fields. *Construction Robotics*, 2025. doi:10.1007/s44290-025-00297-7.

[5] Ming Zhang, Rui Xu, Haitao Wu, Jia Pan, and Xiaowei Luo. Human–robot collaboration for on-site construction. *Automation in Construction*, 150:104812, 2023. doi:https://doi.org/10.1016/j.autcon.2023.104812.

[6] Patrick Rodrigues et al. A multidimensional taxonomy for human–robot interaction in construction. *Automation in Construction*, 150:104845, 2023. doi:10.1016/j.autcon.2023.104845.

[7] Hongrui Yu, Vineet R. Kamat, Carol C. Menassa, Wes McGee, Yijie Guo, and Honglak Lee. Mutual physical state-aware object handover in full-contact collaborative human-robot construction work. *Automation in Construction*, 150:104829, 2023. doi:https://doi.org/10.1016/j.autcon.2023.104829.

[8] Guy Hoffman, Tapomayukh Bhattacharjee, and Stefanos Nikolaidis. Inferring human intent and predicting human action in human–robot collaboration. *Annual Review of Control, Robotics, and Autonomous Systems*, 7:73–95, 2024. doi:https://doi.org/10.1146/annurev-control-071223-105834.

[9] Harish Ravichandar, Athanasios S. Polydoros, Sonia Chernova, and Aude Billard. Recent advances in robot learning from demonstration. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:297–330, 2020. doi:https://doi.org/10.1146/annurev-control-100819-063206.

[10] Debasmita Mukherjee, Kashish Gupta, Li Hsin Chang, and Homayoun Najjaran. A survey of robot learning strategies for human-robot collaboration in industrial settings. *Robotics and Computer-Integrated Manufacturing*, 73:102231, 2022. doi:https://doi.org/10.1016/j.rcim.2021.102231.

[11] Francesco Semeraro, Alexander Griffiths, and Angelo Cangelosi. Human–robot collaboration

and machine learning: A systematic review of recent research. *Robotics and Computer-Integrated Manufacturing*, 79:102432, 2023. doi:https://doi.org/10.1016/j.rcim.2022.102432.

[12] Kevin Black, Noah Brown, Danny Driess, et al. $\pi_0$: A Vision-Language-Action Flow Model for General Robot Control. arXiv preprint, 2024. Published in RSS 2025.

[13] Physical Intelligence, Kevin Black, Noah Brown, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv*, page arXiv:2504.16054, 2025. doi:10.48550/arXiv.2504.16054.

[14] Hanchao Liu, Wenyuan Xue, Yifei Chen, et al. A survey on hallucination in large vision-language models. *arXiv*, page arXiv:2402.00253, 2024. doi:10.48550/arXiv.2402.00253.

[15] Matteo Iovino, Arturs Scukins, Johan Styrud, et al. A survey of behavior trees in robotics and ai. *Robotics and Autonomous Systems*, 154:104096, 2022. doi:10.1016/j.robot.2022.104096.