CROSSWORLD: LEARNING CONTACT-RICH ROBOTIC ULTRASOUND POLICIES FROM COMPLEMENTARY PARTIAL WORLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning robotic ultrasound policies typically relies on complete real-world expert demonstrations that jointly capture spatial probe motion, contact force, and ultrasound observations, making data collection costly and difficult to scale. Simulation offers extensive and efficient spatial exploration but lacks faithful contact-dependent dynamics that are essential for robotic ultrasound scanning. Conversely, such dynamics can be captured through local force-controlled real interactions without requiring complete expert demonstrations. Since neither domain alone provides the full experience required for policy learning, this raises a natural question: can two complementary, low-cost partial worlds be composed to reduce the need for costly complete real-world expert demonstrations? We present **CrossWorld**, a cross-domain world-modeling framework that learns from a *real contact world* and a *simulated spatial world* through *cross-world factor completion*. Specifically, *contact completion* transfers appearance dynamics learned from local real interactions to complement simulated observations, while *spatial completion* transfers action-conditioned spatial dynamics learned in simulation to complement sparse real observations. The policy then jointly perceives each factual B-mode observation and its cross-world completed view through a paired *co-observation* interface, enabling effective policy learning with a limited number of complete real-world expert demonstrations. On L4 standard-plane navigation, CrossWorld with OpenVLA achieves 77.5% cross-subject success in simulation, compared with 67.5% for simulation-only training. With local contact interactions, 50 phantom expert demonstrations, and a fixed set of 60 human demonstrations, CrossWorld achieves 75.0% success on phantom scanning, compared with 25.0% for direct mixing of simulated and expert demonstrations. These results demonstrate that composing complementary partial worlds can substantially reduce reliance on costly real-world expert demonstrations for robotic ultrasound policy learning.

1 INTRODUCTION

Robotic ultrasound aims to automate probe manipulation in clinical practice, supporting standardized imaging and ultrasound-guided interventions (Jiang et al., 2023; Li et al., 2021a; Hase et al., 2020). A robotic ultrasound policy jointly determines *where to move* and *how to maintain physical contact* while navigating toward a target plane. However, learning such contact-rich policies from expert demonstrations remains fundamentally costly: a complete real-world demonstration must synchronously capture probe motion, contact force, and ultrasound observations throughout the scanning procedure. Moreover, obtaining spatially grounded trajectories often requires dense annotations. Therefore, scalable collection of complete real-world expert demonstrations remains a major bottleneck for robotic ultrasound policy learning.

A key observation is that, in ultrasound scanning, the dynamics underlying image acquisition can be largely disentangled into two complementary aspects: *spatial dynamics*, which capture how probe motion changes its pose relative to the anatomy (Ao et al., 2025), and *contact dynamics*, which govern acoustic coupling and induce tissue deformation (Goksel & Salcudean, 2009; Treece et al., 2002; Jiang et al., 2021; 2023). Importantly, these two aspects can be predominantly learned from

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

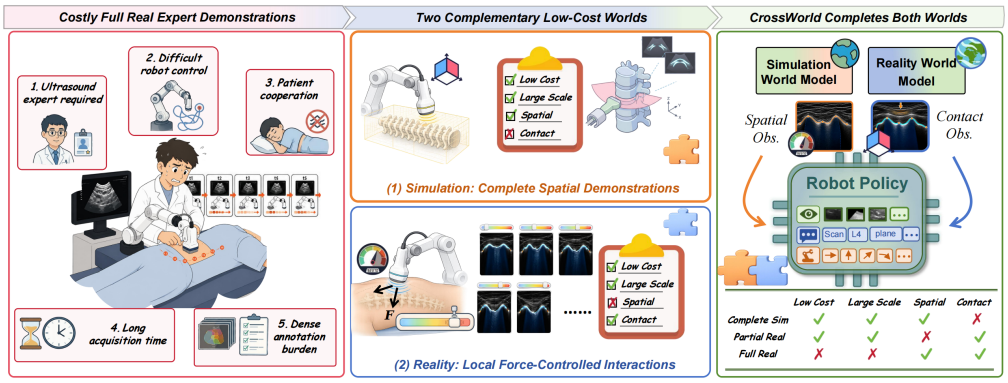


Figure 1: **Motivation and overview of cross-world co-observation.** Full real expert demonstrations for robotic ultrasound are costly to collect and annotate. CrossWorld instead combines two complementary low-cost worlds: complete simulation demonstrations with clear spatial structure, and local force-controlled interactions with measurable contact cues. Cross-world factor completion supplies the complementary factor in each world, so the policy perceives each frame by co-observation rather than reading an entangled B-mode image alone.

different low-cost sources (Figure 1). Spatial transitions and long-horizon anatomical navigation can be efficiently acquired through large-scale simulated exploration, whereas faithful contact dynamics can be learned from local force-controlled interactions in the real world. We therefore view these two data sources as *complementary partial worlds*: a spatially exploratory simulated world and a force-faithful real world. This motivates a data-efficient learning principle for robotic ultrasound: rather than relying on complete real-world demonstrations, can we compose complementary partial worlds to reduce the number of costly complete expert demonstrations required for policy learning? A straightforward solution is to combine simulated trajectories, local real interactions, and a few complete real expert demonstrations within a unified training pipeline. However, directly pooling these sources overlooks their asymmetric fidelity: simulation provides broad spatial coverage and probe-pose transitions but imperfect contact responses, whereas local real interactions provide faithful contact dynamics but lack task-directed navigation labels. Sim-to-real approaches mainly address this discrepancy through one-way transfer. Since faithfully modeling probe–tissue contact, acoustic coupling, tissue deformation, and the resulting ultrasound appearance remains difficult (Chatelain et al., 2017; Akbari et al., 2021; Tirindelli et al., 2020; Treece et al., 2002; Jiang et al., 2021), collecting substantial amounts of costly real-world data for adaptation is still required. We cast the problem as learning from two partial worlds with complementary fidelity. World models provide a natural mechanism for modeling their interaction dynamics. The central challenge is how to transfer the reliable dynamics learned in each world to the other, and compose them into a unified policy capable of both long-horizon anatomical navigation and contact-aware probe control.

We propose **CrossWorld**, a framework for learning robotic ultrasound policies from complementary partial worlds through *cross-world factor completion*. Rather than forcing either simulation or real interaction to provide complete task experience, CrossWorld learns spatial and contact dynamics from the domains where they can be reliably captured, and uses them to complete the missing factor in the other world. Specifically, CrossWorld realizes cross-world completion through a paired *co-observation* interface. For example, a simulated frame provides spatial context but limited contact-dependent appearance; the contact world model complements it with force-conditioned variations learned from real interactions. Conversely, the simulation spatial model complements real observations with action-conditioned spatial cues. We term the resulting processes *contact completion* and *spatial completion*, respectively. These completed views neither replace factual observations nor serve as independent synthetic samples. Instead, the policy jointly observes the factual and completed views during both training and deployment. This design requires neither paired sim–real observations nor dense spatial annotations in real data, and can be integrated with different policy backbones.

We instantiate CrossWorld for L4 standard-plane navigation using three data sources with distinct roles: large-scale simulated trajectories provide spatial exploration, local force-controlled real interactions provide faithful contact dynamics, and a limited number of complete real-world expert

demonstrations provide task-directed supervision for policy learning. The local interactions and simulated trajectories therefore complement, rather than replace, complete expert demonstrations. By composing these lower-cost partial-world experiences, CrossWorld aims to improve policy learning under a limited complete expert-demonstration budget and reduce reliance on costly complete real-world expert demonstrations.

- We formulate robotic ultrasound policy learning as **cross-world factor completion**, with simulation providing spatial supervision and reality providing contact supervision.
- We develop **CrossWorld**, which learns factor-specific world models, performs bidirectional cross-world completion, and trains a paired co-observation policy on factual and completed views (Sections 3.3–3.6).
- We evaluate cross-subject simulation generalization, real-robot navigation, and expert-demonstration efficiency across policy backbones (Section 4). With OpenVLA, CrossWorld achieves 77.5% cross-subject success versus 67.5% for simulation-only training.

2 Related Work

Robotic ultrasound, navigation, and contact. Robotic ultrasound couples probe motion, B-mode feedback, and contact control, with image quality and task progress strongly affected by pressure, acoustic coupling, and tissue deformation (Chatelain et al., 2017; Akbari et al., 2021; Tirindelli et al., 2020; Treece et al., 2002; Jiang et al., 2021; 2023). Learning-based methods have further enabled autonomous navigation toward standard planes using reinforcement or imitation learning (Li et al., 2021a;b; Hase et al., 2020; Bi et al., 2022). Simulation platforms such as SonoGym provide scalable trajectories and privileged pose supervision (Ao et al., 2025), but accurately reproducing force-dependent deformation and contact-induced appearance remains difficult (Goksel & Salcudean, 2009). CrossWorld therefore uses simulation for spatial supervision and real interactions for contact-dependent appearance, rather than requiring either domain to model both faithfully.

Sim-to-real policy learning. Sim-to-real methods commonly improve robustness through visual or dynamics randomization (Tobin et al., 2017; Peng et al., 2018), while modern imitation and vision-language-action policies improve action prediction from demonstrations (Zhao et al., 2023; Chi et al., 2025; Kim et al., 2024; Black et al., 2025; Shukor et al., 2025). These approaches primarily address distribution shift or representation quality. In robotic ultrasound, however, the gap is also informational: simulation exposes spatial progress, whereas inexpensive real interactions expose force-, coupling-, and deformation-dependent appearance but lack dense spatial labels. CrossWorld targets this complementary supervision directly through cross-world factor completion.

World models and generated observations. World models learn predictive dynamics for planning and policy learning (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2020), with recent robotic models coupling visual prediction and action generation (Zhu et al., 2025). Generated observations have also been used for counterfactual augmentation or imagined experience (Pitis et al., 2020; Yu et al., 2023). CrossWorld differs in both the source and use of generation: it learns factor-specific predictive dynamics in the worlds where their supervision is reliably available, transfers these dynamics bidirectionally across simulation and reality to construct completed views, and pairs each completed view with its factual observation for the same policy decision rather than treating it as an independent synthetic sample.

3 CrossWorld: Cross-World Factor Completion

3.1 Overview

Figure 2 illustrates the CrossWorld framework. CrossWorld consists of three coupled stages: factor-specific dynamics learning, bidirectional cross-world completion, and paired co-observation policy learning. We separately learn spatial transition dynamics from simulated exploration and contact-conditioned appearance dynamics from local real-world interactions. These learned dynamics are used to generate complementary views across partial worlds: real-derived contact appearance enriches simulated observations, while simulation-derived spatial dynamics complement real observations. Each generated view is paired with its factual observation to form a *co-observation*.

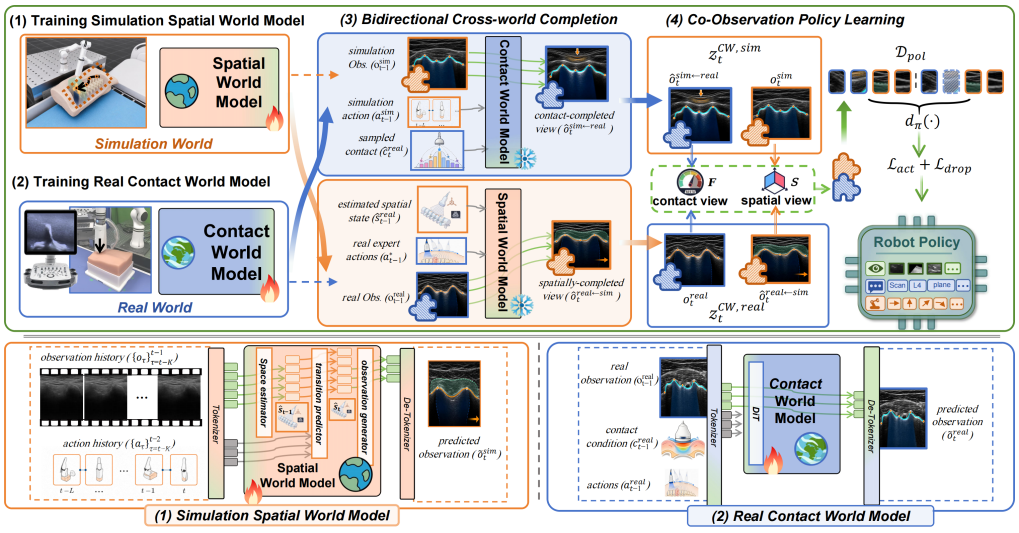


Figure 2: **CrossWorld framework.** CrossWorld learns complementary observation dynamics from two partial worlds: a *spatial world model* trained on simulated exploration with privileged spatial supervision, and a *contact-appearance world model* trained on local real-world interactions with measured contact information. Through bidirectional cross-world completion, the contact-appearance model enriches simulated observations with real-derived contact variations, while the spatial world model provides simulation-derived spatial cues for real observations. CrossWorld pairs each complementary view with its corresponding factual observation to form a *co-observation*. The policy jointly processes these paired views, with action supervision from simulated demonstrations and a limited number of complete real-world expert demonstrations.

3.2 Formulation

At time t, the robot observes a B-mode image o_t and a robot-state input $r_t \in \mathbb{R}^9$, and selects a four-dimensional action a_t. Detailed action and robot-state definitions are provided in Appendix A.2. We consider three datasets of temporally ordered trajectories represented by their per-step records:

$$\begin{aligned} \mathcal{D}_{\text{sim}} &= \{(o_t^{\text{sim}}, a_t^{\text{sim}}, s_t^{\text{sim}}, r_t^{\text{sim}})\} \\ \mathcal{D}_{\text{contact}} &= \{(o_t^{\text{real}}, a_t^{\text{real}}, c_t^{\text{real}})\} \\ \mathcal{D}_{\text{expert}} &= \{(o_t^{\text{real}}, a_t^{\text{real},*}, r_t^{\text{real}})\} \end{aligned} \tag{1}$$

Here, s_t denotes the spatial state, describing the probe pose and its spatial relation to relevant anatomy, while $c_t = \rho(w_t)$ summarizes the probe–tissue contact state from the measured force and torque w_t, as defined in Appendix A.2. Contact-induced changes in acoustic coupling and tissue deformation are captured through their effects on B-mode images. The superscript $*$ denotes the target action for policy learning, we use a_t^* to denote the action target associated with a demonstration, with $a_t^* = a_t^{\text{sim}}$ for simulation demonstrations and $a_t^* = a_t^{\text{real},*}$ for real expert demonstrations.

These datasets provide complementary but incomplete information. We learn spatial dynamics from $\mathcal{D}_{\text{sim}}$ and contact-conditioned appearance dynamics from $\mathcal{D}_{\text{contact}}$, while simulation and expert demonstrations provide action supervision. CrossWorld pairs each factual observation o_t with a cross-world complementary view $\hat{o}_t$ to form a *co-observation*:

$$z_t^{\text{CW}} = (o_t, \hat{o}_t, r_t), \tag{2}$$

where $\hat{o}_t$ provides real-derived contact appearance for simulation or simulation-derived spatial cues for real observations. The policy jointly processes these paired views:

$$\pi_\theta(a_t \mid z_t^{\text{CW}}). \tag{3}$$

Each co-observation retains the action target a_t^* of its factual trajectory.

3.3 SIMULATED SPATIAL WORLD MODEL

Spatial dynamics learning. We learn a spatial world model $M_s^{\text{sim}} = \{E_s, T_s, G_s\}$ from simulation, consisting of a spatial estimator, a transition predictor, and an observation generator. For either domain $d \in \{\text{sim}, \text{real}\}$, we define the K-frame history as

$$\mathcal{H}_{t-1}^{d} = \left(\{o_\tau^d\}_{\tau=t-K}^{t-1}, \{a_\tau^d\}_{\tau=t-K}^{t-2}\right). \tag{4}$$

Given the simulated history, the estimator infers the spatial state at $t-1$, the transition predictor advances it using the last executed action, and the generator predicts the next B-mode observation:

$$\begin{aligned} \hat{s}_{t-1}^{\text{sim}} &= E_s(\mathcal{H}_{t-1}^{\text{sim}}), \\ \hat{s}_{t}^{\text{sim}} &= T_s(\hat{s}_{t-1}^{\text{sim}}, a_{t-1}^{\text{sim}}), \\ \tilde{o}_{t}^{\text{sim}} &= G_s(o_{t-1}^{\text{sim}}, \hat{s}_{t}^{\text{sim}}). \end{aligned} \tag{5}$$

The prediction uses only past observations and executed actions, without accessing the target frame o_t or the current policy action a_t.

Simulation objective. We train the spatial model using privileged spatial-state supervision and B-mode observation prediction:

$$\mathcal{L}_{\text{sim}} = \lambda_s \|\hat{s}_t^{\text{sim}} - s_t^{\text{sim}}\|_1 + \lambda_o \mathcal{L}_{\text{gen}}^{\text{sim}}. \tag{6}$$

Here, $\mathcal{L}_{\text{gen}}^{\text{sim}}$ is the conditional denoising objective of the diffusion transformer, with o_t^{sim} as the target frame.

3.4 REAL CONTACT WORLD MODEL

Contact-appearance dynamics learning. We learn a contact world model $M_c^{\text{real}} = \{T_c, G_c\}$ from $\mathcal{D}_{\text{contact}}$, consisting of a contact transition predictor and an observation generator. Given the previous contact state, B-mode observation, and executed action, the model predicts the next contact state and its corresponding B-mode appearance:

$$\begin{aligned} \hat{c}_t^{\text{real}} &= T_c(c_{t-1}^{\text{real}}, a_{t-1}^{\text{real}}, o_{t-1}^{\text{real}}), \\ \tilde{o}_t^{\text{real}} &= G_c(o_{t-1}^{\text{real}}, a_{t-1}^{\text{real}}, \hat{c}_t^{\text{real}}). \end{aligned} \tag{7}$$

The prediction uses only past observations, measured contact states, and executed actions, without accessing the target frame o_t or the current policy action a_t.

Real Interaction objective. We train the contact world model using measured contact-state supervision and B-mode observation prediction:

$$\mathcal{L}_{\text{real}} = \lambda_c \|\hat{c}_t^{\text{real}} - c_t^{\text{real}}\|_1 + \lambda_o \mathcal{L}_{\text{gen}}^{\text{real}}. \tag{8}$$

Here, $\mathcal{L}_{\text{gen}}^{\text{real}}$ is the conditional denoising objective of the diffusion transformer, with o_t^{real} as the target frame. No spatial-state annotations are required.

3.5 BIDIRECTIONAL CROSS-WORLD COMPLETION

CrossWorld performs bidirectional cross-world completion: real-derived contact-conditioned appearance complements simulated observations, while simulation-derived spatial dynamics complement real observations. We denote each direction as receiving world ← source world.

Contact completion for simulated observations. For a simulated transition, the real contact generator combines the simulated observation and executed action with a contact condition drawn from real interactions:

$$\hat{o}_t^{\text{sim}\leftarrow\text{real}} = G_c(o_{t-1}^{\text{sim}}, a_{t-1}^{\text{sim}}, \tilde{c}_t^{\text{real}}). \tag{9}$$

Here $\tilde{c}_t^{\text{real}}$ is sampled from the empirical contact distribution of the training split of $\mathcal{D}_{\text{contact}}$. Sampling granularity, preprocessing, and the mapping ρ are specified in Appendix A.2. We apply G_c directly with the sampled contact condition, without using T_c on simulated data.

Spatial completion for real observations. For a real transition, we construct $\mathcal{H}_{t-1}^{\text{real}}$ using Equation (4) and apply the simulation spatial model:

$$\begin{aligned} \hat{s}_{t-1}^{\text{real}} &= E_s(\mathcal{H}_{t-1}^{\text{real}}), \\ \hat{s}_t^{\text{real}} &= T_s(\hat{s}_{t-1}^{\text{real}}, a_{t-1}^{\text{real}}), \\ \hat{o}_t^{\text{real}\leftarrow\text{sim}} &= G_s(o_{t-1}^{\text{real}}, \hat{s}_t^{\text{real}}). \end{aligned} \tag{10}$$

The estimated spatial state provides action-conditioned spatial cues for generating the complementary view. In both directions, the generated view is paired with the factual observation at time t for subsequent co-observation policy learning, without requiring paired sim–real trajectories or real samples annotated with both spatial and contact states.

3.6 Co-Observation Policy Learning

Co-observation learning with limited expert demonstrations. The purpose of cross-world completion is to reduce the amount of complete expert experience required for policy learning. Instead of requiring each real expert demonstration to jointly cover rich spatial transitions and physical contact variations, CrossWorld acquires these complementary dynamics from simulated exploration and local real interactions, respectively. The resulting complementary view is paired with the factual observation of a demonstrated transition, allowing the policy to exploit cross-world experience while retaining action supervision from a limited budget of complete real-world expert demonstrations.

For each demonstrated transition, the factual observation o_t, its complementary view $\hat{o}_t$, and the robot state r_t form the co-observation z_t^{CW} defined in Equation (2). The complementary view does not introduce a new demonstration or action target; each co-observation retains the action target a_t^* of its source trajectory. We therefore construct the policy training set as

$$\mathcal{D}_{\text{pol}} = \left\{ (z_t^{\text{CW,sim}}, a_t^*) \right\} \cup \left\{ (z_t^{\text{CW,real}}, a_t^*) \right\}. \tag{11}$$

The two views therefore provide complementary evidence for the same policy decision, rather than forming independently labeled synthetic samples. The policy jointly processes the co-observation to predict the demonstrated action. Its objective is

$$\mathcal{L}_{\text{pol}} = \mathbb{E}_{(z,a^*)\sim\mathcal{D}_{\text{pol}}} \left[\mathcal{L}_{\text{act}} \left(\pi_\theta(\cdot \mid z), a^* \right) \right] + \lambda_{\text{drop}} \mathcal{L}_{\text{drop}}. \tag{12}$$

Here, $\mathcal{L}_{\text{act}}$ denotes the action-learning objective of the policy backbone. Both world models remain frozen during policy learning, and only θ is optimized.

View-dropout regularization. To reduce over-reliance on generated evidence, ACT and Diffusion Policy randomly mask the complementary view during training and enforce stop-gradient consistency between full- and masked-view predictions. OpenVLA uses no additional view-dropout term. Backbone-specific objectives and settings are detailed in Appendix A.6.

Real-world deployment. At deployment, the simulation spatial model generates $\hat{o}_t^{\text{real}\leftarrow\text{sim}}$ from the rolling real-world context and the last executed action, as in Equation (10). The policy then predicts

$$\pi_\theta \left(a_t \mid o_t^{\text{real}}, \hat{o}_t^{\text{real}\leftarrow\text{sim}}, r_t^{\text{real}} \right). \tag{13}$$

Thus, real deployment requires only the spatial world model and the policy; the contact model is used only to construct simulation-side co-observations during training. At initialization, the temporal context is padded by repeating the first observation.

4 Experiments

We ask four questions: **(Q1)** Does CrossWorld improve simulation navigation and cross-subject generalization across backbones? **(Q2)** Does it improve phantom and within-participant human navigation? **(Q3)** Can CrossWorld achieve comparable performance with fewer phantom expert demonstrations? **(Q4)** How do completed views affect navigation performance and behavior?

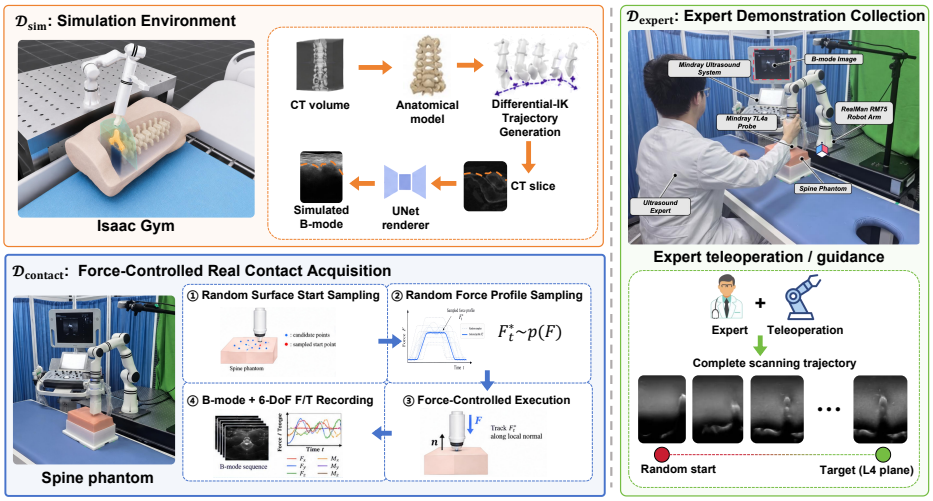


Figure 3: **Three training data sources.** $\mathcal{D}_{\text{sim}}$: complete simulated demonstrations; $\mathcal{D}_{\text{contact}}$: local force-controlled interactions; $\mathcal{D}_{\text{expert}}$: complete real expert demonstrations.

Method	Within-Subject			Cross-Subject		
	Succ. ↑	Pos. (mm) ↓	Ang. (°) ↓	Succ. ↑	Pos. (mm) ↓	Ang. (°) ↓
ACT (Zhao et al., 2023)	76.9	4.08±3.50	8.95±6.55	50.0	6.73±3.35	8.36±1.88
ACT +CrossWorld	85.0(+8.1)	3.67±2.91	**8.02±4.00**	57.5(+7.5)	5.51±4.11	**8.30±1.97**
DP (Chi et al., 2025)	76.2	3.97±3.47	9.27±8.05	52.5	6.16±3.81	8.60±1.87
DP +CrossWorld	84.4(+8.1)	3.35±3.02	8.18±3.67	60.0(+7.5)	5.23±4.12	8.50±1.62
OpenVLA (Kim et al., 2024)	78.8	3.73±3.42	8.71±5.08	67.5	4.88±3.88	8.57±1.81
OpenVLA +CrossWorld	**86.2**(+7.5)	**3.16±2.89**	8.43±4.14	**77.5**(+10.0)	**4.73±3.73**	8.42±2.24

Table 1: **Simulation navigation and cross-subject generalization.** Results are reported over 160 within-subject and 40 cross-subject rollouts. Success rates are reported in percent; parenthesized values denote percentage-point gains over the corresponding factual-view baseline, computed before rounding. Gray rows denote CrossWorld; **bold** and underlined values indicate the best and second-best results.

4.1 Experimental Setup

Task and data. We navigate an ultrasound probe from randomized starts to the standard longitudinal L4 plane using a RealMan RM75 7-DoF arm, a six-axis force/torque sensor, and a Mindray 7L4A linear-array transducer; simulation shares the arm kinematics. Figure 3 summarizes 8,192 simulated demonstrations from eight training anatomies, 512 local phantom contact trajectories, and an expert pool of 100 phantom and 60 human demonstrations. Main real-robot comparisons use 50 phantom and all 60 human demonstrations. Data protocols, collection costs, and robot calibration are detailed in Appendices A.1 and A.2.

Evaluation and comparisons. Simulation success requires position and orientation errors to remain within $5\,\text{mm}$ and 15° for 20 consecutive control steps; we report success rate and terminal pose errors. Two experienced clinicians blinded to policy identity assess real-robot success from terminal B-mode sequences. We compare ACT (Zhao et al., 2023), Diffusion Policy (Chi et al., 2025), and OpenVLA (Kim et al., 2024) under factual-view, direct-mixing, and CrossWorld training. Evaluation and implementation details are provided in Appendices A.5 and A.6.

4.2 Simulation Performance and Generalization (Q1)

Within-subject navigation. Table 1 shows that CrossWorld improves success by 7.5–8.1 percentage points across all three backbones while reducing terminal position error in every case. Open-

Method	Training Data		Succ. ↑	
	$\mathcal{D}_{\text{sim}}$	$\mathcal{D}_{\text{expert}}$	Ph.	Hum.
ACT	○	●	35.0	13.3
ACT	●	●	20.0/-15.0	0.0/-13.3
ACT +CW	●	●	65.0/+45.0	46.7/+46.7
DP	○	●	25.0	6.7
DP	●	●	25.0/0.0	0.0/-6.7
DP +CW	●	●	70.0/+45.0	53.3/+53.3
OpenVLA	○	●	10.0	0.0
OpenVLA	●	●	25.0/+15.0	6.7/+6.7
OpenVLA +CW	●	●	**75.0**/+50.0	**60.0**/+53.3

Table 2: **Real-robot success (%).** Ph./Hum.: $N = 20/15$; human results are within-participant on the same three participants. CW: CrossWorld; ●/○: used/unused. All methods use 50 phantom and 60 human expert demonstrations; +CW additionally uses $\mathcal{D}_{\text{contact}}$ only for contact-world modeling.

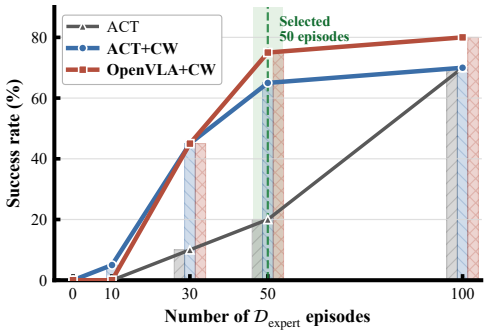


Figure 4: **Phantom data efficiency.** Success versus the phantom expert-demonstration budget, drawn from a pool of 100. All policies retain the same simulation data and 60 human demonstrations; CW also retains 512 local contact interactions. The dashed line marks the main budget of 50 phantom demonstrations.

VLA +CrossWorld reaches 86.2% success with a 3.16 mm mean position error. The consistent gains across ACT, DP, and OpenVLA suggest that cross-world completion is not tied to a particular policy formulation.

Cross-subject generalization. The gains persist on two held-out anatomies: ACT and DP improve by 7.5 percentage points, while OpenVLA rises from 67.5% to 77.5%. The strongest factual-view policy thus receives the largest cross-subject gain, suggesting that stronger backbone capability does not make completion redundant. Mean position errors also decrease, although success remains below within-subject performance for every backbone: completion improves held-out navigation without eliminating the anatomical generalization gap.

4.3 Real-Robot Navigation (Q2)

Table 2 compares expert-only training, direct mixing, and CrossWorld using the same 50 phantom and 60 human expert demonstrations. The contact world model is trained only on phantom interactions; human evaluation uses newly randomized starts on the same three participants and is therefore within-participant.

Direct mixing versus completion. Direct mixing improves only OpenVLA and reaches at most 25.0% phantom and 6.7% human success, whereas CrossWorld improves all three backbones to 65.0–75.0% and 46.7–60.0%, respectively. The gains persist on within-participant human trials despite contact-model training only on phantom interactions, supporting factor-specific completion over direct pooling. A matched control further shows that access to $\mathcal{D}_{\text{contact}}$ alone is insufficient: using the same local interactions for visual pretraining raises ACT phantom success only from 20.0% to 25.0%, versus 65.0% with CrossWorld (Appendix B.8).

4.4 Real-World Expert Data Efficiency (Q3)

Figure 4 varies the phantom expert budget over $\{0, 10, 30, 50, 100\}$ using matched subsets, while keeping $\mathcal{D}_{\text{sim}}$ and 60 human demonstrations fixed. CrossWorld additionally retains the same 512 contact trajectories, so the budget measures complete phantom expert demonstrations rather than total real-world data.

CrossWorld's advantage is concentrated at limited budgets. ACT +CrossWorld reaches 65.0% success with 50 phantom demonstrations, versus 20.0% for direct-mixing ACT, and approaches ACT's 70.0% with 100 demonstrations. The gap closes at 100 demonstrations, while OpenVLA +CrossWorld improves only from 75.0% to 80.0% when doubling the budget from 50 to 100. This satura-

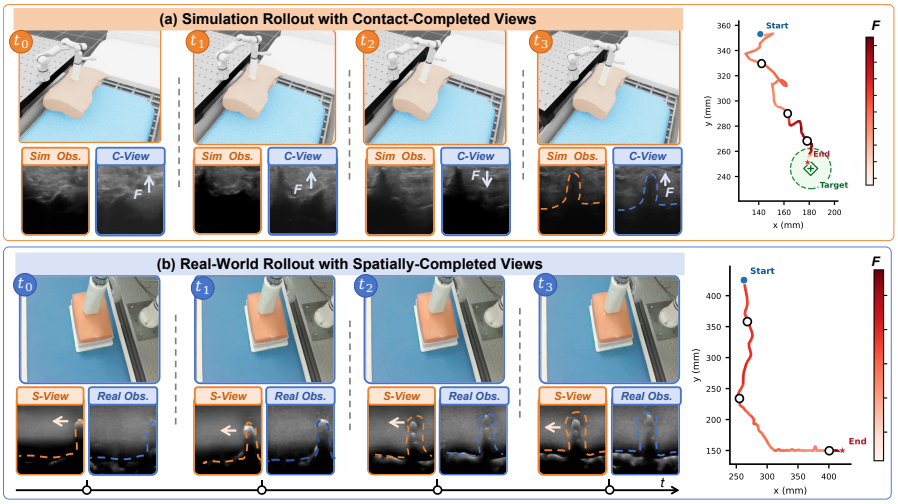


Figure 6: **Co-observation policy rollouts.** Representative simulation and real-world navigation episodes with paired factual B-mode observations and cross-world completed views, key scene frames, and force-colored probe trajectories.

tion suggests that cross-world completion is most valuable when task-directed expert supervision is scarce.

4.5 Cross-World Views: Performance and Behavior (Q4)

Navigation controls and co-observation. With OpenVLA, neither direct expert-data addition nor conventional appearance augmentation reproduces CrossWorld's gains (Appendix B.2). Independent-view training raises within-subject success from 78.8% to 81.9% but leaves cross-subject success at 67.5%, while paired co-observation reaches 86.2% and 77.5%. This shows that the benefit comes from decision-time pairing rather than treating generated views as extra training samples (Figure 5; Appendix B.1).

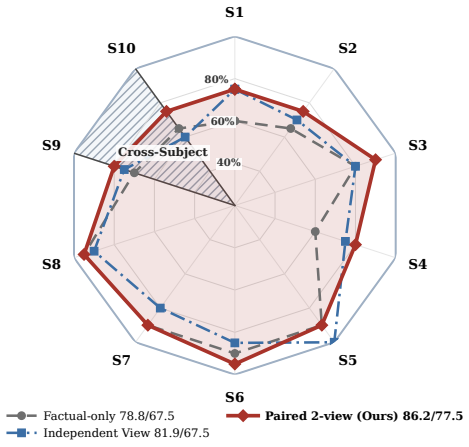


Figure 5: **Effect of co-observation.** Per-anatomy simulation success; S1–S8 are training anatomies and S9–S10 are held out.

Generated views and navigation behavior. Figure 6 illustrates the complementary roles of the two completion directions. Contact-completed simulation views vary contact-dependent appearance while retaining navigation context, whereas spatially-completed real views provide predicted anatomy-relative cues along the probe trajectory. In both cases, completion supplements rather than replaces the factual B-mode observation.

5 Conclusion

We presented CrossWorld, a cross-world completion framework for robotic ultrasound policy learning from scalable simulation and limited force-controlled real interactions. By constructing complementary views for co-observation, CrossWorld integrates simulation-derived spatial cues with real-derived contact dynamics, improving cross-subject simulation and within-participant real-robot navigation across policy backbones while reducing reliance on complete real-world expert demonstrations. Future work will evaluate broader participant populations and the limits of cross-world transfer; Appendix D provides further discussion.

AI USE STATEMENT

We used generative AI tools to assist with literature retrieval and discovery, manuscript drafting and revision, language polishing, LaTeX editing, and figure preparation. Qwen3-VL provides qualitative descriptions of terminal ultrasound images, as identified in Appendix C.1; these descriptions are not used as evaluation labels or clinical diagnoses. The authors reviewed AI-assisted text, references, and figures and take responsibility for the final manuscript.

ETHICS STATEMENT

Real-robot validation in this work involves a spine phantom and three human subjects undergoing non-diagnostic ultrasound scanning. Human-subject data collection follows institutional ethics review procedures, with informed consent from all participants. Institutional and review details are withheld for anonymous review. No identifiable personal data are released. The learned policies are research prototypes and are not intended for clinical diagnosis without regulatory validation.

REPRODUCIBILITY STATEMENT

Simulation trajectories are built on the public SonoGym environment (Ao et al., 2025). Data regeneration, model configurations, and training budgets are described in Section 4 and Appendix A. Source code, configuration files, and data-collection scripts will be released as supplementary material.

REFERENCES

Mojtaba Akbari, Jay Carriere, Tyler Meyer, Ron S. Sloboda, Siraj Husain, Nawaid Usmani, and Mahdi Tavakoli. Robotic ultrasound scanning with real-time image-based force adjustment: Quick response for enabling physical distancing during the covid-19 pandemic. *Frontiers in Robotics and AI*, 8:645424, 2021. doi: 10.3389/frobt.2021.645424.

Yunke Ao, Masoud Moghani, Mayank Mittal, Manish Prajapat, Luohong Wu, Frederic Giraud, Fabio Carrillo, Andreas Krause, and Philipp Fürnstahl. SonoGym: High performance simulation for challenging surgical tasks with robotic ultrasound, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Yuan Bi, Zhongliang Jiang, Yuan Gao, Thomas Wendler, Angelos Karlas, and Nassir Navab. VesNet-RL: Simulation-based reinforcement learning for real-world US probe navigation. *IEEE Robotics and Automation Letters*, 7(3):6638–6645, 2022. doi: 10.1109/LRA.2022.3176112.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems XXI*, 2025. doi: 10.15607/RSS.2025.XXI.010.

Pierre Chatelain, Alexandre Krupa, and Nassir Navab. Confidence-driven control of an ultrasound probe. *IEEE Transactions on Robotics*, 33(6):1410–1424, 2017. doi: 10.1109/TRO.2017.2723618.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10–11):1684–1704, 2025. doi: 10.1177/02783649241273668.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Orçun Goksel and Septimiu E. Salcudean. B-mode ultrasound image simulation in deformable 3-d medium. *IEEE Transactions on Medical Imaging*, 28(11):1657–1669, 2009. doi: 10.1109/TMI.2009.2016561.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 2555–2565, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Hannes Hase, Mohammad Farid Azampour, Maria Tirindelli, Magdalini Paschali, Walter Simson, Emad Fatemizadeh, and Nassir Navab. Ultrasound-guided robotic navigation with deep reinforcement learning. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5534–5541, 2020. doi: 10.1109/IROS45743.2020.9340913.

Zhongliang Jiang, Yue Zhou, Yuan Bi, Mingchuan Zhou, Thomas Wendler, and Nassir Navab. Deformation-aware robotic 3d ultrasound. *IEEE Robotics and Automation Letters*, 6(4):7675–7682, 2021. doi: 10.1109/LRA.2021.3099080.

Zhongliang Jiang, Septimiu E. Salcudean, and Nassir Navab. Robotic ultrasound imaging: State-of-the-art and future perspectives. *Medical Image Analysis*, 89:102878, 2023. doi: 10.1016/j.media.2023.102878.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model, 2024.

Keyu Li, Jian Wang, Yangxin Xu, Hao Qin, Dongsheng Liu, Li Liu, and Max Q.-H. Meng. Autonomous navigation of an ultrasound probe towards standard scan planes with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 8302–8308, 2021a. doi: 10.1109/ICRA48506.2021.9561295.

Keyu Li, Yangxin Xu, Jian Wang, Dong Ni, Li Liu, and Max Q.-H. Meng. Image-guided navigation of a robotic ultrasound probe for autonomous spinal sonography using a shadow-aware dual-agent framework, 2021b.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3803–3810, 2018. doi: 10.1109/ICRA.2018.8460528.

Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. In *Advances in Neural Information Processing Systems*, 2020.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. SmolVLA: A vision-language-action model for affordable and efficient robotics, 2025.

Maria Tirindelli, Maria Victorova, Javier Esteban, Seong Tae Kim, David Navarro-Alarcon, Yong Ping Zheng, and Nassir Navab. Force-ultrasound fusion: Bringing spine robotic-us to the next "level". *IEEE Robotics and Automation Letters*, 5(4):5661–5668, 2020. doi: 10.1109/LRA.2020.3009069.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 23–30, 2017. doi: 10.1109/IROS.2017.8202133.

Graham M. Treece, Richard W. Prager, Andrew H. Gee, and L. Berman. Correction of probe pressure artifacts in freehand 3d ultrasound. *Medical Image Analysis*, 6(3):199–214, 2002. doi: 10.1016/S1361-8415(02)00080-4.

Tianhe Yu, Ted Xiao, Austin Stone, Jonathan Tompson, Anthony Brohan, Su Wang, Jaspiar Singh, Clayton Tan, Jodilyn Peralta, Brian Ichter, Karol Hausman, and Fei Xia. Scaling robot learning with semantically imagined experience. In *Proceedings of Robotics: Science and Systems*, 2023.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems XIX*, 2023. doi: 10.15607/RSS.2023.XIX.016.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets, 2025.

A Experimental Details

This appendix provides detailed data protocols, robot-interface and calibration definitions, world-model architecture, training and evaluation protocols, and hyperparameter settings for the experiments in Section 4.

A.1 Data Protocols

Simulation demonstrations ($\mathcal{D}_{\text{sim}}$). We generate complete navigation demonstrations in SonoGym (Ao et al., 2025) using the RealMan RM75 arm model. CT volumes provide anatomical models, differential inverse kinematics generates probe trajectories, and a U-Net renders the corresponding CT slices into B-mode frames. The dataset contains ten patient models: S1–S8 are used for training and within-subject evaluation, while S9–S10 are held out for cross-subject evaluation. Each training anatomy contributes 1,024 trajectories from randomized initial poses to the L4 standard longitudinal plane, totaling $8 \times 1{,}024 = 8{,}192$ complete simulation demonstrations. Each demonstration records observations o_t^{sim}, actions a_t^{sim}, privileged spatial descriptors s_t^{sim}, and robot-state inputs r_t^{sim}.

Local contact interactions ($\mathcal{D}_{\text{contact}}$). We collect 512 local interaction trajectories on the spine phantom. Each trajectory starts at a sampled surface location and tracks a sampled force profile along the local surface normal. Synchronized B-mode frames, executed commands, and six-axis force/torque measurements capture contact-dependent observation changes without requiring a complete expert navigation episode or dense spatial annotation.

Real expert demonstrations ($\mathcal{D}_{\text{expert}}$). The real expert pool comprises 100 complete phantom demonstrations and 60 complete human demonstrations, with 20 collected on each of three participants. All trajectories start from randomized probe poses and terminate at the L4 target plane. We distinguish the two subsets as

$$\mathcal{D}_{\text{expert}} = \mathcal{D}_{\text{expert}}^{\text{ph}} \cup \mathcal{D}_{\text{expert}}^{\text{hum}}. \tag{14}$$

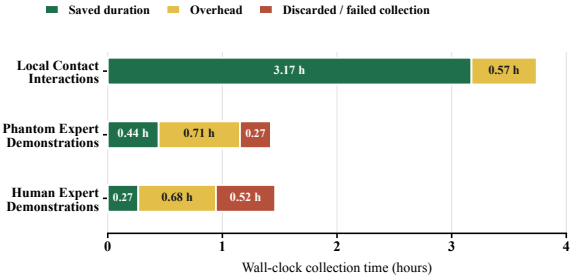
The main real-robot comparison uses 50 phantom demonstrations and all 60 human demonstrations, totaling 110 complete real expert trajectories. Expert-only, direct-mixing, and CrossWorld configurations use the same expert subset. The remaining 50 phantom demonstrations are reserved for the data-efficiency study. The 512 local contact trajectories are counted separately from expert demonstrations. Table 3 summarizes the training and evaluation data.

Phantom expert-demonstration budgets. The data-efficiency study varies the phantom budget over $n \in \{0, 10, 30, 50, 100\}$ using the same selected phantom demonstrations across methods at each budget. All 60 human expert demonstrations remain fixed, giving $n + 60$ complete expert trajectories in each configuration. The simulation dataset and, for CrossWorld, the contact dataset also remain fixed. Thus, $n = 0$ denotes the zero-phantom-expert setting.

Source	Size	Main	Q3	Role
$\mathcal{D}_{\text{sim}}$	$8 \times 1{,}024 = 8{,}192$	✓	Fixed	Spatial WM + policy
$\mathcal{D}_{\text{contact}}$	512 local phantom interactions	✓	Fixed	Contact WM
Phantom expert demos	100 total	50	0–100	Policy supervision
Human expert demos	$3 \times 20 = 60$	60	Fixed	Policy supervision
Simulation test	$8 \times 20 + 2 \times 20$	–	–	Within-/cross-subject
Phantom test	20	–	–	Real-robot evaluation
Human test	$3 \times 5 = 15$	–	–	Within-participant evaluation

Table 3: **Data summary.** Main denotes main real-robot training; WM denotes world model. Test sizes are per configuration. Q3 varies only the phantom expert budget and retains all 60 human demonstrations. The separate Q4 expert-addition control uses 50 phantom and 60 human demonstrations (Appendix B.2).

(a) Collection-time breakdown



(b) Collection statistics

	Local Contact	Phantom Expert	Human Expert
Usable attempts	512	100	60
Failed attempts	0	23	33
Total attempts	512	123	93
Saved duration (h)	3.17	0.44	0.27
Collection time (h)	3.74	1.42	1.47
Mean actions/traj.	669	480	480
Efficiency	**0.85**	0.31	0.18

Figure 7: **Collection burden of real-world data sources.** (a) Wall-clock collection time decomposed into saved duration, overhead, and discarded/failed collection. (b) Acquisition statistics for local contact interactions and complete expert demonstrations. Efficiency is the ratio of saved duration to wall-clock collection time.

Expert demonstration acquisition. Real expert demonstrations are collected by a licensed sonographer who guides the probe at the probe holder while the robot follows the applied wrench through its torque/current control loop, without a separate master arm or teach pendant. An online pose-adaptation loop uses depth-based surface-normal estimates to adjust probe alignment. Robot pose, B-mode frames, six-axis wrench measurements, and demonstration-segment annotations are recorded synchronously.

Collection burden. We additionally quantify the acquisition burden of the three real-world data sources in terms of usable and discarded attempts, retained recording duration, and wall-clock collection time. Figure 7 summarizes these statistics.

A.2 Robot Interface and Calibration

Action representation. The policy predicts a four-dimensional control target,

$$a_t = (\Delta x_t, \Delta y_t, \theta_t, f_t). \tag{15}$$

For a rollout starting at (x_0, y_0) in the calibrated workspace,

$$\Delta x_t = x_t^{\text{cmd}} - x_0, \qquad \Delta y_t = y_t^{\text{cmd}} - y_0. \tag{16}$$

The two positional components are start-relative command targets rather than increments from the previous control step. The angular component θ_t specifies an absolute orientation setpoint along the controlled rotational degree of freedom, and f_t specifies the desired axial contact-force magnitude. The positional reference remains fixed throughout a rollout, including all elements of an action chunk.

Force-target construction. For simulation policy supervision, $f_t^{\text{sim},*}$ is sampled from the empirical commanded-force distribution of $\mathcal{D}_{\text{contact}}$, since the recorded simulation force channel contains

only a zero placeholder. The resulting force targets range from 2.1 to 21.4 N. For real expert demonstrations, $f_t^{\mathrm{real},*}$ is taken directly from the commanded axial-force target of each demonstration, with a range of 2.7–20.0 N. B-mode observations, measured wrench, and action targets are recorded in a synchronized 30 Hz stream without resampling or interpolation. The force component is supervised together with the other action dimensions using the backbone's action-learning objective and is normalized using training-set statistics.

Robot-state input. The three-dimensional robot state contains the current x, y positions relative to the rollout start and the absolute measured angle along the controlled rotational degree of freedom:

$$q_t^{\mathrm{pol}} = (x_t - x_0,\, y_t - y_0,\, \theta_t^{\mathrm{meas}}). \tag{17}$$

The policy also receives a six-dimensional wrench,

$$w_t = (F_{x,t}, F_{y,t}, F_{z,t}, \tau_{x,t}, \tau_{y,t}, \tau_{z,t}) \in \mathbb{R}^6, \tag{18}$$

forming the complete input

$$r_t = (q_t^{\mathrm{pol}}, w_t) \in \mathbb{R}^9. \tag{19}$$

Real observations use synchronized measured wrench, while simulation policy training constructs the wrench input from real contact interactions as described below. The measured angle θ_t^{meas} is distinct from the commanded angle θ_t, and w_t is distinct from the commanded force target f_t. For an action-prediction horizon H, the policy interface is

$$\pi_\theta(A_t \mid o_t, \hat{o}_t, r_t), \qquad A_t = (a_t, \ldots, a_{t+H-1}). \tag{20}$$

A single-step predictor corresponds to $H = 1$. Action targets within a chunk are taken from the corresponding demonstration and retain that demonstration's coordinate reference.

Coordinate calibration. Simulation and physical scanning are expressed in a common workspace frame W, with lateral, cranio–caudal, and surface-normal axes. Let B, F, P, and I denote the robot-base, flange, probe, and metric ultrasound-image frames. The rigid probe mount defines the flange-to-probe transform ${}^{F}T_{P}$, and a standard ultrasound calibration phantom determines the probe-to-image-plane transform ${}^{P}T_{I}$. Robot kinematics supplies ${}^{B}T_{F}(q_t)$. With workspace alignment ${}^{B}T_{W}$, the image pose is

$${}^{W}T_{I}(t) = ({}^{B}T_{W})^{-1}\,{}^{B}T_{F}(q_t)\,{}^{F}T_{P}\,{}^{P}T_{I}. \tag{21}$$

Image pixel spacing and origin are specified separately so that I is expressed in metric coordinates.

Workspace alignment. In the physical phantom setup, the workspace is fixed to the scanning fixture using external geometric references, from which ${}^{B}T_{W}$ is estimated. SonoGym uses the same RM75 kinematics, metric scale, workspace-axis convention, and action bounds; anatomical models are placed using fixture or external surface references. For human scanning, standardized prone positioning and the depth-derived local surface plane establish the workspace orientation. Neither phantom nor human scans use the target L4 plane for alignment, and human scanning does not require rigid registration of internal anatomy. Randomized starts are expressed in this common control coordinate system.

Wrench calibration. The force/torque sensor is mounted between the robot flange and probe. Before each acquisition, it is zeroed and gravity-compensated using the probe-holder mass and center of mass. The corrected wrench is then transformed to the probe frame. For ${}^{P}T_{S} = (R_{PS}, p_{PS})$ mapping sensor frame S to probe frame P,

$$F^P = R_{PS}F^S, \qquad \tau^P = R_{PS}\tau^S + p_{PS} \times F^P. \tag{22}$$

Policy wrench inputs are normalized using training-set statistics.

Contact representation and sampling. The contact representation has dimension $d_c = 7$. After transforming the measured wrench to the probe frame, we augment the six-dimensional wrench with the force magnitude,

$$u_t = (F_x, F_y, F_z, \tau_x, \tau_y, \tau_z, \|F\|_2), \tag{23}$$

and apply per-channel normalization using statistics computed from the training split of $\mathcal{D}_{\text{contact}}$:

$$c_{t,k} = \frac{u_{t,k} - \mu_k}{\max(\sigma_k, 10^{-4}) + 10^{-6}}, \qquad k = 1, \ldots, 7. \tag{24}$$

For each simulated transition, we construct an action-matched contact condition from the real contact bank. Because the original simulation force channel is a zero placeholder, we sample a force value from the real commanded-force distribution and combine it with the simulated motion components to form the retrieval key,

$$\tilde{a}_{t-1}^{\text{key}} = \left(\Delta x_{t-1}^{\text{sim}}, \Delta y_{t-1}^{\text{sim}}, \theta_{t-1}^{\text{sim}}, \tilde{f}_{t-1}\right), \qquad \tilde{f}_{t-1} \sim \widehat{p}_{\text{contact}}(f). \tag{25}$$

We retrieve a real contact transition by nearest-neighbor matching in action space,

$$j = \text{NN}\left(\tilde{a}_{t-1}^{\text{key}}, \mathcal{D}_{\text{contact}}\right), \tag{26}$$

and use the same retrieved transition to provide both the simulated policy wrench and the generator contact condition:

$$\tilde{w}_t^{\text{sim}} = w_j^{\text{real}}, \qquad \tilde{c}_t^{\text{real}} = \rho(w_j^{\text{real}}). \tag{27}$$

A new real contact transition is retrieved for each training sample, independently across timesteps. The retrieval force $\tilde{f}_{t-1}$ belongs to the previous-step conditioning key and is distinct from the current policy target $f_t^{\text{sim},*}$. For factual-view baselines that contain simulated demonstrations, the simulated policy wrench is constructed using the same action-matched retrieval procedure, while contact-completed view generation is disabled.

Simulated contact convention. The simplified contact condition c_0^{sim} corresponds to the nominal fixed-contact setting used by the SonoGym trajectory generation and U-Net B-mode renderer, which does not explicitly model action-dependent force, acoustic coupling, or tissue deformation. The recorded simulation force channel is a zero placeholder and does not directly condition either trajectory generation or B-mode rendering; force variation is introduced later through real-contact sampling for policy supervision and CrossWorld contact completion.

Target-plane reference. The calibration provides the image-plane pose in the workspace frame. An independently registered geometric reference for the anatomical L4 target plane is unavailable in the physical setup, so real-world target-plane attainment is evaluated through the blinded clinical assessment described in Appendix A.5.

A.3 World-Model Architecture

CrossWorld uses a simulation spatial world model $M_s^{\text{sim}} = \{E_s, T_s, G_s\}$ and a real contact world model $M_c^{\text{real}} = \{T_c, G_c\}$. Both world models use a ViT observation encoder (Dosovitskiy et al., 2021) and a conditional diffusion transformer (DiT) generator (Peebles & Xie, 2023). In the spatial world model, the estimation branch summarizes recent observations and actions into a spatial descriptor, the transition branch propagates this descriptor using the last executed action, and the generator conditions on the predicted descriptor to produce the next B-mode observation.

Observation encoder. Each world-model encoder uses a ViT-Base architecture with 12 layers, hidden width 768, 12 attention heads, and patch size 16. The encoders are initialized from `google/vit-base-patch16-224` and further pretrained to learn compact ultrasound representations before world-model training. The spatial and contact world models use separate encoder parameters initialized from the same pretrained weights, with no parameter sharing between them or with the policy encoders. Inputs are resized to 200×150 and scaled to $[0, 1]$, without additional per-channel mean/std normalization; the frozen SDXL VAE pads them to 256×256 and applies its native latent scaling.

Spatial descriptor. The simulation spatial world model is supervised by a 30-dimensional descriptor

$$s_t \in \mathbb{R}^{30}, \tag{28}$$

Feature group	Dim.	Description
Structural coverage	6	Coverage of the target vertebra and surrounding anatomy
Global scan state	2	Global scanned fraction and current visibility
Target–plane geometry	5	Target offsets and target-to-plane/probe distances
Target–beam relation	4	Target centering and alignment with the ultrasound beam
Probe pose	3	Longitudinal spinal position and probe roll
Beam orientation	2	Beam-normal components in the calibrated world frame
Nearby anatomy	4	Types and distances of the two nearest non-target structures
Scan dynamics	4	EMA summaries of coverage, visibility, and plane displacement
Total	**30**	

Table 4: **Composition of the privileged spatial descriptor.** The descriptor provides spatial supervision only in simulation; on real observations, it is estimated from temporal context by the spatial world model.

which summarizes anatomy-relative geometry, probe orientation, scan coverage, nearby anatomy, and short-term scan dynamics. Its composition is summarized in Table 4.

Geometric quantities are expressed in the calibrated coordinate frame and normalized before training. Coverage features summarize accumulated scan history, while the dynamic features encode recent changes using exponential moving averages. The descriptor is directly available only as privileged simulation supervision; real deployment does not require access to ground-truth spatial variables.

The four nearby-anatomy dimensions encode the type and distance of the two nearest non-target structures. Structure types are represented by their index in the non-target structure set and normalized to $[0, 1]$; structures outside the predefined set are assigned to an "other vertebra" category. Distances are normalized by a fixed distance scale $d_{\max}$ and clipped to $[0, 1]$. If fewer than two non-target structures are present, missing entries are filled with a zero type–distance pair.

The scan-dynamics features use exponential moving averages,

$$\mathrm{ema}_t = (1 - \alpha)\mathrm{ema}_{t-1} + \alpha\Delta_t, \tag{29}$$

applied independently to coverage, visibility, and target-plane displacement changes, with zero initialization at the start of each episode.

Conditional generation objective. Both world models use a conditional denoising objective,

$$\mathcal{L}_{\mathrm{gen}} = \mathbb{E}_{\varepsilon,\tau}\left[\left\|\varepsilon - \hat{\varepsilon}_\psi\big(z^{(\tau)}, \tau, u\big)\right\|_2^2\right], \tag{30}$$

where $\varepsilon \sim \mathcal{N}(0, I)$, τ denotes the diffusion timestep, $z^{(\tau)}$ is the noisy observation representation, and u denotes the conditioning variables. For the spatial model, u contains the previous observation and the predicted spatial descriptor; for the contact model, it contains the previous observation, executed action, and predicted contact state. We use a cosine noise schedule and classifier-free guidance. The generation objective is combined with the corresponding spatial or contact prediction loss defined in Sections 3.3 and 3.4.

A.4 Training Protocol

Training stages. We first train the spatial world model on $\mathcal{D}_{\mathrm{sim}}$ and the contact world model on $\mathcal{D}_{\mathrm{contact}}$. The trained world models are then frozen and used to construct cross-world completed views for policy learning. Navigation-policy supervision is provided by the action targets associated with the corresponding simulation or real expert demonstrations.

Policy adaptation. ACT (Zhao et al., 2023), Diffusion Policy (Chi et al., 2025), and OpenVLA (Kim et al., 2024) retain their respective action-learning objectives. CrossWorld supplies the additional completed view and the robot-state input defined in Equation (20). Backbone-specific image fusion, robot-state encoding, initialization, action-prediction horizon, and execution horizon are documented in Table 7.

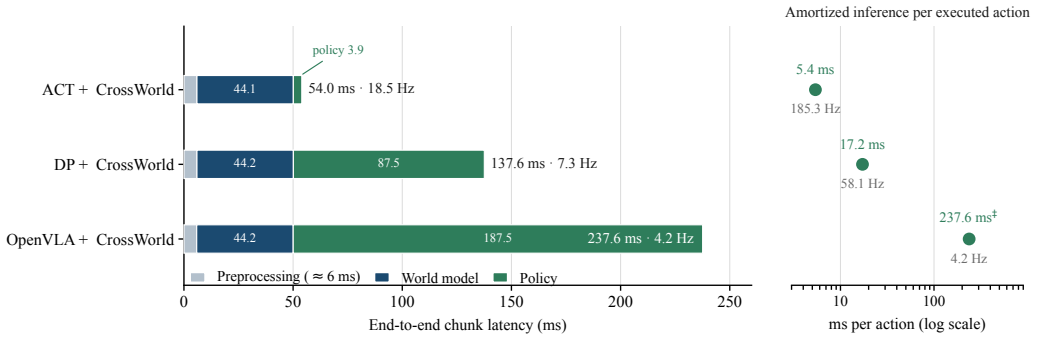


Figure 8: **Deployment inference latency.** Left: mean end-to-end latency for one action-chunk prediction, decomposed into B-mode preprocessing, spatial world-model inference, and policy inference. Right: amortized inference cost per executed action, computed using the execution horizon of each policy backbone.

Compute and optimization. Training uses eight NVIDIA A100 GPUs with PyTorch FSDP. Training the two world models takes approximately one day, and each policy configuration takes approximately one additional day. Within a policy-backbone comparison, policy updates, optimizer schedules, and B-mode augmentations are matched; the additional world-model training is accounted for separately. Trainable parameter subsets follow the backbone-specific settings in Table 7. Core optimization settings are summarized in Table 6.

Checkpoint selection and reporting. Validation uses 160 trajectories, with 20 newly sampled initial probe poses for each of the eight training anatomies S1–S8. These trajectories are generated with the same anatomy-conditioned scene generator as the training data but use newly sampled initial poses. The validation and training sets are episode-disjoint, so no episode or frame is shared between them; the two held-out anatomies S9–S10 are excluded from both training and validation.

Policy checkpoints are saved every 5,000 updates, and all reported policy results use the checkpoint at the fixed training budget of 50,000 updates. World-model training additionally tracks the checkpoint with the lowest monitored loss, with strict improvement required and earlier checkpoints retained in the event of a tie. Each model configuration is trained once using a fixed random seed (1000).

Inference and temporal context. Evaluation uses a single NVIDIA RTX 4090 per policy configuration. On the real robot, the spatial world model generates the complementary view from recent observations and executed actions. The policy receives the current factual image, completed view, and nine-dimensional robot-state input r_t, while temporal observation history is processed by the world model. At rollout initialization, missing image-history entries are padded with the initial frame, and missing action-history entries are padded with the first available executed action. Diffusion sampling settings and backbone-specific action decoding are fixed within each comparison.

Deployment runtime. Figure 8 summarizes deployment inference latency for the three CrossWorld policy backbones. The spatial world model introduces a nearly constant overhead of approximately 44 ms across configurations, while the overall runtime is mainly determined by the policy backbone. Mean end-to-end latency is 54.0 ms for ACT, 137.6 ms for Diffusion Policy, and 237.6 ms for OpenVLA. Accounting for their respective execution horizons, the amortized inference costs are 5.4, 17.2, and 237.6 ms per executed action. All measurements are obtained at batch size 1 on a single NVIDIA RTX 4090 after 20 warm-up decisions, with latency statistics computed over 60 subsequent decisions.

A.5 Evaluation Protocol

Simulation evaluation. Evaluation uses 20 randomized initial poses for each patient model, yielding 160 within-subject rollouts on S1–S8 and 40 cross-subject rollouts on the held-out anatomies S9–S10. A simulation rollout is considered successful when its position and orientation errors re-

Augmentation	Real	Simulation
Photometric	Bright. $[0.75, 1.28]$, cont. $[0.78, 1.32]$, γ $[0.88, 1.15]$; $p = 0.92$	Bright. $[0.70, 1.35]$, cont. $[0.75, 1.35]$, γ $[0.85, 1.18]$; $p = 0.95$
Grayscale	$p = 0.5$	Disabled
Gaussian blur	$\sigma \in [0, 1.6]$ px; $p = 0.50$	$\sigma \in [0, 1.8]$ px; $p = 0.55$
Speckle	$\sigma \in [0, 0.10]$; $p = 0.42$	$\sigma \in [0, 0.12]$; $p = 0.50$
Local speckle	$\sigma \in [0, 0.12]$, area 10–40%; $p = 0.20$	$\sigma \in [0, 0.15]$, area 10–40%; $p = 0.30$
Local intensity	gain $[0.75, 1.25]$, area 10–40%; $p = 0.20$	gain $[0.70, 1.30]$, area 10–40%; $p = 0.30$
Geometric	Scaling, center jitter, rotation $\leq 5^\circ$; $p = 0.92$	Scaling, center jitter, rotation $\leq 5^\circ$; $p = 0.95$

Table 5: **B-mode augmentation settings.** Photometric and texture perturbations are sampled independently. Geometric transforms are shared by the conditioning and target frames, whereas photometric and texture perturbations are applied only to the conditioning observations.

main within $5\,\mathrm{mm}$ and 15°, respectively, for 20 consecutive control steps. We report navigation success rate and the mean $\pm$ standard deviation of terminal position and angular errors across evaluation rollouts.

The maximum rollout length is 600 simulation steps. The simulator uses a fixed 30 Hz discretization, so this corresponds to 20 s of simulated time. Policy inference is synchronous and does not advance simulated time. A rollout terminates successfully once the 20-step success criterion is satisfied; otherwise, it continues until the 600-step limit, with early termination if the probe leaves the target region beyond the configured threshold. Motion-based stopping is disabled.

Real-robot evaluation. Each policy configuration is evaluated on 20 phantom rollouts and 15 human rollouts. Human evaluation comprises five randomized-start trials on each of the same three participants who contributed human expert demonstrations, and success rates are pooled across the three participants. The human experiments therefore measure within-participant navigation on additional rollouts.

For real-robot evaluation, target-plane attainment is determined from the acquired B-mode observations because an independently registered geometric reference for the anatomical L4 target plane is unavailable. Two experienced clinicians blinded to the policy configuration assess whether the terminal B-mode sequence demonstrates the standard longitudinal L4 plane. Physical rollouts terminate after 600 executed control steps or earlier upon a predefined safety stop. Policy inference is synchronous, so the wall-clock duration of a rollout depends on the inference latency of the policy configuration. Target-plane success is assessed retrospectively from the terminal B-mode sequence and does not itself trigger rollout termination.

Blinded clinical assessment. Policy identity is concealed by anonymizing and randomly shuffling rollout identifiers. The two clinicians assess each rollout independently using the predefined criteria for the standard longitudinal L4 plane. Disagreements are resolved by joint review and consensus to obtain the final binary success label. Inter-rater agreement before consensus is reported using Cohen's κ, with $\kappa = 0.82$.

Initial states and rollout matching. Evaluation initial poses are sampled independently of the training trajectories. For simulation, lateral offset, axial offset, and probe roll are sampled within the calibrated workspace bounds, and the same stored set of 20 initial poses per anatomy is replayed in the same order for every policy configuration. Between simulation rollouts, the environment is reinitialized with no state carried across episodes. For physical evaluation, the probe is returned to a neutral pose and the force/torque sensor is re-zeroed between rollouts. Policy configurations are evaluated in a fixed randomized block order, with physical rollouts interleaved in blocks of five.

A.6 Hyperparameters

B-mode augmentation settings are summarized in Table 5.

View-dropout regularization. For ACT and Diffusion Policy, we regularize the policy's reliance on the completed view through asymmetric view-dropout consistency. We sample $m \sim$

$\text{Bernoulli}(1 - p_{\text{drop}})$ and define

$$\hat{o}_t^{(m)} = m\hat{o}_t, \tag{31}$$

while leaving the factual observation and robot-state input unchanged. Using the co-observation notation from Equation (2), we define

$$z_t^+ = (o_t, \hat{o}_t, r_t), \qquad z_t^{(m)} = (o_t, \hat{o}_t^{(m)}, r_t). \tag{32}$$

For backbone b, let $\mathcal{R}_\theta^b(z; \xi)$ denote the prediction used for consistency, where ξ denotes stochastic conditioning shared by the paired evaluations. The full-view prediction serves as a stop-gradient target:

$$\mathcal{L}_{\text{drop}}^b = \mathbb{E}_{\mathcal{B}_{\text{pol}}, m, \xi}\left[d_b\left(\text{sg}[\mathcal{R}_\theta^b(z_t^+; \xi)], \mathcal{R}_\theta^b(z_t^{(m)}; \xi)\right)\right]. \tag{33}$$

Here $\text{sg}[\cdot]$ denotes stop-gradient, so the consistency term backpropagates through the masked-view prediction while the full-view prediction provides the target. View masking is disabled during deployment.

ACT discrepancy. Let $\hat{A}_t^+$ and $\hat{A}_t^m$ denote the normalized action chunks predicted from the full and masked views. A single CVAE latent is sampled from the posterior conditioned on the demonstration action chunk and reused for both evaluations, so the paired predictions share the same latent condition. For valid, non-padded action coordinates $\mathcal{I}_t$, we use mean absolute error:

$$d_{\text{ACT}} = \frac{1}{|\mathcal{I}_t|} \sum_{(h,j)\in\mathcal{I}_t} \left|\text{sg}[\hat{A}_{t,hj}^+] - \hat{A}_{t,hj}^m\right|. \tag{34}$$

Diffusion Policy discrepancy. For demonstration action chunk A_t^*, we sample a diffusion timestep τ and noise $\epsilon \sim \mathcal{N}(0, I)$, and construct

$$A_t^{(\tau)} = \sqrt{\bar{\alpha}_\tau} A_t^* + \sqrt{1 - \bar{\alpha}_\tau}\epsilon. \tag{35}$$

Both branches use the same $A_t^{(\tau)}$, timestep, and noise. Consistency is computed between their predicted noise values:

$$d_{\text{DP}} = \frac{1}{|\mathcal{I}_t|}\left\|\text{sg}\left[\epsilon_\theta(A_t^{(\tau)}, \tau, x_t^+)\right] - \epsilon_\theta(A_t^{(\tau)}, \tau, x_t^m)\right\|_{\mathcal{I}_t,2}^2. \tag{36}$$

OpenVLA training. For OpenVLA, the factual and completed views are jointly encoded as image tokens and trained directly with the backbone's action-token cross-entropy objective. No additional view-dropout consistency term is used for this backbone, i.e.,

$$\lambda_{\text{drop}}^{\text{OpenVLA}} = 0. \tag{37}$$

The consistency weights are $\lambda_{\text{drop}} = 0.1$ for ACT and Diffusion Policy and 0 for OpenVLA. For ACT and Diffusion Policy, the completed-view mask is applied to the raw completed-view tensor before view encoding by replacing the masked view with zeros. The paired evaluations reuse the same stochastic conditioning: ACT shares the same CVAE latent, while Diffusion Policy shares the same diffusion timestep and noise.

For simulation policy training, the wrench input is sampled from the empirical distribution of real force-controlled interactions, whereas real-robot deployment uses the synchronized measured wrench.

B Additional Ablation Studies

B.1 Effect of Co-Observation

We isolate how the generated cross-world view is consumed by the policy while keeping the backbone, training demonstrations, robot-state input, and action supervision unchanged.

Setting	Value
Spatial world-model context length K	10
Simulation policy state input	3-D robot state + 6-D sampled wrench
Real policy state input	3-D robot state + 6-D measured wrench
Action space	4-D
Optimizer / learning rate / weight decay	AdamW / 1×10^{-5} / 1×10^{-4}
Learning-rate schedule	Cosine decay
Batch size / policy updates	64 / 50,000
Spatial world-model loss weights	$\lambda_s = 1.0$, $\lambda_{\text{gen}} = 1.0$, $\lambda_{\text{lpips}} = 0.1$
Contact world-model loss weights	$\lambda_c = 1.0$, $\lambda_{\text{gen}} = 1.0$, $\lambda_{\text{lpips}} = 0.1$
View-dropout probability p_{drop}	0.5
Diffusion sampling steps	10
Action normalization	Per-dimension mean/std normalization

Table 6: **Main implementation settings.** We report the hyperparameters most relevant to world-model training and policy learning.

Setting	ACT	DP	OpenVLA
Initialization	ImageNet-pretrained ResNet-18; remaining policy modules initialized from scratch	Random initialization	OpenVLA-7B with LoRA ($r = 32$, $\alpha = 32$, dropout 0.1)
Trainable parameter subsets	Image encoder, Transformer, CVAE encoder/decoder, and robot-state projection	Observation encoder, diffusion U-Net, and robot-state projection	LoRA adapters and robot-state projection; base OpenVLA parameters frozen
Two-view fusion	Shared ResNet-18; feature concatenation before the Transformer encoder	Shared image encoder; feature concatenation before the U-Net	Both views are encoded as image tokens in the VLM input
Robot-state encoding (r_t)	Linear projection to the Transformer width and concatenation with observation tokens	Linear projection to the observation embedding and concatenation with image features	Linear projection to the language-model width and prepending as a state token
Prediction horizon H	100	16	1
Execution horizon	10	8	1
Action decoding	Deterministic CVAE latent decoding of action chunks	DDPM with 50 denoising steps	Token-based action decoding

Table 7: **Backbone-specific policy settings.** Prediction and execution horizons refer to policy actions and are independent of the spatial world-model context length K.

Factual-only. The policy receives the factual B-mode observation and robot-state input r_t; the generated-view slot is masked during both training and evaluation.

Independent view. Factual and generated observations are treated as separate single-view training samples, each paired with the same robot-state input and action target from the source demonstration. At deployment, the policy receives only the factual observation and r_t.

Paired co-observation. The policy jointly receives the factual observation and its generated cross-world counterpart together with the same r_t. The action target remains that of the source demonstration. This is the full CrossWorld setting.

Quantitative results and per-anatomy comparisons are reported in Figure 5 in the main paper.

B.2 Additional Controls for Cross-World Views

Expert-data addition. The data-source control adds the same 50 phantom and 60 human expert demonstrations used in the main real-robot setting to factual-view simulation training. This control

Configuration	Within-Subject			Cross-Subject		
	Succ. ↑	Pos. (mm) ↓	Ang. (°) ↓	Succ. ↑	Pos. (mm) ↓	Ang. (°) ↓
Simulation only	78.8	3.73±3.42	8.71±5.08	67.5	4.88±3.88	8.57±1.81
+ Real expert demos	78.1	3.86±3.42	**8.06±3.38**	65.0	4.79±4.07	9.18±1.66
+ Augmented view	77.5	4.65±2.99	8.33±4.84	62.5	5.53±3.73	8.61±1.46
+ CrossWorld	**86.2**	**3.16±2.89**	8.43±4.14	**77.5**	**4.73±3.73**	**8.42±2.24**

Table 8: **OpenVLA controls for cross-world completion.** Results use 160/40 within-/cross-subject rollouts. The real-expert control tests direct data addition, while the augmented-view control tests whether a generic second view reproduces the benefit of cross-world completion. Gray rows denote CrossWorld; **bold**/underlined indicate best/second-best.

Configuration	Contact	Spatial	Sim. input (S, C)	Real input (S, C)	Ph. Succ. ↑
No completion	○	○	$(o_t^{\text{sim}}, \mathbf{0})$	$(\mathbf{0}, o_t^{\text{real}})$	25.0
Contact only	●	○	$(o_t^{\text{sim}}, \hat{o}_t^{\text{sim}\leftarrow\text{real}})$	$(\mathbf{0}, o_t^{\text{real}})$	40.0
Spatial only	○	●	$(o_t^{\text{sim}}, \mathbf{0})$	$(\hat{o}_t^{\text{real}\leftarrow\text{sim}}, o_t^{\text{real}})$	30.0
Full CrossWorld	●	●	$(o_t^{\text{sim}}, \hat{o}_t^{\text{sim}\leftarrow\text{real}})$	$(\hat{o}_t^{\text{real}\leftarrow\text{sim}}, o_t^{\text{real}})$	**75.0**

Table 9: **Ablation of cross-world completion directions.** Each policy uses two semantically fixed observation slots, ordered as spatial (S) and contact (C). $\mathbf{0}$ denotes a masked missing view.

tests whether CrossWorld's gain can be reproduced by directly adding the same complete real expert data without cross-world completion.

Conventional appearance augmentation. The appearance control adds a conventionally perturbed view, $\mathcal{T}(o_t)$, through the same paired-view interface used by CrossWorld. The operator $\mathcal{T}$ composes (i) brightness/contrast jitter (gain 0.90–1.10, bias $\pm 5\%$), (ii) gamma scaling ($\gamma \in [0.8, 1.25]$), (iii) additive Gaussian noise ($\sigma = 0.02$ on $[0, 1]$), (iv) Gaussian blur ($\sigma_b = 1.0$ px), and (v) multiplicative speckle (Gamma-distributed, $\sigma_s = 0.10$). Each operator is applied independently with probability 0.5. The composition is applied only to the paired view, leaving the factual view unchanged. The backbone, robot-state interface, and action supervision are held fixed.

Table 8 summarizes these controls. Simulation-only training achieves 78.8% within-subject and 67.5% cross-subject success. Directly adding the same 50 phantom and 60 human expert demonstrations yields 78.1% and 65.0%, while the conventional augmented-view control reaches 77.5% and 62.5%. CrossWorld achieves 86.2% and 77.5%. Thus, neither direct addition of the same complete expert data nor a generic augmented second view reproduces the gain from cross-world completion.

B.3 Ablation of Cross-World Completion Directions

We isolate the two directions of cross-world completion while keeping the policy architecture, training data, robot-state input, and optimization budget fixed. Missing completed views are replaced by a masked second-view slot so that all variants use the same two-view policy interface.

Table 9 isolates the two cross-world completion directions under a fixed two-slot policy interface. Contact-only completion raises phantom success from 25.0% to 40.0%, showing that injecting real-derived contact appearance into simulated policy training improves transfer even without spatial completion on real observations. Spatial-only completion yields a smaller improvement to 30.0%, indicating that simulation-derived spatial cues are useful on real observations but are insufficient on their own.

The strongest effect emerges when the two directions are combined. Adding spatial completion on top of contact completion raises success from 40.0% to 75.0%, while adding contact completion on top of spatial completion raises it from 30.0% to 75.0%. Thus, the two completion directions provide complementary information whose joint use produces a substantially larger gain than either direction alone, supporting the bidirectional design of CrossWorld.

Contact Strategy	Axial Action	Wrench	Succ. ↑	Contact Loss ↓	Cap Act. ↓
Position-only	z_t	•	40.0	6.4	19.8
Constant-force	f_0	•	50.0	**0.3**	**0.0**
Learned force w/o wrench	f_t	○	65.0	0.9	2.1
Learned force (Ours)	f_t	•	**75.0**	1.9	3.7

Table 10: **Effect of contact-control strategy on phantom navigation.** All values are percentages over $N = 20$ phantom rollouts. Contact Loss denotes the fraction of contact-phase time with measured axial force below the 2 N valid-contact threshold, and Cap Act. denotes the fraction of contact-phase time for which the 25 N safety limiter is activated. •/○ indicates whether measured wrench is provided to the policy.

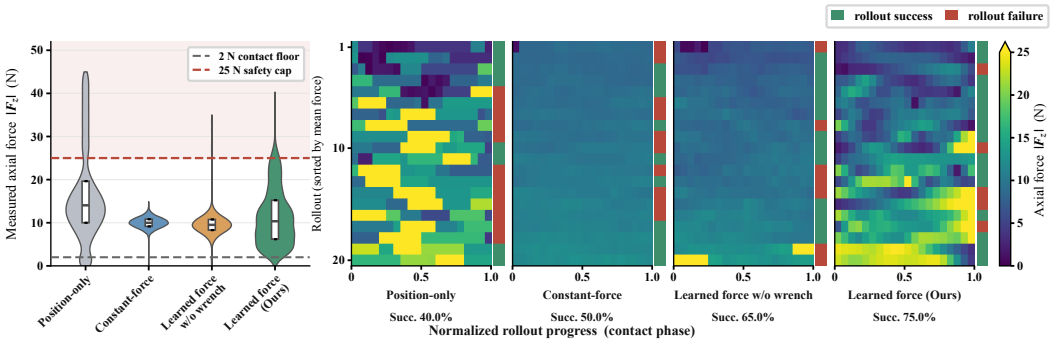


Figure 9: **Contact-force behavior under different control strategies.** Left: distributions of measured probe-normal axial force during the contact phase; dashed lines mark the 2 N contact floor and 25 N safety cap. Right: per-rollout axial-force profiles over normalized contact-phase progress, with rollouts sorted by mean force within each strategy and side bars indicating navigation success or failure.

B.4 Effect of Contact-Control Strategy

We further examine whether the learned axial-force action and measured-wrench feedback contribute to physical contact control. We compare four strategies under the same phantom navigation protocol: position-only control directly commands the axial position z_t; constant-force control uses a fixed target f_0; learned force without wrench predicts f_t while masking the measured wrench input; and the full model predicts f_t with wrench feedback. All configurations retain the same 25 N safety limiter.

Table 10 and Figure 9 show that explicit axial contact regulation is important for phantom navigation. Position-only control obtains the lowest success rate (40.0%) and exhibits substantially more contact loss and safety-limit activation, consistent with the broad measured-force distribution in Figure 9. Constant-force control stabilizes contact most strongly, reducing contact loss to 0.3% and eliminating safety-limit activation, but reaches only 50.0% navigation success. Thus, maintaining a fixed force alone is insufficient for task-adaptive scanning.

Allowing the policy to predict the force target raises success to 65.0% even without measured-wrench input, indicating that visual and robot-state cues already support useful force adaptation. Providing wrench feedback further increases success to 75.0%. The corresponding force profiles span a broader operating range than constant-force control, suggesting that the learned policy modulates contact according to the evolving navigation state rather than collapsing to a single force setpoint. Overall, these results indicate that contact stability and navigation performance are distinct objectives: fixed-force control produces the tightest force distribution, whereas feedback-informed learned force achieves the highest task success.

B.5 Effect of the Spatial World-Model Context Length

We study the temporal context used by the spatial world model by varying $K \in \{1, 4, 8, 10, 16\}$ while keeping the policy, contact model, training data, action horizon, and evaluation protocol fixed.

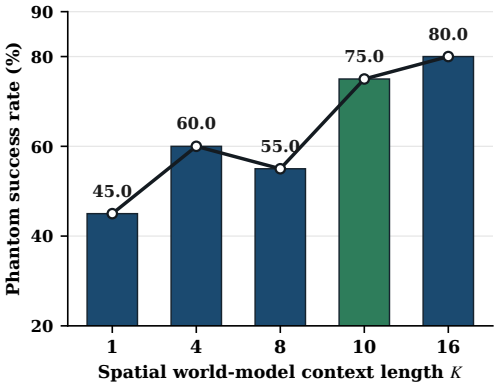


Figure 10: **Effect of spatial world-model context length.** Phantom navigation success over 20 real-robot rollouts; the main setting is $K = 10$.

The main configuration uses $K = 10$. Temporal observations are processed only by the spatial world model; the policy itself receives the current factual frame, its generated counterpart, and the current robot-state input.

Figure 10 shows that very short temporal context limits spatial completion: success is 45.0% with a single-frame context and 60.0% with $K = 4$. Performance is not strictly monotonic, dropping to 55.0% at $K = 8$, but increases substantially to 75.0% at the main setting $K = 10$ and reaches 80.0% at $K = 16$. Overall, the results indicate that spatial completion benefits from multi-frame context, consistent with the spatial descriptor containing scan-progress and temporal information that cannot be inferred reliably from a single observation.

B.6 Effect of Measured Contact Conditioning

We isolate the role of the measured contact state during contact world-model training. All policy configurations receive the same robot-state input, and the four-dimensional action $a_t = (\Delta x_t, \Delta y_t, \theta_t, f_t)$ is unchanged throughout. Thus, the commanded contact force f_t remains available even when measured-contact conditioning is removed.

In full CrossWorld, the contact transition is predicted as

$$\hat{c}_t^{\text{real}} = T_c(c_{t-1}^{\text{real}}, a_{t-1}^{\text{real}}, o_{t-1}^{\text{real}}), \tag{38}$$

whereas the ablated variant removes the previous measured contact state,

$$\hat{c}_t^{\text{real}} = T_c(a_{t-1}^{\text{real}}, o_{t-1}^{\text{real}}). \tag{39}$$

Both variants use the same observation-generator architecture,

$$\hat{o}_t^{\text{real}} = G_c(o_{t-1}^{\text{real}}, a_{t-1}^{\text{real}}, \hat{c}_t^{\text{real}}), \tag{40}$$

and c_t^{real} remains available as the supervision target during contact-model training.

For each variant, T_c and G_c are retrained jointly using the same data, initialization, optimization schedule, and training budget. Gradients from the observation-generation objective are propagated through $\hat{c}_t^{\text{real}}$ to T_c, together with the direct contact-prediction supervision. After training, each contact-model variant is used to regenerate the contact-completed simulation views under the same contact-sampling protocol. The downstream policy is then retrained from the same initialization using the same demonstration data and policy-training settings. The spatial world model is kept fixed throughout the comparison.

Table 11 shows that removing the previous measured contact state reduces phantom success from 75.0% to 45.0%. The ablated CrossWorld model still exceeds the 25.0% factual-view baseline, indicating that the previous observation and commanded action retain useful contact information, while explicit conditioning on the measured contact state provides a substantial additional benefit.

Configuration	Contact-transition input	Phantom success
Baseline	N/A	25.0%
CrossWorld w/o c_{t-1}	o_{t-1}, a_{t-1}	45.0%
CrossWorld	$o_{t-1}, a_{t-1}, c_{t-1}$	**75.0%**

Table 11: **Effect of measured contact conditioning.** Results are reported over 20 phantom rollouts. All configurations use the same policy input and training protocol; the ablation removes only c_{t-1} from the contact transition model.

Configuration	Within-Subject	Cross-Subject
CrossWorld w/o view dropout	80.6	52.5
CrossWorld + view dropout	**85.0**	**57.5**

Table 12: **Effect of view-dropout regularization.** Simulation navigation success (%) with and without view-dropout regularization using ACT.

B.7 Effect of View-Dropout Regularization

We evaluate the effect of view-dropout regularization in simulation using ACT as the policy backbone. We compare CrossWorld trained without view dropout against the full model with asymmetric stop-gradient view-dropout consistency. Both configurations use the same training data, initialization, action normalization, policy-update budget, and checkpoint-selection protocol. For the regularized model, we use $p_{\text{drop}} = 0.5$ and $\lambda_{\text{drop}} = 0.1$. Both configurations are evaluated using the standard co-observation input with the generated view available.

Table 12 shows that view-dropout regularization improves within-subject success from 80.6% to 85.0% and cross-subject success from 52.5% to 57.5%. The consistent gains in both settings support regularizing the policy's reliance on the generated view during co-observation learning.

B.8 Use of Local Contact Data

We further examine whether the improvement of CrossWorld can be explained by using $\mathcal{D}_{\text{contact}}$ for additional representation learning, rather than by cross-world completion itself. To this end, we construct a factual-view control that uses the same local contact interactions for visual pretraining, while retaining the same simulation-side force-target sampling and wrench-retrieval procedure used by the direct-mixing baseline.

Specifically, we pretrain the ACT visual encoder on local contact interactions using an image-to-wrench prediction objective. A temporary prediction head g_η maps the visual representation to the normalized measured wrench $\bar{w}_t$:

$$\mathcal{L}_{\text{pre}} = \mathbb{E}_{(o_t, w_t)\sim\mathcal{D}_{\text{contact}}} \left[\| g_\eta(E_\phi(o_t)) - \bar{w}_t \|_1 \right]. \tag{41}$$

The temporary wrench-prediction head is discarded after pretraining, and the pretrained visual encoder is used to initialize the factual-view ACT policy.

The contact-pretraining control is subsequently trained on exactly the same simulation and complete expert demonstrations as the direct-mixing baseline. All configurations containing simulated demonstrations use the same $\mathcal{D}_{\text{contact}}$-based force-target sampling and action-matched wrench retrieval to construct the simulation-side policy interface. They also share the same navigation supervision, robot-state interface, action space, policy-training budget, checkpoint-selection protocol, and real-robot evaluation procedure.

The configurations differ only in the additional use of $\mathcal{D}_{\text{contact}}$ for representation or world-model learning: direct mixing uses no additional contact-data learning, the contact-pretraining control uses $\mathcal{D}_{\text{contact}}$ for visual representation pretraining, and CrossWorld uses it to train the contact world model and construct contact-completed simulation views.

Table 13 shows that contact-data pretraining improves phantom success only modestly, from 20.0% to 25.0%, despite using the same $\mathcal{D}_{\text{contact}}$ available to CrossWorld. In contrast, CrossWorld achieves 65.0% success by using these local interactions to model contact transitions and construct comple-

Configuration	Phantom success
ACT direct mixing	20.0%
ACT + contact-data pretraining	25.0%
ACT + CrossWorld	**65.0%**

Table 13: **Alternative uses of local contact data.** All configurations use the same simulation and expert navigation demonstrations. Contact-data pretraining and CrossWorld additionally use the same $\mathcal{D}_{\text{contact}}$, respectively for visual representation pretraining and contact-world modeling. Results are reported over 20 phantom rollouts.

Method	Within-Subject			Cross-Subject		
	Succ. ↑	Pos. (mm) ↓	Ang. (°) ↓	Succ. ↑	Pos. (mm) ↓	Ang. (°) ↓
OpenVLA	79.2±2.5	3.76±0.11	8.68±0.29	68.3±3.8	4.85±0.14	8.61±0.34
OpenVLA +CrossWorld	**86.7±2.8**	**3.12±0.09**	**8.47±0.27**	**78.3±3.8**	**4.00±0.81**	**8.31±0.29**

Table 14: **Training-seed robustness.** Results are mean ± standard deviation over three independent policy-training seeds evaluated on the same initial-state sets. Pos. and Ang. denote terminal position and angular errors, respectively. Unlike Table 1, the reported standard deviations reflect variation across training seeds rather than across evaluation rollouts.

mentary views. This comparison indicates that the gain is not explained by access to additional real contact observations alone, but depends on how those observations are incorporated into policy learning.

B.9 Robustness Across Training Seeds

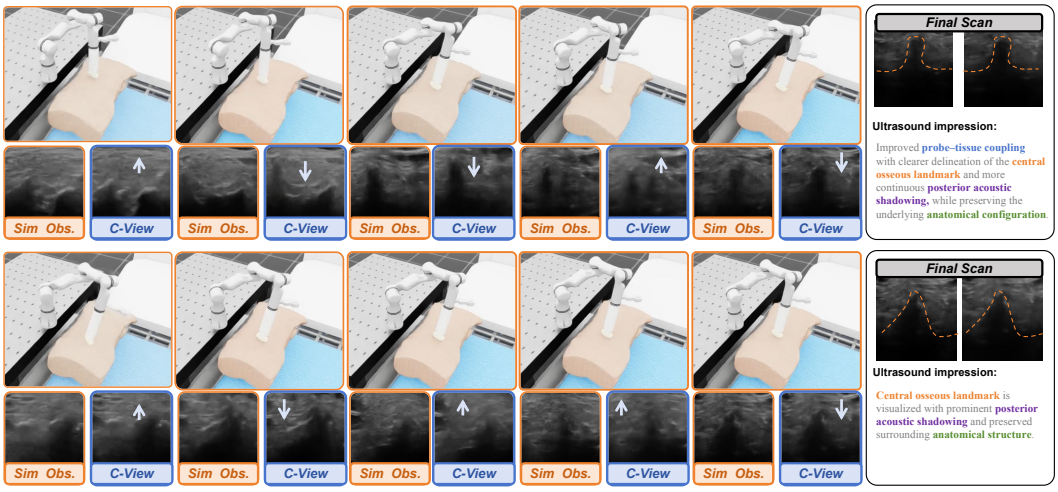
To examine sensitivity to policy-training randomness, we repeat the OpenVLA comparison over three independent training seeds while keeping the training data, evaluation initial states, optimization budget, and evaluation protocol fixed. For each seed, terminal pose errors are first averaged across evaluation rollouts; Table 14 reports the mean ± standard deviation of these seed-level results.

The gains remain consistent across training seeds. CrossWorld improves within-subject success from 79.2% to 86.7% and cross-subject success from 68.3% to 78.3%, corresponding to gains of 7.5 and 10.0 percentage points, respectively. The improvement is accompanied by lower terminal position and angular errors in both settings: cross-subject position error decreases from 4.85 mm to 4.00 mm, while angular error decreases from 8.61° to 8.31°. The comparable seed-level variability between the two methods, together with the persistent gains in success and pose accuracy, indicates that the observed advantage is not specific to a single policy-training seed.

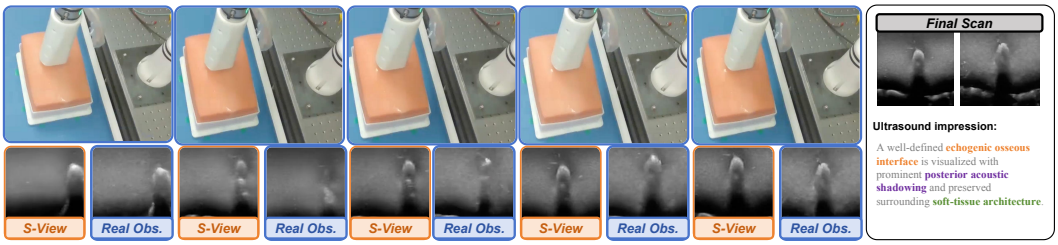
C Qualitative Results

C.1 Visualization of Policy Rollouts

Figure 11 shows representative simulation and real-robot phantom rollouts. Each timestep pairs the factual B-mode observation with its cross-world counterpart, followed by the terminal scan. Simulation uses contact-completed views, whereas real-robot deployment uses spatially-completed views. To examine whether the final scans preserve clinically interpretable ultrasound cues, we additionally provide the terminal B-mode images to Qwen3-VL (Bai et al., 2025) for a qualitative image assessment. The resulting text is displayed in the rightmost panels. Colored phrases highlight different categories of image evidence: **probe–tissue coupling**, **osseous landmarks**, **posterior acoustic shadowing**, and **preserved anatomical structure**. These VLM-generated assessments are used only as qualitative interpretation aids and are not treated as clinical diagnoses or evaluation labels. To preserve participant privacy, qualitative human rollout visualizations are omitted; human performance is reported quantitatively in Section 4.3.



(a) Simulation rollouts.



(b) Real-robot phantom rollouts.

Figure 11: **CrossWorld policy rollouts.** Representative simulation and real-robot phantom episodes with paired factual and cross-world observations, terminal scans, and VLM-based qualitative scan assessments.

VLM inference settings. The Qwen3-VL checkpoint is `Qwen/Qwen3-VL-8B-Instruct`. Image preprocessing uses the official Qwen3-VL `AutoProcessor` with its default image resizing and normalization pipeline, and decoding uses temperature 0.7, top-p 0.8, and a maximum of 128 output tokens. We perform one evaluation per image. The exact prompts are reproduced below. Text postprocessing for the figures is limited to whitespace normalization and removal of line breaks, without semantic rewriting.

> **System prompt:** You are an assistant for qualitative analysis of B-mode spinal ultrasound images. Describe only visible image features using concise ultrasound terminology. Do not provide diagnostic or navigation-success judgments.
>
> **User prompt:** Provide a concise ultrasound impression of this terminal B-mode image, focusing on image quality, osseous landmarks, posterior acoustic shadowing, and preserved anatomical structure. Write one or two sentences.

C.2 Visualization of CrossWorld Predictions

We further visualize the two complementary prediction mechanisms of CrossWorld. The contact world model is examined through controlled force sweeps on simulated observations, while the spatial world model is examined through action-conditioned rollouts initialized from local real contact observations. Together, the two visualizations illustrate how contact information learned from reality modulates simulated appearance and how spatial dynamics learned in simulation extend locally observed real interactions.

Force-conditioned appearance completion. Figure 12 shows the response of the contact world model to different force conditions. Each row corresponds to an independent simulated B-mode observation. The first column shows the factual simulated frame, and the remaining columns show

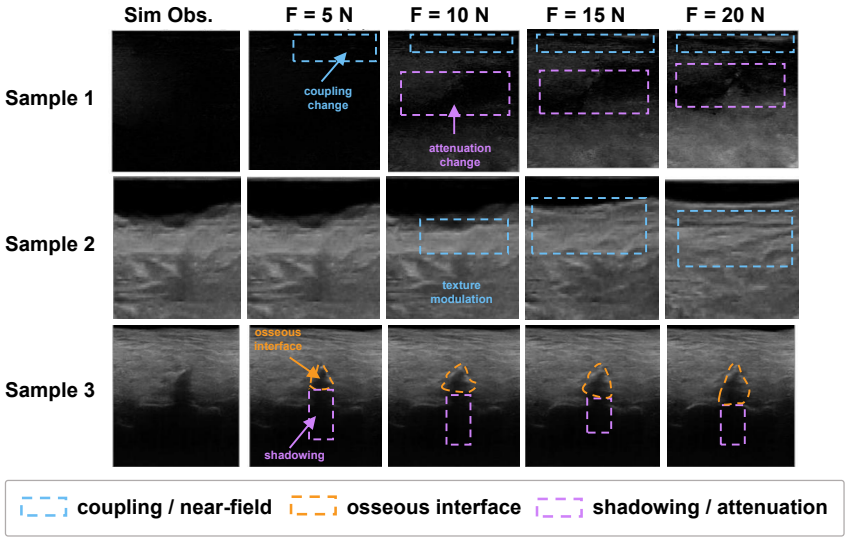


Figure 12: **Force-conditioned appearance completion.** Each row shows one simulated observation followed by contact-completed predictions at $F \in \{5, 10, 15, 20\}$ N. Annotations highlight force-dependent changes in coupling, tissue appearance, osseous interfaces, and posterior attenuation.

contact-completed predictions conditioned on $F \in \{5, 10, 15, 20\}$ N. The predictions exhibit systematic changes in near-field coupling, soft-tissue texture, osseous-interface visibility, and posterior attenuation or shadowing as the force condition changes, while the underlying spatial configuration remains largely preserved. The columns represent controlled contact conditions rather than successive timesteps.

Action-conditioned spatial completion. Figure 13 visualizes spatial rollouts initialized from local real observations in $\mathcal{D}_{\text{contact}}$. For each row, the first column is the factual real B-mode observation, while the subsequent columns are predictions produced by the spatial world model under the executed action sequence. The rollout progressively changes the spatial content of the observation, including soft-tissue appearance and the relative position of bone and posterior structures, thereby extending a local contact sample beyond its original spatial observation. Longer rollouts can accumulate prediction error. We therefore mark, with red dashed boxes, regions in which the generated sequence begins to exhibit visually noticeable rollout drift or degraded anatomical plausibility. These annotations highlight the evolution of prediction quality over the rollout horizon.

C.3 Visualization of Failure Cases

Figure 14 shows representative failure rollouts in simulation and on the real-robot phantom. As in the successful examples, each timestep presents the factual B-mode observation together with its cross-world counterpart. The rightmost panels show the terminal scan and the corresponding qualitative ultrasound assessment. In both cases, the policy fails to recover a clearly identifiable target plane by the end of the rollout. In the simulation example, the terminal scan contains poorly visualized osseous landmarks, indistinct posterior shadowing, and limited anatomical delineation. The real-robot phantom example similarly terminates without a sufficiently clear target-plane appearance. The terminal assessments follow the VLM-based qualitative protocol in Section C.1 and summarize visible image characteristics.

The two examples suggest different limitations of the current co-observation pipeline. In Figure 14(a), an unfavorable initial probe pose can leave the system in a low-information region where few recognizable anatomical landmarks are visible over multiple observations. Under such conditions, the policy receives limited anatomical evidence from the paired observations, making it difficult to recover a meaningful navigation trajectory. In Figure 14(b), the spatially completed view can occasionally emphasize target-like anatomical structure before the factual observation provides sufficient supporting evidence. This may cause the policy to overestimate its progress toward the target plane and converge prematurely to an incorrect location. These cases motivate two possi-

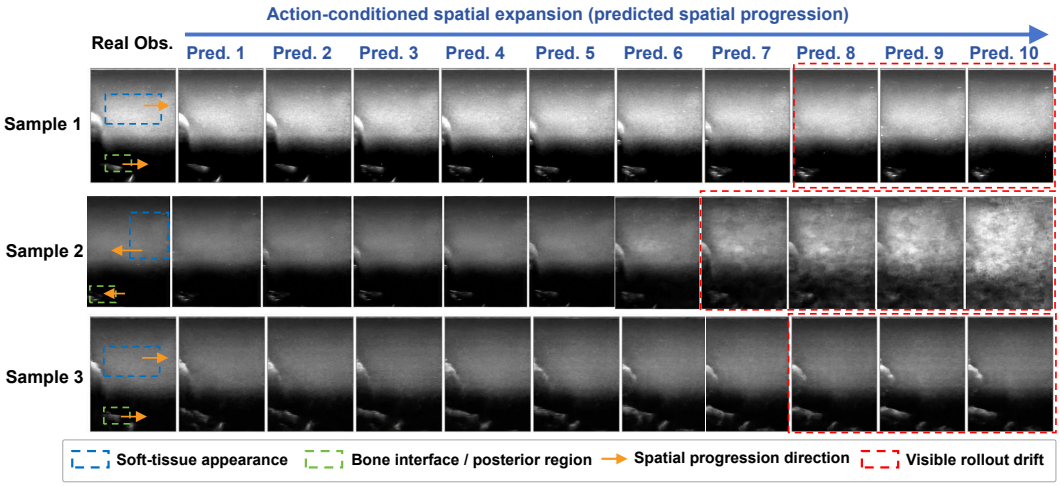


Figure 13: **Action-conditioned spatial completion.** Each row starts from one factual observation in $\mathcal{D}_{\text{contact}}$ and shows the subsequent spatial predictions under action rollout. Blue and green annotations highlight evolving soft-tissue and bone/posterior structures, while red dashed boxes indicate the onset of visible rollout drift in longer predictions.

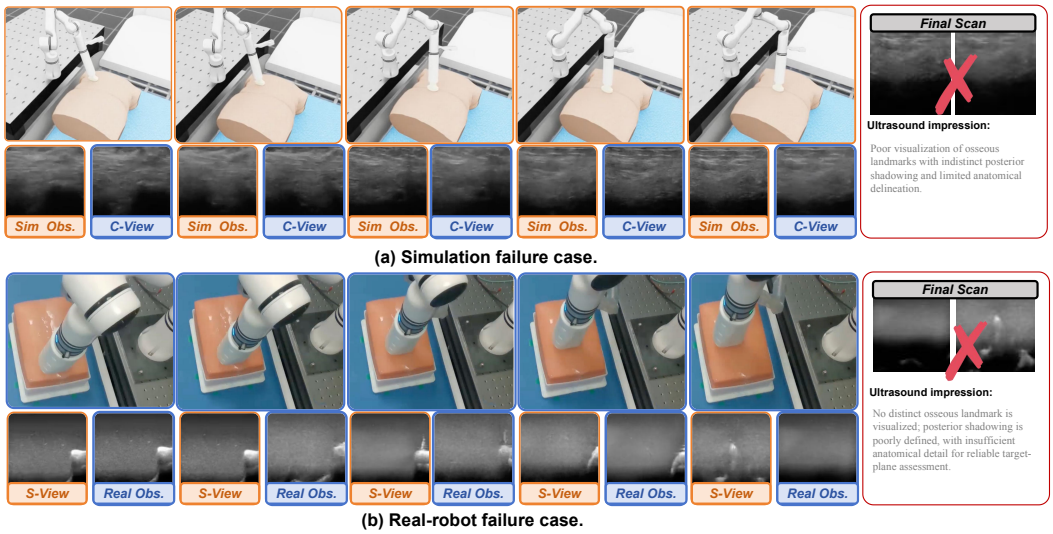


Figure 14: **Representative failure cases.** (a) Simulation failure case. (b) Real-robot phantom failure case. Paired factual and cross-world observations are shown throughout each rollout, and the rightmost panels show the unsuccessful terminal scans with qualitative image assessments.

ble extensions. First, cross-world completion could incorporate additional independent localization cues, such as RGB or depth observations, external optical tracking, or sparse anatomical localization signals, to periodically re-anchor the estimated spatial state. Second, the policy could explicitly estimate the confidence or uncertainty of the generated view and reduce its influence when the completed anatomy is inconsistent with the factual observation. Such confidence-aware fusion may help prevent unreliable spatial completion from dominating the control decision while retaining its benefit when the generated view is informative.

D Discussion and Limitations

D.1 On the Scale of Real-Robot and Human-Subject Data

Real-robot evaluation covers one spine phantom and three human participants. The main training set includes 50 phantom and 60 human expert demonstrations, while the contact model uses 512 additional phantom interaction trajectories. Evaluation comprises 20 phantom and 15 human rollouts per configuration. Because the same participants contribute expert demonstrations and evaluation episodes, human results characterize within-participant navigation; generalization to held-out anatomy is evaluated in simulation. The data-efficiency study varies the number of phantom expert demonstrations while keeping the 60 human demonstrations and contact dataset fixed. Its budget therefore measures complete phantom expert demonstrations rather than total real-data collection. Extending contact acquisition and evaluation to more participants and scanning conditions will test how the framework transfers beyond the current acquisition setting.

D.2 On the Assumptions of Cross-World Data Exchange

CrossWorld uses complementary supervision to learn spatial and contact-dependent predictions, then exchanges these models across environments. The exchange relies on aligned action conventions and preservation of action-relevant information in the generated views. Spatial and contact effects can interact: substantial deformation or contact loss may affect both the image and the appropriate control action. Extending the contact-data coverage and evaluating controlled contact perturbations will help characterize the operating range of the completion models.

The factual image remains available alongside the generated view, and view-dropout consistency trains the policy with both full and masked co-observations. Real spatial completion is estimated from observation history. Sparse anatomical tracking and independent target-plane registration are future directions for evaluating its geometric accuracy.