SURGURDF-CALIB: LEARNING ROBUST GEOMETRIC OPTIMIZATION FOR SURGICAL ROBOT CAMERA CALIBRATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Camera–robot extrinsic calibration is essential for aligning surgical robot kinematics with endoscopic observations, yet remains challenging under specular appearance, articulated motion, occlusion, and partial visibility. We introduce **SurgURDF-Calib**, a geometry-structured learned optimization framework that preserves an analytical SE(3) solver rather than directly regressing camera extrinsics. Given a surgical video, robot joint states, URDF geometry, and an initial extrinsic estimate, we first construct a compact optimization candidate pool using motion diversity and instrument visibility. A learned coarse calibration module then predicts a sequence-level correction to move the estimate into a favorable convergence basin. Starting from this initialization, the second stage performs geometry-guided refinement through observability-aware D-optimal frame scheduling and a learned robust sparse Levenberg–Marquardt (LM) solver, which adaptively estimates geometric factor weights and anisotropic damping while retaining analytical pose updates. We further introduce **SurgURDF-100K**, a geometry-consistent benchmark containing 1,000 surgical sequences and 100K annotated observations. Experiments on synthetic and real surgical videos demonstrate improved calibration accuracy and robustness under challenging initialization and unreliable observations, together with generalization to unseen instrument geometries. Code and video demo are available at `https://surgurdf-calib.pages.dev/`.

1 INTRODUCTION

Camera–robot extrinsic calibration establishes the geometric relationship between the endoscopic camera and the robot coordinate system, enabling robot kinematics to be aligned with surgical video for image-guided perception and manipulation Lu et al. (2022). Accurate geometric alignment is essential in robot-assisted minimally invasive surgery, where reliable instrument localization supports surgical navigation and autonomous robotic tasks Xu et al. (2025); Wu et al. (2025). However, estimating accurate extrinsics from surgical videos remains challenging due to unreliable visual evidence. Surgical instruments are metallic and highly specular, with limited texture, while articulated motions introduce substantial configuration changes over time Shen et al. (2026). Moreover, frequent occlusions and partial visibility make stable geometric correspondences difficult to obtain Wu et al. (2026). Therefore, robust calibration requires not only extracting geometric constraints from observations, but also effectively utilizing heterogeneous measurements during optimization.

Camera–robot calibration has been extensively studied in robotics. Classical hand–eye calibration estimates the rigid transformation between camera and robot coordinates from paired robot and visual motions, typically formulated as the $AX = XB$ problem Tsai & Lenz (1989). Recent markerless approaches incorporate robot kinematics through visual keypoints Lee et al. (2020), differentiable rendering Lu et al. (2023), and temporal robot-structure reasoning Tian et al. (2023); Lu et al. (2025). Geometry-driven methods further exploit explicit robot structures by optimizing the alignment between projected models and visual observations, including render-and-compare refinement Labbé et al. (2021) and multi-frame geometric calibration Xie et al. (2025); Jangir et al. (2025).

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.

In surgical robotics, camera–robot calibration becomes more challenging due to articulated instrument motion, limited visibility, specular surfaces, and kinematic uncertainties. Existing approaches address these difficulties by exploiting surgical-specific constraints, such as RCM-based hand–eye calibration Lu et al. (2022), differentiable rendering for markerless instrument alignment Liang et al. (2025), and sequence-based geometric optimization Cui et al. (2026); Wu et al. (2026). However, these approaches mainly focus on improving geometric representation, while the optimization process remains largely manually designed. When observations are redundant, heterogeneous, or affected by occlusion and visibility changes, fixed optimization strategies may produce unreliable updates and poorly conditioned estimates. This motivates a fundamental question: how can geometric evidence be intelligently selected, weighted, and optimized for robust calibration?

We address this question with **SurgURDF-Calib**, a geometry-structured learned optimization framework for surgical robot–camera extrinsic calibration. Given an initial camera–robot transformation, URDF geometry, joint states, and instrument masks, our goal is to estimate a sequence-shared rigid correction that aligns projected geometry with visual observations. Instead of directly predicting extrinsics, we learn complementary optimization components at three levels: an information-aware scheduler for selecting geometrically informative observations, a learned reliability model for weighting sparse SDF factors, and anisotropic damping for improving six-dimensional optimization. Furthermore, we introduce **SurgURDF-100K**, a geometry-consistent benchmark with 100K observations across five articulated surgical instrument types, providing URDF models, joint states, instrument poses, and camera calibration annotations for supervised refinement.

2 METHOD

2.1 TASK DEFINITION AND PROBLEM FORMULATION

We formulate surgical robot–camera calibration as a geometry-guided SE(3) refinement problem. Given a surgical video sequence with instrument masks $\mathcal{S}$, robot joint trajectories $\mathcal{Q}$, a URDF model $\mathcal{U}$, camera intrinsics K, and an initial camera–robot transformation T^{init}, the goal is to estimate a correction $\Delta T \in SE(3)$ that aligns the projected robot geometry with visual observations.

For each configuration q_t, the URDF defines the instrument geometry through forward kinematics:

$$X_t = F_{\mathcal{U}}(q_t), \qquad P_t(T) = \pi_K(TX_t), \tag{1}$$

where X_t denotes the transformed 3D instrument geometry and $\pi_K(\cdot)$ represents perspective projection. The optimal extrinsic transformation is obtained by minimizing the discrepancy between projected URDF geometry and observed masks:

$$T^* = \arg\min_T \sum_{t=1}^{N} \mathcal{D}(P_t(T), S_t), \qquad T = \Delta T T^{init}. \tag{2}$$

Instead of directly predicting T^*, SurgURDF-Calib performs iterative refinement: $T^{(k+1)} = T^{(k)} \oplus \Delta\xi^{(k)}$, where $\Delta\xi^{(k)}$ denotes the six-dimensional update estimated by the learned optimizer.

2.2 SURGURDF-100K: GEOMETRY-CONSISTENT CALIBRATION BENCHMARK

Large-scale datasets for surgical robot–camera calibration remain limited, as existing benchmarks mainly focus on visual understanding or instrument pose estimation without complete robot-aware geometric supervision. As summarized in Table 1, existing surgical instrument datasets lack essential calibration-related annotations, such as URDF models, joint states, and camera extrinsics, which are required for geometry-guided optimization. To address this gap, we introduce **SurgURDF-100K**, a geometry-consistent synthetic benchmark for surgical robot–camera calibration, as shown in Figure 1. The dataset contains articulated surgical motions rendered with diverse camera viewpoints and appearance variations, together with synchronized URDF models, joint states, instrument poses, and camera calibration annotations across 1,000 surgical sequences with 100K RGB observations. This provides the geometric supervision required for controlled training and evaluation of extrinsic refinement. More details on dataset generation and annotations are provided in Sec. A.

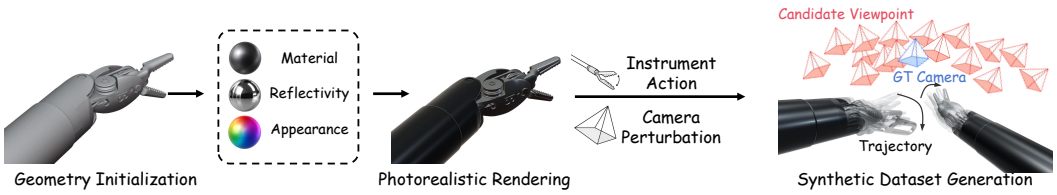


Figure 1: Overview of the SurgURDF-100K dataset generation pipeline. The dataset is synthesized with 5 surgical instruments, diverse camera viewpoints, and URDF-based geometry annotations.

Table 1: Comparison of surgical instrument datasets with geometry and calibration annotations.

Dataset	Size	Instrument	6D Pose	URDF	Joint State	Camera Extrinsic	Real
SurgRIPE Xu et al. (2025)	2K	2	✓	✗	✗	✗	✓
SurgPose Wu et al. (2025)	60K	6	✗	✗	✓	✗	✓
Barragan et al. Barragan et al. (2024)	8K	2	✓	✗	✗	✗	✗
EndoVis2018 Allan et al. (2020)	6K	6	✗	✗	✗	✗	✓
SynSurg6D Shen et al. (2026)	60K	6	✓	✗	✗	✗	✗
SurgURDF-100K (Ours)	**100K**	**5**	✓	✓	✓	✓	✗

2.3 Motion-Visibility-Aware Observation Pool Construction

Dense surgical videos contain abundant temporal observations, but their contributions to extrinsic optimization are highly non-uniform. Frames with similar robot configurations provide redundant constraints, while low-visibility observations introduce unreliable geometric evidence. Therefore, we construct a compact observation pool that preserves motion diversity, temporal coverage, and sufficient visibility before geometric refinement, as shown in Figure 2.

For each frame t of the target manipulator a, we encode its kinematic and temporal properties as:

$$z_t = [(q_{t,a} \oslash s)^\top, \tau_t]^\top, \qquad v_t = \frac{1}{HW} \sum_{u,v} S_{t,a}(u, v), \tag{3}$$

where s denotes the joint scaling vector, τ_t is the normalized timestamp, and v_t measures observation availability from the instrument mask.

We first select temporally distributed high-visibility anchors. Specifically, the sequence is divided into B temporal intervals, and the frame with maximum visibility in each interval is selected: $t_b^* = \arg\max_{t \in B_b} v_t$. Starting from these anchors, we progressively expand the observation pool to capture complementary motion states while maintaining visibility. For each unselected frame, we define:

$$\phi_t = \min_{s \in \mathcal{C}} \|z_t - z_s\|_2 + \beta \frac{v_t}{v_{\max}}, \qquad t^* = \arg\max_t \phi_t. \tag{4}$$

The process continues until reaching a predefined candidate pool size $P = 64$, which provides a compact yet diverse set of candidate observations for subsequent observability-aware scheduling and geometric refinement.

2.4 Step 1: Learned Coarse Extrinsic Initialization

Although the constructed observation pool provides diverse measurements, large initial extrinsic errors may prevent geometry-based optimization from converging to the correct solution. Therefore, we introduce a learned coarse extrinsic initialization module that predicts an initial SE(3) correction and moves the optimization trajectory into a favorable convergence basin, as shown in Figure 2-Step 1. Given K_c sampled observations from the observation pool, we first project the URDF geometry under the initial transformation:

$$S_t^{(0)} = \Pi \left(K T^{init} F_{\mathcal{U}}(q_t) \right), \tag{5}$$

where $S_t^{(0)}$ denotes the projected instrument mask induced by the initial extrinsic transformation.

The coarse module takes the projected masks $S_t^{(0)}$ and observed masks S_t as inputs. A mask encoder extracts frame-wise discrepancy features, which are further processed by a mask-convolution

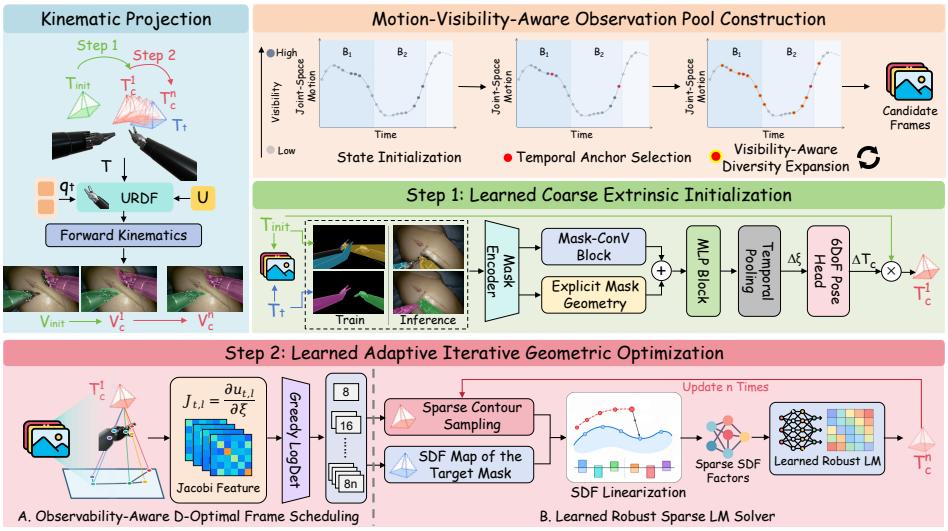


Figure 2: Overview of SurgURDF-Calib. The framework constructs a kinematic-visibility-aware observation pool, followed by Step 1 learned coarse extrinsic correction and Step 2 geometry-guided progressive refinement with informative frame scheduling and robust sparse LM optimization.

branch and an explicit mask-geometry branch. The resulting features are fused, passed through an MLP block, and aggregated across frames by temporal pooling to predict a six-dimensional correction vector $\delta\xi = [\Delta t_x, \Delta t_y, \Delta t_z, \Delta r_x, \Delta r_y, \Delta r_z]$. The predicted update is converted into an SE(3) transformation ΔT_c and applied to the initial estimate: $T^c = \Delta T_c T^{init}$. This coarse initialization module improves global mask-level alignment and provides a better starting point for the subsequent geometry-guided refinement.

2.5 Step 2: Learned Adaptive Iterative Geometric Optimization

Starting from the coarse estimate T^c, we perform learned adaptive iterative geometric optimization to progressively refine the camera–robot alignment. Surgical calibration requires adapting optimization to heterogeneous geometric evidence, where robot configurations offer varying observability of the six-dimensional extrinsic parameters and geometric factors exhibit different reliability due to occlusion, specular appearance, and partial visibility. To address these challenges, we introduce two complementary optimization components: (1) an observability-aware D-optimal frame scheduling strategy that selects informative observations for SE(3) refinement, and (2) a learned robust sparse LM solver that adapts geometric factor reliability and solver conditioning while preserving the analytical optimization structure.

2.5.1 Observability-Aware D-Optimal Frame Scheduling

The observation pool constructed in Sec. 2.3 provides diverse candidates, but their contributions to SE(3) refinement are not explicitly quantified. We therefore select refinement observations according to their expected geometric observability, as shown in Figure 2-Step 2-A. Instead of dense rendering, we exploit URDF link origins as lightweight geometric proxies to efficiently approximate the local sensitivity of each observation. For each link l at timestamp t, the projection Jacobian with respect to the extrinsic perturbation is computed as:

$$J_{t,l}^{uv} = \frac{\partial u_{t,l}}{\partial \delta\xi} \in \mathbb{R}^{2\times 6}, \qquad H_t = \eta_t \sum_l (J_{t,l}^{uv} S)^\top (J_{t,l}^{uv} S), \tag{6}$$

where $\delta\xi$ denotes the six-dimensional extrinsic perturbation, S balances translation and rotation dimensions, and η_t is the visibility factor derived from the instrument mask. The resulting matrix H_t approximates the geometric information contributed by each observation without requiring dense rendering or ground-truth correspondences.

Given the current selected observation set $\mathcal{F}$, we greedily select additional observations by maximizing the marginal increase of a D-optimal objective:

$$t^* = \arg\max_{t \notin \mathcal{F}} \left[\log\det\left(\epsilon I + \sum_{i \in \mathcal{F} \cup \{t\}} H_i\right) - \log\det\left(\epsilon I + \sum_{i \in \mathcal{F}} H_i\right) \right]. \tag{7}$$

The selected observations form nested subsets with increasing budgets of 8, 16, 32, and 64 during refinement. This progressive scheduling strategy enables the optimizer to first exploit highly informative geometric constraints and gradually incorporate additional observations.

2.5.2 Learned Robust Sparse LM Solver

However, fixed LM optimization is insufficient for handling heterogeneous geometric evidence from surgical observations. Given the selected observations, we introduce a learned robust sparse LM solver that preserves the analytical LM update while adapting geometric factor reliability and solver conditioning during refinement, as shown in Figure 2-Step 2-B. For each observation, we construct correspondence-free sparse geometric factors by sampling projected URDF contours and measuring their distances to observed instrument boundaries using signed distance fields (SDF):

$$r_i = D_t(u_i), \qquad u_i = \pi_K(T^{(k)} X_i), \tag{8}$$

where $D_t(\cdot)$ denotes the SDF of the mask and X_i represents a sampled URDF surface point. This correspondence-free formulation directly optimizes URDF-to-mask alignment without requiring explicit point matching. Each geometric factor is linearized around the current estimate as:

$$r_i(\delta\xi) \approx r_i + J_i \delta\xi, \qquad J_i = \nabla D_t(u_i) \frac{\partial u_i}{\partial \delta\xi}. \tag{9}$$

where J_i denotes the residual sensitivity with respect to the SE(3) perturbation.

To adapt optimization to heterogeneous geometric evidence, each factor is encoded using its residual, SDF gradient, geometric location, Jacobian, and kinematic cues. A shared network predicts a reliability score $\hat{w}_i$, while pooled factor features predict a six-dimensional damping vector $\lambda \in \mathbb{R}^6$. The effective factor weight combines the learned reliability with a Cauchy kernel:

$$w_i = \hat{w}_i \frac{1}{1 + (r_i/c)^2}. \tag{10}$$

Using these weights, we construct the normal system with $H = \sum_i w_i J_i^\top J_i$ and $b = \sum_i w_i J_i^\top r_i$. To account for uneven observability across the six pose dimensions, the damping matrix is parameterized by $\Lambda = \mathrm{diag}\big(\lambda \odot \max(\mathrm{diag}(H), 10^{-4}) + 10^{-4}\mathbf{1}\big)$, and the pose increment is obtained analytically from $(H + \Lambda)\delta\xi = -b$. The network therefore learns factor reliability and parameter-wise conditioning, while the pose update remains an explicit six-dimensional geometric solve.

2.6 Training Objective

We train SurgURDF-Calib as an unrolled geometric optimizer. Given synthetic calibration tasks with known camera–robot transformations, the training objective supervises three properties of the learned optimizer: trajectory convergence, geometric consistency, and solver stability. For each refinement stage, we apply SE(3) trajectory supervision. Given the predicted correction $C^{(k)}$ and the ground-truth correction C^{GT}, their relative transformation error is defined as: $E^{(k)} = C^{GT}(C^{(k)})^{-1}$. The trajectory loss is formulated as:

$$\mathcal{L}_{traj} = \sum_{k=1}^{N} \alpha_k \left[\rho\left(\frac{e_t^{(k)}}{\sigma_t}\right) + \rho\left(\frac{e_r^{(k)}}{\sigma_r}\right) \right], \tag{11}$$

Table 2: Quantitative comparison of camera extrinsic calibration methods on synthetic and real surgical videos. Synthetic data are evaluated using ground-truth camera parameters with E_R, E_T, E_{2D}, E_{3D}, E_{tip}, and IoU, while real data are evaluated using masks with IoU and E_{flow}.

Method	Synthetic Data						Real Data	
	$\mathbf{E_R}\downarrow$	$\mathbf{E_T}\downarrow$	$\mathbf{E_{2D}}\downarrow$	$\mathbf{E_{3D}}\downarrow$	$\mathbf{E_{tip}}\downarrow$	$\mathbf{IoU}\uparrow$	$\mathbf{IoU}\uparrow$	$\mathbf{E_{flow}}\downarrow$
DREAM Lee et al. (2020)	8.407	19.528	32.208	14.995	12.325	0.576	0.294	2.973
RoboPose Labbé et al. (2021)	10.534	20.438	**29.834**	15.216	11.831	0.520	**0.341**	2.529
EasyHeC Chen et al. (2023)	**6.996**	**19.712**	80.889	18.759	16.769	0.611	0.283	2.901
CalibAll Xie et al. (2025)	11.022	28.189	49.076	19.382	14.196	0.519	0.268	2.404
RobotArena∞ Jangir et al. (2025)	8.342	23.138	47.763	17.863	13.948	0.559	0.235	2.733
SurgCalib Wu et al. (2026)	8.137	21.844	37.848	**15.204**	**11.407**	**0.691**	0.308	**2.445**
SurgURDF-Calib (Ours)	**4.844**	**13.110**	**10.142**	**7.304**	**4.310**	**0.836**	**0.724**	**1.408**

where $\rho(\cdot)$ denotes Smooth L1 loss and α_k assigns larger weights to later refinement stages.

The coarse initialization module is additionally supervised using the same SE(3) objective: $\mathcal{L}_{init}$, encouraging a favorable starting point for subsequent refinement. We further introduce image-space geometric consistency: $\mathcal{L}_{geo} = \mathcal{L}_{mask} + \mathcal{L}_{contour}$, where $\mathcal{L}_{mask}$ enforces region-level consistency between projected URDF geometry and observed instrument masks, while $\mathcal{L}_{contour}$ promotes boundary alignment through SDF-based geometric constraints.

To regularize the learned solver behavior, we introduce solver supervision:

$$\mathcal{L}_{solver} = \frac{1}{NB}\sum_{k,b}(\bar{w}_b^{(k)} - \mu)^2 - \gamma\frac{1}{NB}\sum_{k,b}\log\det\left(SH_b^{(k)}S + \epsilon I\right), \tag{12}$$

where $\bar{w}_b^{(k)}$ denotes the average effective geometric weight at refinement stage k, and $H_b^{(k)}$ represents the corresponding geometric information matrix. The first term avoids excessive suppression of geometric factors, and the second term encourages sufficient observability throughout refinement. The overall training objective is: $\mathcal{L} = \lambda_t\mathcal{L}_{traj} + \mathcal{L}_{init} + \mathcal{L}_{geo} + \lambda_s\mathcal{L}_{solver}$.

3 Experiments and Results

3.1 Implementation Details

Datasets and Evaluation Protocol. We evaluate SurgURDF-Calib on the proposed SurgURDF-100K benchmark and three real surgical video datasets. SurgURDF-100K is split into 900 training sequences and 100 testing sequences with separated sequence identities and motion templates. For real-world evaluation, we use videos from SurgSync Zhou et al. (2026), SutureBot Haworth et al. (2026), and ImitateChole Hansen et al. (2026), covering ex vivo suturing, simulated suturing, and in vivo tissue manipulation scenarios. Since real procedures lack ground-truth camera extrinsics, instrument masks are obtained using SAM3-based segmentation with manual correction. For synthetic data with known calibration ground truth, we report rotation error E_R ($^\circ$), translation error E_T (mm), 2D reprojection error E_{2D} (px), 3D URDF error E_{3D} (mm), tool-tip position error E_{tip} (mm), and rendered mask IoU. For real videos, we evaluate geometric alignment using mask IoU and temporal motion consistency error E_{flow}. The detailed formulations of all evaluation metrics are provided in Sec. C.

Baselines and Training Details. We compare SurgURDF-Calib with representative geometry-based and learning-based calibration methods, including DREAM Lee et al. (2020), RoboPose Labbé et al. (2021), EasyHeC Chen et al. (2023), CalibAll Xie et al. (2025), RobotArena∞ Jangir et al. (2025), and SurgCalib Wu et al. (2026). All methods are evaluated under the same training split, initialization conditions, URDF models, camera intrinsics, and evaluation metrics. We train SurgURDF-Calib with AdamW for 10 epochs on a single NVIDIA H100 GPU. The coarse initialization and learned LM solver use learning rates of 5×10^{-5} and 2×10^{-4}, respectively, with a weight decay of 10^{-4} and gradient clipping of 5.0. The loss weights are set to $(\lambda_t, \lambda_s) = (10, 0.01)$.

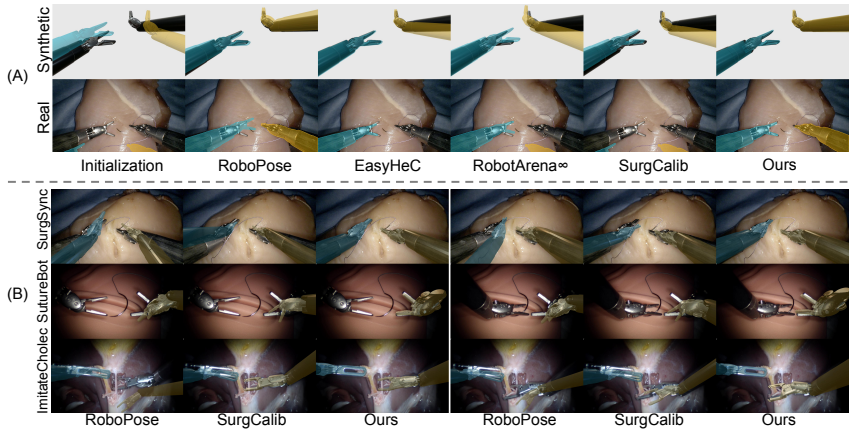


Figure 3: Qualitative comparison of SurgURDF-Calib with existing calibration methods on synthetic and real surgical sequences. The visualization shows URDF-based instrument projections under the estimated camera extrinsics.

Table 3: Progressive ablation of candidate-pool construction, coarse initialization, learned robust sparse LM, and D-optimal scheduling.

Candidate Pool	Coarse Init.	D-Optimal Scheduling	Robust Sparse LM	$E_R \downarrow$	$E_T \downarrow$	$E_{3D} \downarrow$	IoU ↑
×	×	×	×	9.158	25.672	23.206	0.195
×	✓	×	×	8.653	19.849	19.947	0.505
✓	✓	×	×	8.474	17.975	18.035	0.598
✓	✓	×	✓	**6.857**	**15.319**	**15.450**	**0.636**
✓	✓	✓	✓	**4.844**	**13.110**	**7.304**	**0.836**

During inference, each 100-frame sequence first constructs a 64-frame observation pool using motion and visibility cues. The coarse initialization uses 8 observations, followed by four unrolled LM refinement stages with nested D-optimal subsets of 8, 16, 32, and 64 observations.

3.2 Comparison with Existing Calibration Methods

Quantitative Comparison. We first evaluate whether the proposed geometry-aware optimization framework improves surgical robot–camera extrinsic calibration compared with existing approaches. As summarized in Table 2, SurgURDF-Calib consistently achieves lower calibration errors and better URDF projection alignment than representative learning-based and geometry-based calibration methods. On the synthetic benchmark with complete geometric annotations, our method reduces the rotation error E_R from 8.137° to 4.844°, translation error E_T from 21.844 mm to 13.110 mm, and URDF alignment error E_{3D} from 15.204 mm to 7.304 mm compared with the previous surgical calibration method SurgCalib. On real surgical videos, SurgURDF-Calib further achieves an IoU of 0.724 and an E_{flow} of 1.408, demonstrating improved projection and temporal consistency.

Qualitative Visualization. Figure 3-A provides qualitative comparisons of URDF projection alignment under estimated camera extrinsics on both synthetic and real surgical sequences. Compared with existing calibration approaches, SurgURDF-Calib produces more accurate alignment between projected URDF geometries and observed instruments, especially around articulated joints and tooltip regions. The improvement is more evident in real surgical videos, where visual evidence is affected by occlusion, specular appearance, and partial visibility. Furthermore, Figure 3-B presents results across three different real surgical scenarios. SurgURDF-Calib consistently maintains accurate geometric correspondence between URDF projections and surgical instruments, demonstrating robustness to diverse surgical environments and instrument motions.

3.3 Analysis of Learned Geometric Optimization

Progressive Component Analysis. Table 3 shows that each added component improves all reported metrics. The initial extrinsic estimates, before any correction, yield $E_R = 9.158°$, $E_T = 25.672$ mm, and IoU=0.195. Coarse initialization and candidate-pool construction increase IoU to 0.505 and 0.598, respectively. Adding learned robust sparse LM with candidate-uniform scheduling further

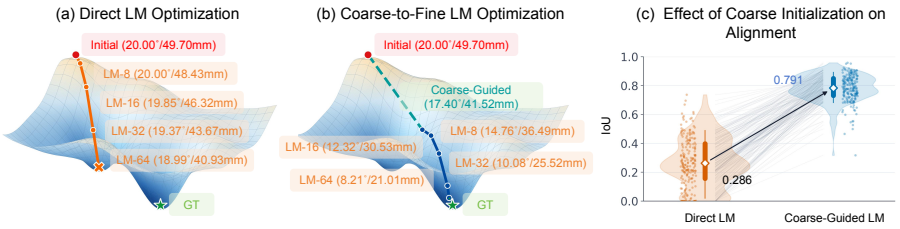


Figure 4: **Effect of coarse initialization on geometric optimization convergence.** Under paired 20°/30 mm perturbations, (a,b) compare direct and coarse-to-fine LM using E_R and E_T, while (c) reports IoU. The 30 mm denotes the applied translation component, with rotation composition yielding a median initial translation error of 49.70 mm.

Table 4: Ablation study of observability-aware scheduling and robust sparse LM solver.

Module	$E_R \downarrow$	$E_T \downarrow$	$E_{2D} \downarrow$	$E_{3D} \downarrow$	$E_{tip} \downarrow$	IoU ↑
Different Observation Selection						
Random-uniform	7.988	16.034	28.322	16.312	9.415	0.632
Candidate-uniform	6.857	15.319	26.127	15.450	9.048	0.636
Single-frame Information	7.638	15.757	24.968	16.349	**8.062**	0.693
Visibility-only	**6.540**	**14.656**	**19.058**	**12.197**	8.834	**0.740**
D-optimal (Ours)	**4.844**	**13.110**	**10.142**	**7.304**	**4.310**	**0.836**
Different LM Optimizations						
Classical Robust LM	7.959	17.426	20.315	21.888	16.742	0.467
Learned Scalar Damping	7.235	19.472	19.972	17.240	13.288	0.467
Learned Anisotropic Damping	7.059	17.347	18.959	17.328	10.227	0.486
Learned Weighting	5.846	16.256	11.170	9.594	**6.125**	0.803
Learned Weighting + Scalar Damping	**5.144**	**14.279**	**12.408**	**8.955**	**5.793**	**0.812**
Full Learned LM (Ours)	**4.844**	**13.110**	**10.142**	**7.304**	**4.310**	**0.836**

improves IoU to 0.636. Finally, D-optimal scheduling reduces E_R to 4.844°, E_T to 13.110 mm, and E_{3D} to 7.304 mm, with IoU=0.836. These results suggest that effective geometric refinement depends on both a suitable initialization and informative observation scheduling.

Analysis of Observation Scheduling and Learned LM Optimization. Table 4 compares observation scheduling strategies and LM variants. D-optimal scheduling outperforms visibility-only scheduling across all six metrics, reducing E_{3D} from 12.197 to 7.304 mm and increasing IoU from 0.740 to 0.836. For the solver, learning damping alone yields limited improvements in projection alignment, whereas learned factor weighting increases IoU from 0.467 to 0.803 over classical robust LM. Combining weighting with scalar damping further improves most metrics, although E_{2D} increases from 11.170 to 12.408 px. Replacing scalar damping with anisotropic damping improves all six metrics, reducing E_T from 14.279 to 13.110 mm and increasing IoU from 0.812 to 0.836. These results suggest that factor weighting drives the main alignment gains, with further improvements from parameter-wise damping.

Coarse Initialization Enables Reliable Optimization. We investigate the effect of coarse initialization under challenging extrinsic perturbations. Both methods start from identical initial extrinsics generated by left-multiplying the ground-truth transformations with perturbations having a rotation magnitude of 20° and a translation component of 30 mm. As shown in Figure 4-(a), direct LM refinement without coarse initialization produces an inaccurate estimate. In contrast, the coarse-to-fine strategy reduces the errors to $E_R = 8.21°$ and $E_T = 21.01$ mm (Figure 4-(b)), while improving IoU from 0.286 to 0.791 (Figure 4-(c)). These results demonstrate that coarse initialization improves geometric refinement under large initial perturbations.

3.4 Robustness and Generalization Analysis

Progressive Optimization Convergence. We analyze how calibration performance evolves with increasing observation budgets. Starting from the same initial perturbation, progressively expanding the selected set from 8 to 64 frames consistently improves all metrics, reducing E_R from 7.267° to 4.844° and increasing IoU from 0.650 to 0.836, as shown in Figure 5. We additionally refine using all 100 frames as an analysis case. The slight performance degradation beyond 64 frames shows that more observations are not necessarily more informative, as redundant or unreliable frames can

Table 5: Cross-instrument generalization evaluation on unseen surgical instruments.

Held-out Instrument	$\mathbf{E_R} \downarrow$	$\mathbf{E_T} \downarrow$	$\mathbf{E_{2D}} \downarrow$	$\mathbf{E_{3D}} \downarrow$	$\mathbf{E_{tip}} \downarrow$	IoU ↑
420006 · Large Needle Driver	6.959	17.958	15.623	10.788	6.630	0.771
420172 · Maryland Bipolar Forceps	5.619	16.089	14.660	8.900	5.397	0.775
420093 · ProGrasp Forceps	6.224	16.646	11.498	9.476	5.919	0.781
420049 · Cadiere Forceps	6.935	18.051	21.725	11.344	7.598	0.752
420327 · Medium-Large Clip Applier	5.116	14.940	10.642	8.532	5.266	0.828
All Held-out Instruments	**6.171**	**16.737**	**14.830**	**9.808**	**6.162**	**0.782**

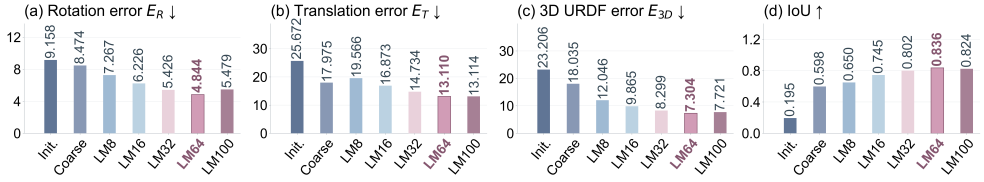


Figure 5: Convergence behavior of the proposed coarse-to-fine calibration framework. Starting from the same initial perturbation, performance improves progressively from 8 to 64 frames, while using all 100 frames causes slight degradation due to redundant or unreliable observations.

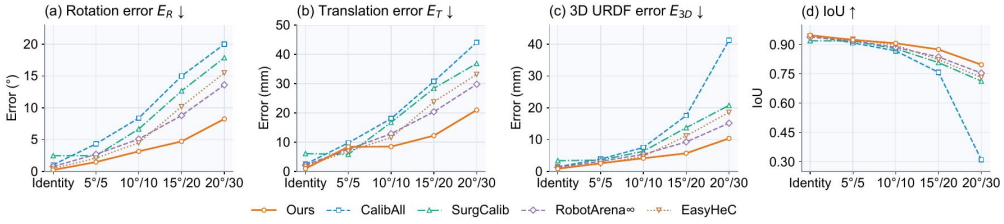


Figure 6: Comparison of camera extrinsic calibration methods under different initial perturbations.

adversely affect geometric optimization. This finding further motivates the proposed candidate-pool construction, which retains a compact set of complementary observations before refinement.

Robustness to Initialization Perturbations. We evaluate the robustness of SurgURDF-Calib and other baselines under different initial extrinsic conditions by gradually increasing rotation and translation perturbations from identity initialization to 20° and 30 mm. As shown in Figure 6, our method consistently achieves accurate calibration across both small and large initialization errors, while competing approaches exhibit increasing performance degradation under larger perturbations.

Generalization to Unseen Surgical Instruments. To evaluate whether the learned optimizer captures transferable geometric properties rather than instrument-specific patterns, we perform a zero-shot cross-instrument evaluation. During training, one instrument type is held out, while the remaining four instruments are used for model optimization. As shown in Table 5, SurgURDF-Calib maintains consistent performance across different unseen instrument geometries, achieving an average E_R of 6.171° and IoU of 0.782. These results demonstrate that our framework learns transferable geometric reasoning and generalizes beyond the instrument configurations observed during training.

4 Conclusion

We presented **SurgURDF-Calib**, a geometry-structured learned optimization framework for surgical robot–camera extrinsic calibration. Rather than directly regressing extrinsics, our method retains an analytical SE(3) solver and performs coarse-to-fine refinement through motion–visibility-aware candidate selection, learned coarse initialization, and observability-aware robust sparse LM optimization. Together with **SurgURDF-100K**, experiments on synthetic and real surgical videos demonstrate improved robustness to challenging initialization and unreliable geometric observations, as well as generalization to unseen instruments. Beyond calibration, accurate URDF–camera alignment also provides a geometric foundation for constructing physically grounded surgical world models with explicit robot structure and motion constraints. Despite the strong performance, future work could further improve calibration around weakly observable articulated joints.

AI USE STATEMENT

Generative AI tools were used for language polishing, including improving grammar, clarity, and readability of the manuscript. They were not used to generate experimental results, conduct data analysis, or make scientific claims. All AI-assisted text was carefully reviewed and revised by the authors, who take full responsibility for the final content of the paper.

ETHICS STATEMENT

This work does not involve new human-subject recruitment or clinical intervention. The experiments use synthetic data and previously released real-world surgical datasets for research purposes. No personally identifiable information is used in our study. The proposed method is intended for research on surgical robot calibration and does not directly provide clinical diagnosis or treatment decisions.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the SurgURDF-Calib framework, optimization formulation, training procedure, and evaluation protocol in the main paper and appendix. Additional details on synthetic data generation, real-world datasets, network architectures, geometric factor construction, optimization parameters, and evaluation metrics are provided in the supplementary material. Code and dataset resources will be released to facilitate reproduction of the reported results.

REFERENCES

Max Allan, Satoshi Kondo, Sebastian Bodenstedt, Stefan Leger, Rahim Kadkhodamohammadi, Imanol Luengo, Felix Fuentes, Evangello Flouty, Ahmed Mohammed, Marius Pedersen, et al. 2018 robotic scene segmentation challenge. *arXiv preprint arXiv:2001.11190*, 2020.

Juan Antonio Barragan, Jintan Zhang, Haoying Zhou, Adnan Munawar, and Peter Kazanzides. Realistic data generation for 6d pose estimation of surgical instruments. In *2024 IEEE international conference on robotics and automation (ICRA)*, pp. 13347–13353. IEEE, 2024.

Linghao Chen, Yuzhe Qin, Xiaowei Zhou, and Hao Su. Easyhec: Accurate and automatic hand-eye calibration via differentiable rendering and space exploration. *IEEE Robotics and Automation Letters*, 8(11):7234–7241, 2023.

Zejian Cui et al. On-the-fly hand-eye calibration for the da vinci surgical robot. *arXiv preprint arXiv:2601.14871*, 2026.

Pascal Hansen, Ji Woong Brian Kim, Antony Goldenberg, Juo Tung Chen, Yuanzhe Amos Li, Anton Deguet, Brandon White, De Ru Tsai, Richard Cha, Jeffrey Jopling, et al. Imitatecholec: A multimodal dataset for long-horizon imitation learning in robotic cholecystectomy. *Scientific Data*, 13 (1):210, 2026.

Jesse Haworth, Juo-Tung Chen, Nigel Nelson, Ji Woong Kim, Masoud Moghani, Chelsea Finn, and Axel Krieger. Suturebot: A precision framework & benchmark for autonomous end-to-end suturing. *Advances in Neural Information Processing Systems*, 38, 2026.

Yash Jangir, Yidi Zhang, Kashu Yamazaki, Chenyu Zhang, Kuan-Hsun Tu, Tsung-Wei Ke, Lei Ke, Yonatan Bisk, and Katerina Fragkiadaki. Robotarena∞: Scalable robot benchmarking via real-to-sim translation. *arXiv preprint arXiv:2510.23571*, 2025.

Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Single-view robot pose and joint angle estimation via render & compare. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1654–1663. IEEE, 2021.

Timothy E Lee, Jonathan Tremblay, Thang To, Jia Cheng, Terry Mosier, Oliver Kroemer, Dieter Fox, and Stan Birchfield. Camera-to-robot pose estimation from a single image. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9426–9432. IEEE, 2020.

Zekai Liang, Zih-Yun Chiu, Florian Richter, and Michael C Yip. Differentiable rendering-based pose estimation for surgical robotic instruments. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 20898–20905. IEEE, 2025.

Bo Lu, Bin Li, Qi Dou, and Yunhui Liu. A unified monocular camera-based and pattern-free hand-to-eye calibration algorithm for surgical robots with rcm constraints. *IEEE/ASME Transactions on Mechatronics*, 27(6):5124–5135, 2022.

Jingpei Lu, Florian Richter, and Michael C Yip. Markerless camera-to-robot pose estimation via self-supervised sim-to-real transfer. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21296–21306. IEEE, 2023.

Jingpei Lu, Zekai Liang, Tristin Xie, Florian Richter, Shan Lin, Sainan Liu, and Michael C Yip. Ctrnet-x: Camera-to-robot pose estimation in real-world conditions using a single camera. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1914–1920. IEEE, 2025.

Daiyun Shen, Shuojue Yang, Chang Han Low, Qian Li, Mengya Xu, Qi Dou, and Yueming Jin. Surfsurg6d: Geometry consistent dense correspondence for textureless surgical instrument pose estimation. *IEEE Robotics and Automation Letters*, 2026.

Yang Tian, Jiyao Zhang, Zekai Yin, and Hao Dong. Robot structure prior guided temporal attention for camera-to-robot pose estimation from image sequence. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8917–8926. IEEE, 2023.

Roger Y Tsai and Reimar K Lenz. A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, 5(3):345–358, 1989.

Zijian Wu, Adam Schmidt, Randy Moore, Haoying Zhou, Alexandre Banks, Peter Kazanzides, and Septimiu E Salcudean. Surgpose: a dataset for articulated robotic surgical tool pose estimation and tracking. In *2025 IEEE International conference on robotics and automation (ICRA)*, pp. 10507–10514. IEEE, 2025.

Zijian Wu, Shuojue Yang, Yu Chung Lee, Eitan Prisman, Yueming Jin, and Septimiu E Salcudean. Surgcalib: Gaussian splatting-based hand-eye calibration for robot-assisted minimally invasive surgery. *arXiv preprint arXiv:2603.08983*, 2026.

Sicheng Xie, Lingchen Meng, Zijie Diao, Haidong Cao, Zhiying Du, Shuyuan Tu, Jiaqi Leng, Qiuyue Wang, Mingsheng Li, Shuai Bai, et al. Unify robot actions in camera frame. *arXiv preprint arXiv:2511.17001*, 2025.

Haozheng Xu, Alistair Weld, Chi Xu, Alfie Roddan, João Cartucho, Mert Asim Karaoglu, Alexander Ladikos, Yangke Li, Yiping Li, Daiyun Shen, et al. Surgripe challenge: Benchmark of surgical robot instrument pose estimation. *Medical Image Analysis*, 105:103674, 2025.

Haoying Zhou, Chang Liu, Yimeng Wu, Junlin Wu, Zijian Wu, Yu Chung Lee, Sara Martuscelli, Spetimiu E Salcudean, Gregory S Fischer, and Peter Kazanzides. Surgsync: Time-synchronized multi-modal data collection framework and dataset for surgical robotics. *arXiv preprint arXiv:2603.06919*, 2026.

APPENDIX

A DATASET SUPPLEMENT

A.1 DETAILS ON SYNTHETIC DATASET CONSTRUCTION

SurgURDF-100K combines recorded robot motions with URDF-defined instrument geometry and calibrated stereo rendering. The construction pipeline transfers constrained joint motions to predefined camera–robot configurations, renders articulated instrument meshes, and generates synchronized geometric annotations.

Motion sources and preprocessing. We construct 1,000 bimanual sequences using 200 motion windows from the existing surgical videos. Each sequence contains 100 frames sampled at 10 Hz, corresponding to approximately 10 seconds. When motion amplitude, velocity, or acceleration exceeds the prescribed limits, a shared scaling factor is applied to both manipulators. The resulting motions are therefore derived from real robot recordings through constrained transfer, rather than exact reproductions of the original trajectories.



Figure 7: Five da Vinci instrument geometries used in SurgURDF-100K.

Instrument geometry and motion transfer. Each scene contains two manipulators, with the instrument for each arm sampled uniformly from five types: Large Needle Driver, Maryland Bipolar Forceps, ProGrasp Forceps, Cadiere Forceps, and Medium-Large Clip Applier, as shown in Figure 7. This sampling produces diverse bimanual instrument configurations with inter-instrument occlusion. URDF models specify the kinematic chains, link transformations, and associated instrument meshes; forward kinematics determines the articulated geometry at each timestamp.

For transferred trajectories, the synthetic joint configuration is constructed as

$$q_{\mathrm{syn}}(t) = q_{\mathrm{template}}(0) + \alpha_{\mathrm{m}} \left[\widetilde{q}_{\mathrm{source}}(t) - \widetilde{q}_{\mathrm{source}}(0)\right], \tag{13}$$

where $\widetilde{q}_{\mathrm{source}}$ denotes the preprocessed source trajectory, $q_{\mathrm{template}}(0)$ specifies the initial configuration, and α_{m} controls motion amplitude.

Camera and scene configuration. Camera parameters and manipulator placements are drawn from a pool of up to 97 previously accepted LND scene templates. Camera extrinsics and manipulator root transformations remain fixed within each sequence, while joint motion changes the instrument configurations. Stereo images are rendered at 1280×720 resolution using known intrinsics shared by both cameras and a fixed stereo transformation with an approximately 4.18 mm baseline. Viewpoint diversity arises from the template pool. SutureBot contributes relative joint motion; its unavailable source-camera calibration is not required for constructing the synthetic scenes.

Training sequences are generated using nvdiffrast GPU rasterization with a custom physically based surgical-steel shader. At each timestamp, forward kinematics transforms the link meshes into the camera frame. Rasterization and depth buffering resolve self-occlusion and occlusion between instruments. Interpolated surface positions and smooth vertex normals are used for shading, followed by antialiasing of RGB values and silhouette coverage. Component-specific base color, metallicity, and roughness represent coated shafts, steel wrists, and polished jaws. GGX specular shading, Fresnel reflection, procedural surgical-light panels, and environment reflections produce pose-dependent highlights. Sequences use a near-black background and largely fixed material and lighting configurations; no tissue backgrounds are composited.

Table 6: Instrument coverage of the three real-world data sources used to construct our 600-sequence test set.

	Target Instrument				
Data Source	**Large Needle Driver**	**Maryland Bipolar Forceps**	**ProGrasp Forceps**	**Cadiere Forceps**	**Medium-Large Clip Applier**
SurgSync Zhou et al. (2026)	✓	✓	✓	–	–
SutureBot Haworth et al. (2026)	✓	–	–	–	–
ImitateCholec Hansen et al. (2026)	–	–	–	✓	✓

A.2 Details of the real-world dataset

We construct a real-world test set comprising 600 sequences from SurgSync (Zhou et al., 2026), SutureBot (Haworth et al., 2026), and ImitateCholec (Hansen et al., 2026). SurgSync provides ex-vivo, tabletop phantom, and clinical recordings; SutureBot contributes robotic suturing demonstrations from 10 tissue samples; and ImitateCholec contributes recordings from the tissue subset. The instrument coverage of these sources is summarized in Table 6. All 600 sequences are used exclusively for testing. Since ground-truth extrinsics are unavailable, evaluation focuses on agreement between projected URDF geometry and visual observations. The same available masks are used for calibration and scoring, so mask IoU measures within-video calibration consistency.

B Model details

B.1 Step 1: Learned Coarse Extrinsic Initialization

Table 7 details the coarse initializer, including its input construction, network layers, and output dimensions. For each target arm, we uniformly select eight entries from the 64-frame candidate pool and pair the observed masks with URDF projections under the initial extrinsics. The source projections account for occlusion between the two arms. The network processes the source and target masks, their difference, and normalized image coordinates to characterize the initial projection discrepancy.

Two parallel branches provide complementary information: the convolutional branch extracts spatial features, while the explicit geometry branch summarizes mask area, centroid, and second-order moments. These features are concatenated with normalized timestamps and target-arm identity, then embedded into frame tokens. Learned attention pooling aggregates the eight tokens into a sequence-level representation.

The pose head predicts a bounded translation and rotation vector, which are converted into a rigid correction C_{coarse}. This correction is shared across the sequence and left-multiplied onto the target-arm extrinsics: $T_a^{\text{coarse}} = C_{\text{coarse}} T_a^{\text{init}}$. The resulting estimate initializes the subsequent geometric refinement.

B.2 Step 2: Learned Geometric Refinement

Starting from the coarse correction C_a^0, refinement produces four successive corrections $C_a^1, \ldots, C_a^4$ for target arm a. Each correction is shared across the sequence and applied to the nominal camera-from-root transformation $B_{t,a}$. Only the target-arm extrinsics are updated; the other arm remains part of the geometry used for occlusion estimation. The network predicts factor reliability and damping, while pose increments are obtained by solving a six-dimensional linear system.

Fixed observability-aware frame scheduling. At the coarse estimate, we project the seven modeled link origins and compute their projection Jacobians $A_{t,\ell} \in \mathbb{R}^{2\times 6}$ with respect to left-multiplicative perturbations. Translation components are expressed in millimeters and rotation-vector components in degrees. With $S = \text{diag}(10, 10, 10, 5, 5, 5)$, the per-frame information proxy is

$$H_t^{\text{proxy}} = \nu_t \sum_{\ell} \mathbf{1}[Z_{t,\ell} > 0.01](A_{t,\ell} S)^\top (A_{t,\ell} S), \tag{14}$$

Table 7: Architecture of the learned coarse extrinsic initializer. Module colors correspond to the framework diagram.

Layer / Function	Detail	Output
	Mask Encoder	
Warm-frame selection	Uniformly sample 8 entries from the 64-frame candidate pool	$B \times 8$
Source mask	Occlusion-aware sparse URDF projection Bilinear splatting $\rightarrow 1 - e^{-2d}$ 5×5 max pooling $\rightarrow$ threshold at 0.2	$B \times 8 \times 1 \times 180 \times 320$
Target mask	Observed target-arm masks at the selected frames	$B \times 8 \times 1 \times 180 \times 320$
Conv input	$\mathrm{Concat}[S_{\mathrm{src}}, S_{\mathrm{tgt}}, S_{\mathrm{tgt}} - S_{\mathrm{src}}, x, y]$ Merge batch and frame axes	$8B \times 5 \times 180 \times 320$
	Mask-Conv Block	
Conv 1	$\mathrm{Conv2d}(5 \rightarrow 24,\ k = 5,\ s = 2,\ p = 2)$ $\mathrm{GN}(6) \rightarrow \mathrm{GELU}$	$8B \times 24 \times 90 \times 160$
Conv 2	$\mathrm{Conv2d}(24 \rightarrow 48,\ k = 3,\ s = 2,\ p = 1)$ $\mathrm{GN}(8) \rightarrow \mathrm{GELU}$	$8B \times 48 \times 45 \times 80$
Conv 3	$\mathrm{Conv2d}(48 \rightarrow 80,\ k = 3,\ s = 2,\ p = 1)$ $\mathrm{GN}(10) \rightarrow \mathrm{GELU}$	$8B \times 80 \times 23 \times 40$
Conv 4	$\mathrm{Conv2d}(80 \rightarrow 128,\ k = 3,\ s = 2,\ p = 1)$ $\mathrm{GN}(16) \rightarrow \mathrm{GELU}$	$8B \times 128 \times 12 \times 20$
Visual token	Global average pooling Restore batch and frame axes	$B \times 8 \times 128$
	Explicit Mask Geometry	
Source moments	$g_{\mathrm{src}} = g(S_{\mathrm{src}})$	$B \times 8 \times 6$
Target moments	$g_{\mathrm{tgt}} = g(S_{\mathrm{tgt}})$	$B \times 8 \times 6$
Geometry difference	$\Delta g = [\Delta A, \Delta c_x, \Delta c_y]$	$B \times 8 \times 3$
Geometry feature	$\mathrm{Concat}[g_{\mathrm{src}}, g_{\mathrm{tgt}}, \Delta g]$	$B \times 8 \times 15$
	MLP Block	
Time feature	$\tau_f = 2(f - f_{\min})/(f_{\max} - f_{\min}) - 1$	$B \times 8 \times 1$
Arm feature	Target-arm one-hot encoding	$B \times 8 \times 2$
Concatenation	Concatenate visual, geometry, time, and arm features $128 + 15 + 1 + 2 = 146$	$B \times 8 \times 146$
Token embedding 1	$\mathrm{Linear}(146 \rightarrow 192)$ $\mathrm{LN}(192) \rightarrow \mathrm{GELU}$	$B \times 8 \times 192$
Token embedding 2	$\mathrm{Linear}(192 \rightarrow 192) \rightarrow \mathrm{GELU}$	$B \times 8 \times 192$
	Temporal Pooling	
Frame scoring	$s_f = \mathrm{Linear}(192 \rightarrow 1)(h_f)$	$B \times 8 \times 1$
Attention weights	$\alpha_f = \exp(s_f) / \sum_{j=1}^{8} \exp(s_j)$	$B \times 8 \times 1$
Weighted sum	$h_{\mathrm{seq}} = \sum_{f=1}^{8} \alpha_f h_f$	$B \times 192$
	6DoF Pose Head	
Pose embedding	$\mathrm{Linear}(192 \rightarrow 192) \rightarrow \mathrm{GELU}$	$B \times 192$
Pose projection	$[z_t, z_r] = \mathrm{Linear}(192 \rightarrow 6)(\cdot)$	$B \times 6$
Bounded output	$\Delta t = 36\ \tanh(z_t)$ mm $\Delta r = 18\ \tanh(z_r)\ ^\circ$	$B \times 6$
Transform construction	Convert translation to meters and rotation to radians Rotation-vector conversion; assemble C_{coarse}	$B \times 4 \times 4$
Extrinsic update	$T_a^{\mathrm{coarse}} = C_{\mathrm{coarse}} T_a^{\mathrm{init}}$	$B \times 4 \times 4$

Notation. B is the batch size. k, s, and p denote kernel size, stride, and padding. $\mathrm{GN}(g)$ uses g groups, and $\mathrm{LN}(c)$ normalizes c features. x, y are normalized coordinate maps, and d is the accumulated splat density. $g(S) = [A, c_x, c_y, \sigma_x^2, \sigma_y^2, \sigma_{xy}]$ contains mask area, centroid coordinates, variances, and covariance; Δ denotes target minus source. $f_{\min}, f_{\max}$ are the minimum and maximum selected frame indices. Mask-Conv and Explicit Mask Geometry are parallel branches. Pose bounds are component-wise, and the correction is left-multiplied onto the target-arm extrinsics only.

where $Z_{t,\ell}$ is camera-frame depth in meters and ν_t is derived from the target-mask area. We initialize the selection with one informative frame from each of eight temporal bins, then greedily expand the selected set:

$$t^\star = \arg\max_{t \in \mathcal{P} \setminus \mathcal{F}} \log\det\left(10^{-3} I + \sum_{s \in \mathcal{F}} H_s^{\text{proxy}} + H_t^{\text{proxy}}\right). \tag{15}$$

The ordering is computed once from the 64-frame candidate pool $\mathcal{P}$. Its prefixes define nested subsets $\mathcal{F}_8 \subset \mathcal{F}_{16} \subset \mathcal{F}_{32} \subset \mathcal{F}_{64}$, used in the four successive updates. These link-origin Jacobians serve only as scheduling proxies and are distinct from the SDF residual Jacobians used below.

Reconstruction of sparse geometric factors. For a surface point $X_{\ell j}$ in link-local coordinates, its camera-frame position at stage k is

$$P_{t,\ell j}^{(k)} = C_a^k B_{t,a} F_{t,a,\ell}(q_{t,a}) X_{\ell j}, \qquad u_{t,\ell j}^{(k)} = \pi_K(P_{t,\ell j}^{(k)}), \tag{16}$$

where $F_{t,a,\ell}$ is the link-to-root transformation obtained by forward kinematics. We pre-sample 256 surface points per modeled link. At each stage, points from both arms are projected into a 90×160 point depth buffer, with a 2 mm occlusion tolerance. Visible target-arm points are ranked by the tangency score $1 - |n^\top v|$, where n and v are unit surface-normal and viewing directions. At most 96 contour candidates are retained per frame, with a validity threshold of 0.35. This procedure approximates the projected contour using sparse surface samples.

The target-mask SDF D_t is precomputed with positive values outside the mask, negative values inside, and clipping to $[-40, 40]$ pixels. Bilinear sampling at each projected candidate yields

$$r_i^{(k)} = D_{t(i)}(u_i^{(k)}). \tag{17}$$

The four stages contain at most 768, 1,536, 3,072, and 6,144 candidate factors, respectively; visibility, image-boundary, and tangency checks further reduce the valid set.

Local linearization. For a camera-frame point P_i, the projection Jacobian under a left-multiplicative perturbation is

$$A_i = D\pi_K(P_i)\left[10^{-3} I \quad -\frac{\pi}{180}[P_i]_\times\right]. \tag{18}$$

For a continuous differentiable SDF, the chain rule gives $J_i = \nabla D_{t(i)}(u_i)^\top A_i$. Our implementation instead computes Sobel gradients on the discrete SDF, normalizes nonzero gradient vectors, and bilinearly samples this gradient field at u_i. Denoting the sampled vector by $\widetilde{g}_i$, we use

$$\widetilde{J}_i = \widetilde{g}_i^\top A_i, \qquad r_i(C \oplus d) \approx r_i(C) + \widetilde{J}_i d. \tag{19}$$

Thus, $\widetilde{J}_i$ is a local approximation rather than the exact derivative of the bilinearly sampled, clipped residual.

Learned reliability and anisotropic damping. Each factor is represented by a 32-dimensional descriptor containing residual, gradient, image and 3D coordinates, Jacobian, link and arm identities, time, joint state, and inter-arm depth-margin information. A shared network encodes the factors into h_i and aggregates a global context $\bar{h}$. Reliability and damping are predicted as

$$\alpha_i = 0.02 + 0.98\,\sigma\big(f_\theta([h_i, \bar{h}])\big), \tag{20}$$

$$w_i = m_i\,\alpha_i \frac{1}{1 + (r_i/12)^2}, \tag{21}$$

$$\lambda = \text{softplus}(g_\theta(\bar{h})) + 10^{-3}\mathbf{1}, \tag{22}$$

where m_i is the validity indicator and $\lambda \in \mathbb{R}^6$ contains one damping value per pose parameter. The Cauchy scale is 12 pixels.

Table 8 gives the layer dimensions of the shared learned LM controls used at all four refinement stages. The factor encoder produces one 128-dimensional token per descriptor; only valid tokens contribute to the sequence-level mean.

Table 8: Architecture of the learned robust LM module. Colored bands group factor construction, shared encoding, pooling, and the two prediction heads. N is minibatch size, F_k is the stage-k frame count, and M_k is the number of factor slots.

Layer / Function	Detail	Output
Sparse Factor Descriptor		
Factor slots	Up to 96 contour candidates per selected frame; $M_k = 96F_k$, $F_k \in \{8, 16, 32, 64\}$	$N \times M_k$
SDF and gradient	$r_i/40$ (1), normalized sampled Sobel gradient (2)	$N \times M_k \times 3$
Image and 3D point	Image coordinates mapped to $[-1, 1]$ (2); $P_i/(0.20, 0.20, 0.30)$ (3)	$N \times M_k \times 5$
Jacobian	Approximate residual Jacobian $\widetilde{J}_i/4$	$N \times M_k \times 6$
Identity and motion	Link one-hot (7), arm one-hot (2), normalized time (1), joint state divided by $(\pi, \pi, 0.25, \pi, \pi, \pi, 1)$ (7)	$N \times M_k \times 17$
Depth margin	$\mathrm{clip}((z_{\mathrm{other}} - z_i)/0.02, -1, 1)$; set to 1 if the other arm has no depth	$N \times M_k \times 1$
Concatenation	$3 + 5 + 6 + 17 + 1 = 32$ features; validity mask m_i	$N \times M_k \times 32$
Shared Factor Encoder		
Encoder 1	Linear($32 \to 128$), LN(128) $\to$GELU	$N \times M_k \times 128$
Encoder 2	Linear($128 \to 128$) $\to$GELU	$N \times M_k \times 128$
Global Pooling		
Valid-factor mean	$\bar{h} = \sum_i m_i h_i / \max(\sum_i m_i, 1)$	$N \times 128$
Context broadcast	Repeat $\bar{h}$ across factor slots	$N \times M_k \times 128$
Reliability Head f_θ		
Context fusion	Concatenate $[h_i, \bar{h}]$	$N \times M_k \times 256$
MLP 1	Linear($256 \to 128$), LN(128) $\to$GELU	$N \times M_k \times 128$
MLP 2	Linear($128 \to 64$) $\to$GELU	$N \times M_k \times 64$
Reliability logit	Linear($64 \to 1$)	$N \times M_k \times 1$
Learned reliability	$\alpha_i = 0.02 + 0.98\,\sigma(f_\theta([h_i, \bar{h}]))$	$N \times M_k \times 1$
Effective factor weight	$w_i = m_i \alpha_i / [1 + (r_i/12)^2]$	$N \times M_k \times 1$
Anisotropic Damping Head g_θ		
Damping hidden	Linear($128 \to 64$) $\to$GELU	$N \times 64$
Damping logits	Linear($64 \to 6$)	$N \times 6$
Positive damping	$\lambda = \mathrm{softplus}(g_\theta(\bar{h})) + 10^{-3}\mathbf{1}$	$N \times 6$
Differentiable LM Update		
Normal equations	Weighted $H = \sum_i w_i \widetilde{J}_i^\top \widetilde{J}_i$, $b = \sum_i w_i \widetilde{J}_i^\top r_i$	$N \times 6 \times 6$, $N \times 6$
Damped solve	$(H + \Lambda)d^\star = -b$; Λ is diagonal and scaled by λ	$N \times 6$
Bounded update	$\delta_j = s_j \tanh(d_j^\star / s_j)$; $s = (12, 12, 12, 7, 7, 7)$ in mm/degrees	$N \times 6$

For the current factors, we construct

$$H = \sum_i w_i \widetilde{J}_i^\top \widetilde{J}_i, \qquad b = \sum_i w_i \widetilde{J}_i^\top r_i, \tag{23}$$

$$\Lambda = \mathrm{diag}\big(\lambda \odot \max(\mathrm{diag}(H), 10^{-4}) + 10^{-4}\mathbf{1}\big), \tag{24}$$

where the maximum is element-wise. Holding these quantities fixed, the local update solves

$$d^\star = \arg\min_d \frac{1}{2} \sum_i w_i (r_i + \widetilde{J}_i d)^2 + \frac{1}{2} d^\top \Lambda d, \qquad (H + \Lambda)d^\star = -b. \tag{25}$$

We solve this system using `torch.linalg.solve` without explicitly forming an inverse.

Bounded rigid updates. Before composition, the solution is softly bounded component-wise:

$$\delta_j = s_j \tanh(d_j^\star / s_j), \qquad s = (12, 12, 12, 7, 7, 7)^\top, \tag{26}$$

with translation bounds in millimeters and rotation-vector bounds in degrees. Writing $\delta = (\delta_t, \delta_r)$, we apply

$$C_a^{k+1} = \mathcal{R}(\delta) C_a^k, \qquad \mathcal{R}(\delta) = \begin{bmatrix} \mathrm{Exp}\big([\frac{\pi}{180}\delta_r]_\times\big) & 10^{-3}\delta_t \\ 0^\top & 1 \end{bmatrix}. \tag{27}$$

This retraction directly inserts the translation into the rigid transformation; it is not the full $\mathrm{SE}(3)$ exponential map. The bounded increment is also distinct from the unconstrained minimizer $d^\star$.

Unrolling and gradient flow. Each of the four stages performs one update and then reconstructs projections, visibility, contour candidates, residuals, and Jacobians at the updated pose. Network parameters are shared across stages. During training, discrete frame selection is excluded from backpropagation, and the per-stage geometric descriptors, residuals, and Jacobians are detached. Gradients propagate through the predicted weights and damping, the linear solve, pose composition, and supervision losses. Local geometry is therefore recomputed at every stage but held fixed for differentiation through that stage. The coarse estimate and intermediate refined poses receive pose supervision, while final geometric supervision uses sparse soft-mask IoU and SDF losses, rather than full-mesh evaluation IoU.

B.3 Training Parameters.

For the motion–visibility candidate pool in Sec. 2.3, we use $\beta = 0.15$, $B = 8$ temporal intervals, and $K_c = P = 64$ candidate frames. A frame is initially eligible when the target-arm mask occupies at least 10^{-3} of the image; when fewer than K_c frames satisfy this criterion, remaining frames are added in descending visibility order. During training, uniform random score jitter of magnitude 0.05 is used for candidate selection; evaluation is deterministic. The subsequent D-optimal schedule uses 8 temporal intervals, an information-matrix prior of $10^{-3}I$, and nested frame counts $(8, 16, 32, 64)$.

For the four LM stages ($k = 1, \ldots, 4$), the trajectory weights are $\alpha_k = 0.7^{4-k}$, namely $(0.343, 0.49, 0.7, 1)$. The Smooth L1 inputs use $\sigma_t = 30$ mm and $\sigma_r = 15^\circ$. In the solver loss, $\mu = 0.5$ and $\gamma = 0.01$; with the main-text $\lambda_s = 0.01$, the effective log-determinant coefficient is 10^{-4}. The coarse pose loss is weighted by 2.0, while the final sparse-mask IoU and SDF contour-alignment terms each have unit weight. The configured batch size is 128: batches preserve complete groups of 16 perturbations per calibration task and contain up to eight tasks (with smaller final batches per task group).

Each task's 16 initial perturbations are sampled in translation and rotation-vector coordinates with per-axis bounds of ± 30 mm and $\pm 15^\circ$, respectively. For each of the translation and rotation groups, we draw a three-dimensional standard-normal direction, divide by its ℓ_∞ norm, and multiply by an independently drawn uniform radius in one of $[0.08, 0.34]$, $[0.34, 0.67]$, and $[0.67, 1.0]$. The three radial intervals are cycled across variants before shuffling; the resulting rigid perturbation left-multiplies the ground-truth extrinsic. The perturbation and training seeds are 260909 and 260915, respectively.

C Evaluation metrics

We evaluate synthetic sequences using ground-truth extrinsics and real sequences using observation consistency. Let $\hat{T} = [\hat{R}, \hat{t}]$ and $T^* = [R^*, t^*]$ denote the estimated and ground-truth transformations from the target manipulator root frame to the camera frame. For frame f, let X_{fp} be a URDF surface point obtained through forward kinematics, with camera-frame coordinates $\hat{x}_{fp} = \hat{R}X_{fp} + \hat{t}$ and $x^*_{fp} = R^*X_{fp} + t^*$. All 3D coordinates are expressed in meters.

Rotation and translation errors. Rotation error is the geodesic distance on $\mathrm{SO}(3)$, reported in degrees:

$$E_R = \frac{180}{\pi} \arccos\left[\mathrm{clip}\left(\frac{\mathrm{tr}(\hat{R}R^{*\top}) - 1}{2}, -1, 1\right)\right]. \tag{28}$$

Translation error measures the distance between the estimated and ground-truth manipulator root origins in camera coordinates:

$$E_T = 1000\|\hat{t} - t^*\|_2. \tag{29}$$

It is reported in millimeters and evaluates the final extrinsics, rather than the magnitude of the predicted correction.

2D reprojection error. We measure the mean pixel distance between corresponding URDF points projected under the estimated and ground-truth extrinsics:

$$E_{2D} = \frac{1}{|\mathcal{V}|} \sum_{(f,p)\in\mathcal{V}} \left\| \pi_K(\hat{x}_{fp}) - \pi_K(x^*_{fp}) \right\|_2 , \tag{30}$$

where π_K is perspective projection and $\mathcal{V}$ contains the valid point pairs retained by the evaluator. Camera intrinsics are scaled to the evaluation image resolution.

3D URDF alignment error. We sample 256 surface points per link and compute their mean 3D displacement across all $F = 100$ frames:

$$E_{3D} = \frac{1000}{FP} \sum_{f=1}^{F} \sum_{p=1}^{P} \|\hat{x}_{fp} - x^*_{fp}\|_2, \tag{31}$$

where P is the total number of sampled points per frame. This metric is reported in millimeters and uses known point correspondences, rather than nearest-neighbor matching.

Tool-tip error. The evaluated tip is a geometric proxy constructed from the two jaw meshes. Specifically, endpoint proxies $b_f^{(1)}$ and $b_f^{(2)}$ are derived from their bounding boxes and transformed into the manipulator root frame. Their midpoint is $a_f = (b_f^{(1)} + b_f^{(2)})/2$, yielding

$$E_{\text{tip}} = \frac{1000}{F} \sum_{f=1}^{F} \left\| (\hat{R}a_f + \hat{t}) - (R^* a_f + t^*) \right\|_2 . \tag{32}$$

The error is reported in millimeters. The proxy is derived from mesh geometry rather than a manually annotated tip.

Projection alignment. For real sequences without ground-truth extrinsics, we evaluate the spatial consistency between the projected URDF geometry and the observed instrument mask using intersection-over-union (IoU). For frame f, let $\hat{M}_f$ denote the instrument mask rendered from the URDF under the estimated extrinsics and M_f the observed mask. We compute

$$\text{IoU}_f = \frac{\left|\hat{M}_f \cap M_f\right|}{\left|\hat{M}_f \cup M_f\right|}. \tag{33}$$

The reported IoU is averaged over valid frames and evaluated separately for each manipulator; higher values indicate better URDF–image alignment.

Temporal motion consistency. For real sequences, we further evaluate whether the calibrated URDF motion is consistent with the observed instrument motion in the video. We use CoTracker3 to track points sampled within each target-instrument mask and retain trajectories that are visible and remain inside the corresponding instrument masks in two adjacent frames. For a valid tracked point i between frames f and $f+1$, the observed image-plane displacement is $d_{fi}^{\text{obs}} = x_{f+1,i} - x_{f,i}$. Using the estimated sequence-level extrinsic correction together with the URDF, robot actions, and camera motion, we render the corresponding URDF motion field and bilinearly sample it at $x_{f,i}$ to obtain d_{fi}^{URDF}.

We define the temporal motion consistency error as

$$E_{\text{flow}} = \frac{1}{|\mathcal{Q}|} \sum_{(f,i)\in\mathcal{Q}} \left(\left| d_{fi,x}^{\text{URDF}} - d_{fi,x}^{\text{obs}} \right| + \left| d_{fi,y}^{\text{URDF}} - d_{fi,y}^{\text{obs}} \right| \right), \tag{34}$$

where $\mathcal{Q}$ denotes the set of valid tracked motion samples. The error is first averaged within each video–arm calibration task and then macro-averaged across all valid tasks, preventing longer sequences from receiving larger weights. E_{flow} is reported in pixels at 320×180 resolution; lower values indicate better temporal motion consistency.

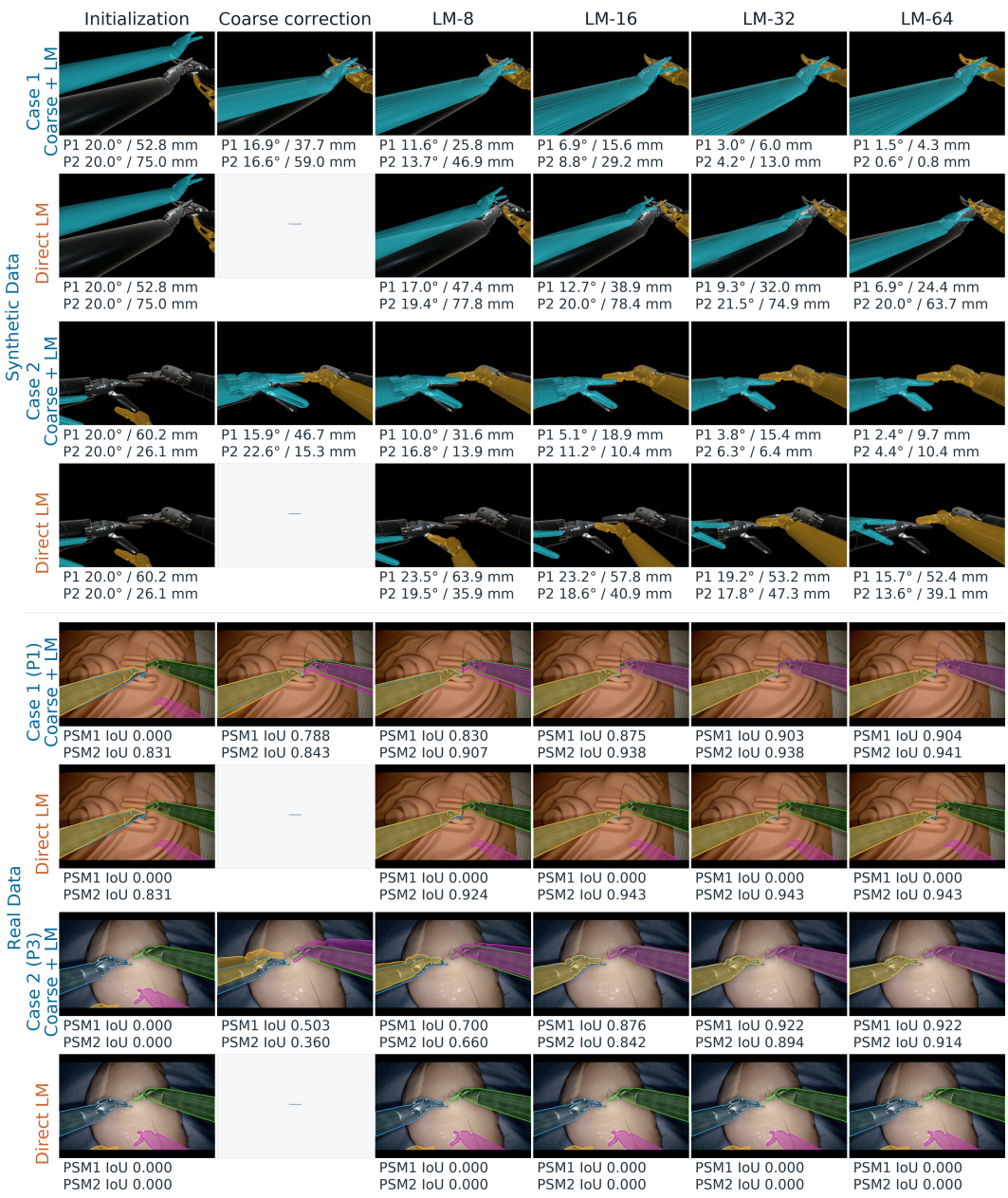


Figure 8: Qualitative convergence comparison between direct LM and the proposed coarse-to-fine optimization on synthetic and real surgical sequences. Columns show the progression from initialization to coarse correction and successive LM refinement stages with 8, 16, 32, and 64 observations.

D Supplementary Results and Visualization

Qualitative Optimization Convergence. Figure 8 visualizes the progressive calibration process on both synthetic and real surgical sequences. Starting from inaccurate initial extrinsics, the proposed coarse correction substantially improves global URDF–image alignment and provides a favorable starting point for subsequent refinement. As the observation budget increases from 8 to 64 frames, the projected instrument geometry progressively approaches the observed instrument boundaries. In contrast, direct LM optimization without coarse initialization exhibits substantially slower and less reliable convergence, particularly under large initial misalignment.

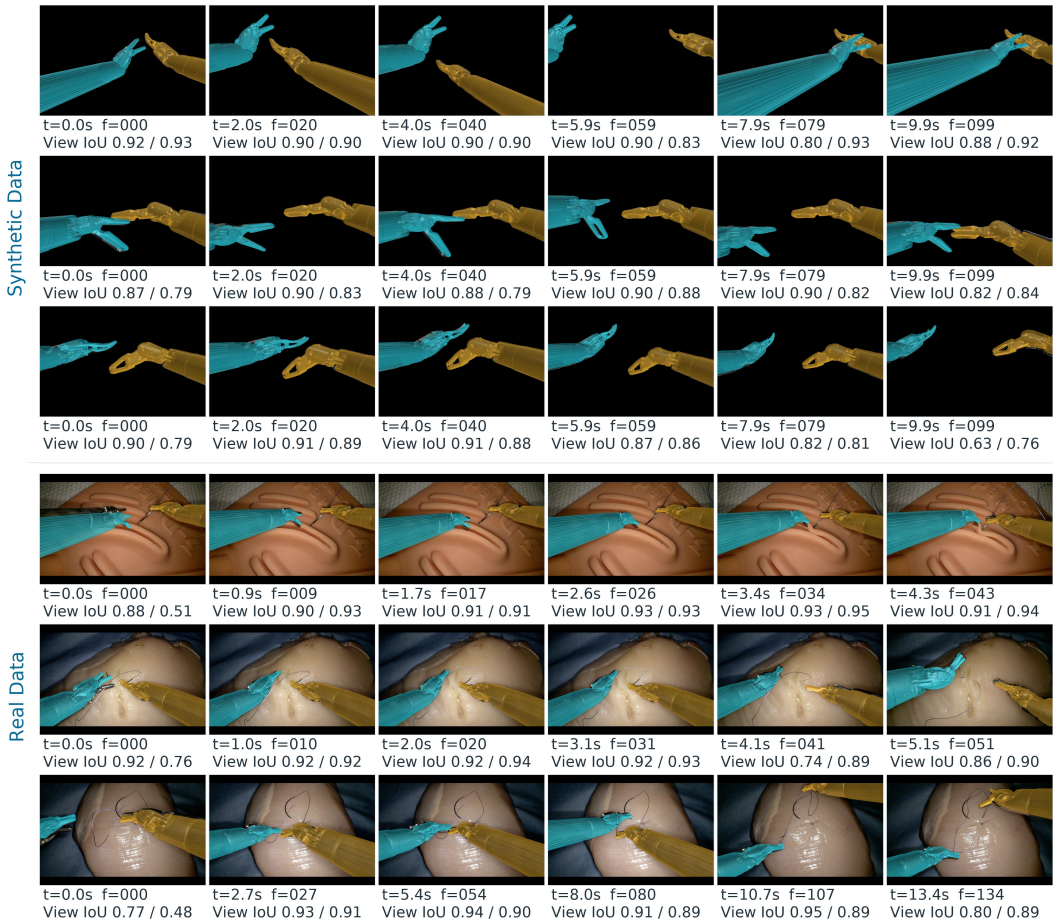


Figure 9: Temporal consistency of URDF projection alignment on synthetic and real surgical sequences. Each row shows representative frames sampled over time, with the estimated extrinsics fixed for the entire sequence.

Temporal Alignment Consistency. Figure 9 shows representative frames sampled throughout synthetic and real surgical sequences using a single sequence-level calibration estimate. Despite substantial changes in instrument configuration and image position, the projected URDF geometry remains consistently aligned with the observed instruments over time. This behavior is observed for both manipulators and across different surgical environments, indicating that the estimated extrinsics capture a stable camera–robot relationship rather than fitting only a small subset of frames.

Robustness to Initialization Perturbations. Figure 10 provides qualitative comparisons under increasingly challenging initialization errors ranging from $5°/5$ mm to $20°/30$ mm. While several baseline methods show increasing projection misalignment as the perturbation magnitude grows, SurgURDF-Calib maintains more stable URDF–image correspondence across the evaluated conditions. The difference becomes more evident under larger perturbations, where reliable coarse initialization and subsequent geometric refinement help recover from substantial initial misalignment.

Coarse-to-Fine Refinement under Different Initializations. Figure 11 further illustrates how SurgURDF-Calib recovers from different levels of initial calibration error. Across all perturbation settings, the coarse module first removes a substantial portion of the global misalignment, after which the nested LM stages progressively refine local geometric correspondence. The improvement becomes especially important for larger perturbations, where direct local refinement would otherwise start far from a favorable convergence basin. These examples visually support the complementary roles of coarse initialization and progressive geometric optimization.

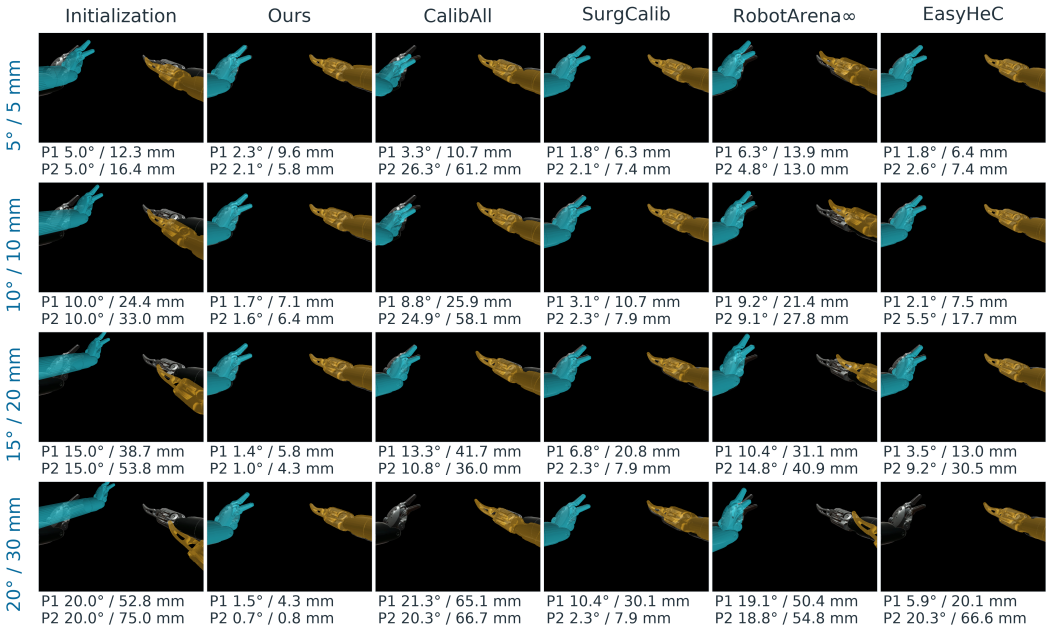


Figure 10: Qualitative robustness comparison under increasing initial extrinsic perturbations. Rows correspond to progressively larger rotation/translation perturbations, while columns compare SurgURDF-Calib with representative calibration baselines.

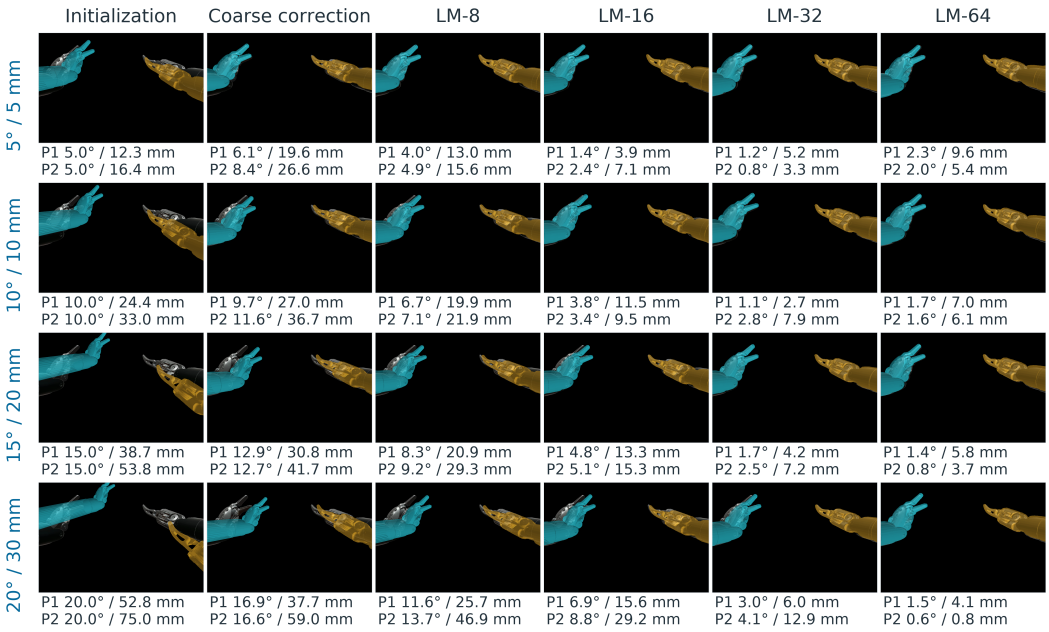


Figure 11: Qualitative convergence of SurgURDF-Calib under different initialization perturbations. Across increasingly challenging initial errors, coarse correction first reduces global misalignment, followed by progressive LM refinement with 8, 16, 32, and 64 selected observations.