CALIB-X: TRAINING-FREE CAMERA-TO-ROBOT POSE ESTIMATION VIA DENSE FEATURE MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Camera-to-robot pose estimation is a fundamental yet challenging problem for vision-based robotic manipulation, requiring accurate recovery of the rigid transformation between a camera and a robot. Conventional calibration approaches often rely on fiducial markers or manual procedures, limiting scalability in real-world deployments. Recent learning-based methods alleviate this burden by directly predicting poses from images, but they typically require large-scale pose annotations and costly retraining when transferring to new robots or environments. In this paper, we introduce **Calib-X**, a **training-free** framework that reformulates camera-to-robot calibration as an **end-to-end dense feature matching** problem. Leveraging robot kinematics, Calib-X first generates multi-view robot renderings from predefined viewpoints and establishes dense correspondences between rendered templates and an uncalibrated camera image using semantic features from the foundation model DINOv3. These correspondences enable direct estimation of the initial camera pose without any task-specific optimization or data collection. Furthermore, we propose an iterative pose refinement strategy that continuously re-renders the robot under the updated pose and performs dense feature alignment with the query image, progressively improving calibration accuracy. Extensive experiments on four public benchmarks demonstrate that Calib-X achieves accurate and robust camera-to-robot pose estimation across diverse robotic platforms and environments. Without requiring markers, annotations, or retraining, Calib-X provides a scalable and generalizable solution for real-world robot calibration. *The code will be released publicly upon publication.*

1 INTRODUCTION

Camera-to-robot pose estimation aims to recover the rigid transformation between a camera and a robot coordinate system. It is a fundamental requirement for vision-based robotic manipulation, as visual observations must be accurately transformed into the robot frame for grasping, motion planning, and visual servoing. However, camera poses may change after sensor replacement, robot relocation, or workspace reconfiguration, making efficient and reliable recalibration necessary in practical deployments.

Traditional calibration methods usually rely on fiducial markers or specially designed calibration targets Garrido-Jurado et al. (2014); Olson (2011); Fiala (2005). Although accurate, they require manual target placement, repeated data collection, and sufficient target visibility, which interrupts normal robot operation and limits scalability. To reduce manual effort, recent marker-free methods directly estimate the camera pose from robot images by detecting robot keypoints, aligning silhouettes, or regressing poses with neural networks Lee et al. (2020); Lu et al. (2023); Labbé et al. (2021); Chen et al. (2023). However, learning-based methods often depend on large-scale pose annotations and robot-specific training. Their generalization may also degrade when applied to new robot structures, camera viewpoints, or environments, requiring additional data collection and retraining. Practical manipulation data further complicates this setting: camera placement is often dictated by the task, and robot arms may be only partially observed in large-scale datasets such as Open X-Embodiment and DROID Lu et al. (2025); Padalkar et al. (2023); Khazatsky et al. (2024).

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 this work, we revisit camera-to-robot pose estimation as a dense correspondence problem. Given the robot model, kinematics, and current joint configuration, the robot can be rendered under arbitrary camera viewpoints. Each pixel on the rendered robot can further be associated with a known 3D point. Therefore, dense matches between a rendered image and an uncalibrated query image naturally provide 2D-to-3D correspondences for geometric camera pose estimation. However, rendered and real robot images usually differ significantly in texture, lighting, background, and appearance. Their viewpoints can also be substantially different, making conventional feature matching unreliable. Moreover, matching against a finite set of rendered templates can only provide a discrete pose hypothesis, which is insufficient for accurate camera pose estimation.

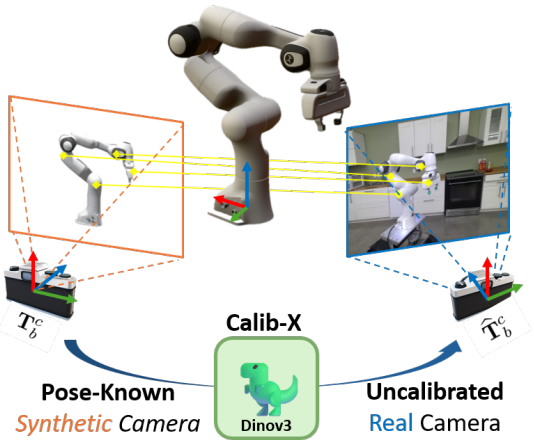


Figure 1: Overview of Calib-X. Given a pose-known synthetic camera and an uncalibrated real camera, Calib-X establishes dense correspondences between the rendered robot and the real-world observation using semantic features from the foundation model. These cross-domain correspondences bridge the known robotic geometry and the real-world image, enabling training-free camera-to-robot pose estimation.

To address these challenges, we propose **Calib-X**, a training-free framework that formulates camera-to-robot calibration as iterative rendered-to-real dense matching. Calib-X first renders the robot from a set of predefined template viewpoints according to its current kinematic configuration. It then uses the pretrained RoMaV2 matcher Edstedt et al. (2025) to establish dense correspondences between each rendered template and the real query image. Based on the predicted matches and confidence scores, reliable 2D-to-3D correspondences are selected to estimate an initial camera pose. Since the initial estimate is constrained by the discrete template viewpoints, we further introduce an iterative render-and-match refinement strategy. Starting from the current pose estimate, Calib-X re-renders the robot from the estimated viewpoint and matches the re-rendered image with the query image. As the viewpoint discrepancy becomes smaller, the predicted correspondences become more accurate and geometrically consistent. The camera pose is then updated using the new correspondences. Repeating this process progressively refines the pose from a coarse template-based estimate to an accurate continuous solution.

Our main contributions are summarized as follows:

- We propose **Calib-X**, a training-free camera-to-robot pose estimation framework based on dense matching between rendered robot views and real observations.
- We construct reliable sim-to-real 3D-to-2D correspondences without calibration annotations or robot-specific fine-tuning.
- We introduce an iterative render-and-match refinement strategy that progressively reduces viewpoint discrepancies and improves pose accuracy.
- Extensive experiments on three public benchmarks demonstrate the effectiveness and generalization ability of Calib-X.

2 Related Work

2.1 Camera-to-Robot Pose Estimation

Traditional camera-to-robot calibration uses fiducial markers or calibration boards to estimate the rigid camera-to-robot transformation Garrido-Jurado et al. (2014); Olson (2011); Fiala (2005). For articulated robots, kinematics associates marker observations with 3D points, enabling geometric calibration Horaud & Dornaika (1995); Ilonen & Kyrki (2011). Although accurate, these methods require careful setup and inconvenient recalibration after camera or robot changes.

Markerless methods instead use the robot as the calibration target. Early systems localized articulated robots from depth or object-coordinate predictions Schmidt et al. (2014); Michel et al. (2015), while recent RGB methods are mainly keypoint- or rendering-based. Keypoint methods detect predefined landmarks and solve for the camera pose, often using synthetic data and sim-to-real transfer Lambrecht & Kästner (2019); Zuo et al. (2019); Lu et al. (2022). DREAM Lee et al. (2020) predicts single-image keypoints for PnP. CtRNet Lu et al. (2023) uses rendering-supervised keypoint prediction, while CtRNet-X Lu et al. (2025) selects keypoints from visible components to handle partial views; both require robot-specific training or adaptation. Rendering methods compare observations with robot renderings. Earlier work estimated robotic-tool poses in this way Hao et al. (2018), while RoboPose Labbé et al. (2021) jointly estimates robot pose and joint angles via render-and-compare. EasyHeC Chen et al. (2023) aligns rendered and observed masks and selects informative configurations. These methods avoid markers but often rely on trained perception models or multiple observations.

Recent studies have explored training-free calibration using pretrained vision models and robot kinematics. EasyHeC++ Hong et al. (2024) combines automatic segmentation with differentiable rendering; ARC-Calib Chanrungmaneekul et al. (2025) uses visual trajectories from exploratory motions; Kalib Tang et al. (2025) tracks a predefined point across frames for kinematics-based PnP; and CalibAll Xie et al. (2025) combines an annotated end-effector point with tracking, temporal PnP, and mask refinement. In contrast, Calib-X requires neither manual reference points, exploratory motion, nor multi-frame tracking. It obtains dense correspondences from rendered templates and a single real image, then iteratively re-renders and updates the pose using new 2D-to-3D correspondences.

2.2 Feature Matching for Pose Estimation

Feature matching aims to find corresponding visual elements across images and provides important geometric constraints for pose estimation. It is widely used in visual SLAM and camera localization to recover camera motion from matches between different views. In 6D object pose estimation, image correspondences are further associated with known 3D object points, allowing the object pose to be recovered using PnP solver. Traditional methods mainly rely on sparse keypoints and local descriptors, such as SIFT Lowe (2004), while recent methods predict dense correspondences using pretrained visual representations. These advances have also enabled training-free object pose estimation. FoundPose Örnek et al. (2024) matches DINOv2 Oquab et al. (2024) descriptors between real images and rendered templates, while FreeZe Caraffa et al. (2024) combines image and 3D point features for RGB-D pose estimation. Pos3R Deng et al. (2025) further leverages the 3D-aware features of MASt3R Leroy et al. (2024) for template selection and dense correspondence estimation, enabling unseen object pose estimation from a single RGB image without task-specific training.

Calib-X extends dense feature matching to camera-to-robot pose estimation. Given the current robot configuration, Calib-X leverages dense features from the foundation model DINOv3 Siméoni et al. (2025) to match rendered robot views with a real observation. The resulting matches are associated with the known robot geometry to construct 3D-to-2D correspondences. Furthermore, the camera pose is further refined through iterative re-rendering and feature matching.

3 Methodology

Given a single RGB observation of a robot, Calib-X estimates the camera pose with respect to the robot base without any training process. Firstly, it leverages the robot's known joint state to render the articulated model from a fixed set of viewpoints. Then Calib-X these synthetic templates against the query image to a coarse initial pose. Finally, the robot is re-rendered from the estimated viewpoint, and the pose is refined through iterative matching and pose estimation.

3.1 Problem Formulation

Given an RGB image $\mathbf{I}$, intrinsic matrix $\mathbf{K}$, robot model $\mathcal{G}$, and joint configuration $\mathbf{q}$, we aim to estimate the 6-DoF camera pose $\mathbf{T}_{R \leftarrow C} = (\mathbf{R}, \mathbf{t}) \in SE(3)$, where $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$ transform points from the camera frame to the robot-base frame. The joint state $\mathbf{q}$ defines the robotic articulated geometry via forward kinematics. Calib-X leverages the known geometry to render the robot from multiple viewpoints and establish dense pixel-wise correspondences with the

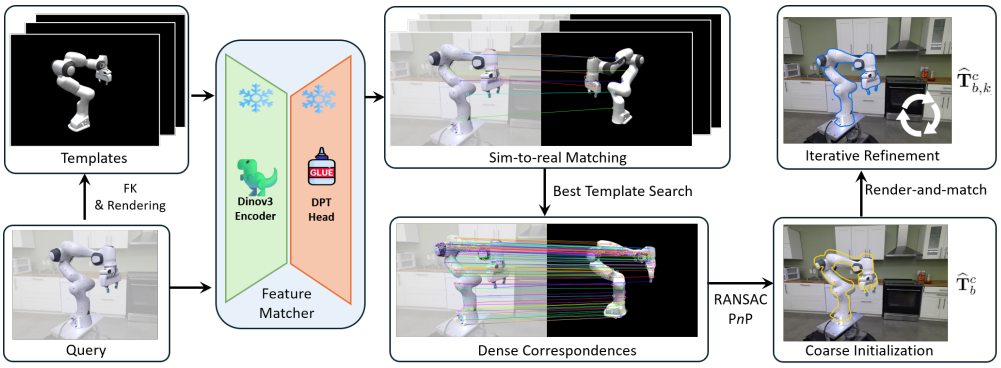


Figure 2: Calib-X pipeline. Calib-X renders pose-known robot templates, establishes dense correspondences with the uncalibrated query image using frozen DINOv3 features, and progressively refines the estimated camera-to-robot pose.

input image. These correspondences are converted into 3D-to-2D matches, from which the camera pose is directly estimated. Notably, our method requires only a single RGB observation and operates without fiducial markers, ground-truth pose annotations, or specific training.

3.2 Training-free Pipeline

Overview. Following the standard pipeline for camera-to-robot extrinsic calibration, Calib-X is decomposed into two sequential stages: coarse pose estimation and pose refinement. Each stage is kept frozen throughout the calibration process, eliminating the need for robot-specific training.

Coarse Initialization. To obtain a coarse calibration, previous methods usually rely on either keypoint-based or rendering-based approaches, which often require task-specific training or time-consuming optimization. When the robot state is known, however, the robot can be rendered from arbitrary viewpoints and used as a reference. Together with foundation models that encode semantic and geometric priors learned from large-scale data, camera-to-robot pose estimation can be reformulated as a relative pose estimation problem.

Given the current robot state, we first update the robot model through forward kinematics and render robot from T virtual cameras around the robot on an orbit centered at the robot. The cameras share a fixed elevation and viewing distance, while their azimuth angles are uniformly sampled as

$$\theta_t = \theta_0 + \frac{2\pi t}{T}, \quad t = 0, \dots, T-1, \tag{1}$$

where θ_0 denotes the initial azimuth. The viewing distance is adjusted according to the spatial extent of the current robot configuration, and each camera is oriented toward the robot center. This produces a set of reference views $\mathcal{I}_{\text{ref}} = \{\mathbf{I}^t_{\text{ref}}\}_{t=1}^{T}$, together with their corresponding camera poses and visible robot geometry. Then we suppress the background of the observed image before matching correspondence between the reference images and observation, since complex background regions may introduce incorrect correspondences. Specifically, we use the Segment Anything Model Kirillov et al. (2023) to obtain a robot mask $\mathbf{M}_o$ and isolate the visible robot region as $\mathbf{I}_o = \mathbf{M}_o \odot \mathbf{I}$, where $\odot$ denotes element-wise multiplication.

Given the masked observation $\mathbf{I}_o$ and a rendered reference view $\mathbf{I}^t_{\text{ref}}$, we use the pretrained RoMaV2 as our dense image matcher. RoMaV2 first extracts frozen DINOv3 features from both images. These features provide robust semantic and structural cues, which help match the rendered robot with its real observation despite their differences in appearance. The features from the two images are jointly processed by a multi-view transformer to predict a coarse dense warp and its matching confidence. The coarse prediction is then progressively updated by convolutional refiners to obtain fine-grained correspondences at the original image resolution. The matching process can be represented as

$$(\mathbf{W}_t, \mathbf{P}_t) = \mathcal{F}_{\text{match}}\left(\mathbf{I}^t_{\text{ref}}, \mathbf{I}_o\right), \tag{2}$$

where $\mathbf{W}_t$ is the dense warp from the rendered reference to the observed image, and $\mathbf{P}_t$ is the corresponding confidence map. The resulting matches are initially 2D-to-2D correspondences. To recover their 3D positions, we additionally render a position map $\mathbf{G}_t$ for each reference view, where every valid pixel stores the corresponding robot surface point in the robot-base coordinate system. For a matched reference pixel $\mathbf{u}_j^{\mathrm{ref}}$, its 3D position and matched observation are given by

$$\mathbf{X}_j^R = \mathbf{G}_t\left(\mathbf{u}_j^{\mathrm{ref}}\right), \qquad \mathbf{u}_j^{\mathrm{obs}} = \mathbf{W}_t\left(\mathbf{u}_j^{\mathrm{ref}}\right). \tag{3}$$

Each valid 2D-to-2D match is then lifted to a 3D-to-2D correspondence using the position map of the rendered view. Based on these correspondences, we estimate one camera pose for each reference view using PnP-RANSAC. The resulting pose candidates are evaluated by the number of PnP inliers and their reprojection errors, and the most reliable candidate is selected as the coarse camera pose $\widehat{\mathbf{T}}_b^c$.

Pose Refinement. The coarse pose is selected from a discrete set of reference viewpoints and may still contain errors when the real camera viewpoint is not well covered by the rendered templates. We therefore introduce an iterative render-and-match process to further refine the pose.

Starting from the coarse pose $\widehat{\mathbf{T}}_b^c$, we re-render the robot from the current estimated viewpoint:

$$\mathbf{I}_{\mathrm{coarse}}^k = \mathcal{R}\left(\mathcal{G}(\mathbf{q}), \mathbf{K}, \widehat{\mathbf{T}}_{b,k}^c\right), \tag{4}$$

where $\widehat{\mathbf{T}}_{b,k}^c$ denotes the pose estimate at the k-th iteration. The re-rendered image is matched with the observed image, and the resulting matches are converted into new 3D-to-2D correspondences to update the camera pose. As the estimated viewpoint gradually approaches the real viewpoint, the difference between the rendered and observed images becomes smaller, leading to more reliable correspondences. This render-and-match process is repeated until the pose change becomes sufficiently small or the maximum number of iterations is reached.

4 Experiments

4.1 Datasets

We evaluate Calib-X on four public robot datasets covering different camera sensors, robot platforms, environments, and levels of robot visibility.

DREAM-real Dataset. The DREAM-real dataset Lee et al. (2020) contains two subsets, Panda 3CAM and Panda ORB. Panda 3CAM includes approximately 17K images of a Franka Emika Panda robot at a resolution of 640×480. The images are captured using three different cameras: Azure Kinect (AK), Xbox 360 Kinect (XK), and Intel RealSense (RS). Panda ORB contains approximately 40K image frames of the Panda robot under diverse joint configurations and camera viewpoints.

Panda Manipulation Dataset. The Panda Manipulation Dataset Lu et al. (2025) contains 60 video episodes, including 30 robot-in-view episodes and 30 robot-in-and-out-of-view episodes. It represents practical manipulation scenarios in which the visibility of different robot parts changes throughout an episode.

DROID Dataset. The DROID dataset Khazatsky et al. (2024) contains real-world manipulation data collected across different laboratories, with complex backgrounds, diverse workspaces, and real-world viewpoints.

RoboChallenge Table30v2 Dataset. The RoboChallenge Table30v2 dataset Ma et al. (2026) includes multiple robot platforms such as the ARX5 and UR5 arms, which is employed to assess the generalization capability of Calib-X over diverse robotic embodiments.

4.2 Evaluation Metrics

We evaluate the calibration accuracy using the average distance (ADD) metric:

$$\mathrm{ADD} = \frac{1}{n}\sum_{i=1}^{n}\left\|\widetilde{\mathbf{T}}_b^c\mathbf{p}_i - \mathbf{T}_b^c\mathbf{p}_i\right\|_2, \tag{5}$$



Figure 3: Qualitative comparison on the DROID dataset. From top to bottom, we show the input RGB images and the robot renderings projected using the calibration result provided by DROID dataset, CtRNet, and Calib-X, respectively. Calib-X produces more accurate alignment between the rendered robot model and the observed robot across complex background and partial robot visibility.

Method	Type	Training-free	Panda 3CAM-AK		Panda 3CAM-XK		Panda 3CAM-RS		Panda ORB		All	
			AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓
DREAM-F	Keypoint	✘	68.912	11.413	24.359	491.911	76.130	2.077	61.930	95.319	60.740	113.029
DREAM-Q	Keypoint	✘	52.382	78.089	37.471	54.178	77.984	0.027	57.087	67.248	56.988	59.284
DREAM-H	Keypoint	✘	60.520	0.056	64.005	7.382	78.825	0.024	69.054	25.685	68.584	17.477
RoboPose	Rendering	✘	76.497	0.024	85.926	**0.014**	76.863	0.023	80.504	0.019	80.094	0.020
CtRNet	Keypoint	✘	**89.928**	**0.013**	79.465	0.032	**90.789**	**0.010**	85.289	0.021	85.962	0.020
CtRNet-X	Keypoint	✘	–	–	–	–	–	–	–	–	86.231	0.014
Calib-X	Keypoint	✓	85.224	0.015	**86.398**	**0.014**	87.953	0.012	**88.350**	**0.012**	**87.704**	**0.012**

Table 1: Comparison with state-of-the-art methods on DREAM-real using the ADD metric. We report the AUC (%) and mean ADD error (m) for each subset and the complete dataset.

where $\widetilde{\mathbf{T}}_b^c$ and $\mathbf{T}_b^c$ denote the estimated and ground-truth transformations from the robot base frame to the camera frame, respectively. Here, $\mathbf{p}_i$ denotes the homogeneous coordinate of the i-th point sampled from the robot model, and n is the total number of sampled points.

We further report the area-under-the-curve (AUC), computed as

$$\text{AUC} = \frac{1}{KN} \sum_{k=1}^{K} \sum_{j=1}^{N} \mathbb{I}\left(d_j < \tau_k\right), \tag{6}$$

where d_j denotes the ADD error of the j-th sample, and $\{\tau_k\}_{k=1}^{K}$ are K uniformly sampled thresholds within $[0, \tau_{\max}]$. A higher AUC indicates more accurate calibration.

4.3 Comparison with State-of-the-art

Evaluation on DREAM-real Dataset. We compare Calib-X with several state-of-the-art camera-to-robot pose estimation methods, including DREAM, RoboPose, CtRNet, and CtRNet-X. The results of these methods are taken from CtRNet-X Lu et al. (2025). As shown in Table 1, Calib-X achieves the best overall performance on the four scenes of the DREAM-real dataset, reaching the highest ADD AUC and the lowest mean ADD error. Existing methods mainly follow either

Method	Refinement	Steps	Time (s)	DREAM-real		Panda: Robot in View		Panda: Robot In-and-Out	
				AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓
Calib-X	Differentiable Rendering	3,000	120	**88.347**	**0.012**	83.106	**0.019**	65.404	0.111
	Iterative Feature Matching	0	/	85.878	0.014	80.293	0.022	61.279	0.115
		1	2	87.557	0.012	84.522	0.020	66.810	0.094
		2	4	87.683	0.012	85.285	0.020	68.808	**0.091**
		3	6	87.704	**0.012**	**85.379**	0.020	**69.977**	**0.091**

Table 2: Comparison of pose refinement strategies on DREAM-real and Panda Manipulation. Differentiable rendering optimizes the pose for 3,000 steps and requires approximately 120 seconds, while iterative feature matching progressively refines the initial pose, with each iteration requiring approximately 2 seconds. Best results are in bold.

Method	Robot in View		Robot In-and-Out	
	AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓
CtRNet	16.764	0.381	35.944	0.335
CtRNet-X	60.317	0.059	59.828	0.056
CtRNet-X†	70.817	0.038	79.665	0.022
Calib-X	85.379	0.020	69.977	0.091
Calib-X†	**88.541**	**0.011**	**83.488**	**0.016**

Table 3: Quantitative comparison on the Panda Manipulation Dataset. We report the AUC (%) and mean ADD error (m) under the robot-in-view and robot-in-and-out settings. Methods marked with † use batch estimation over multiple frames, while the remaining results are obtained using single-frame estimation.

keypoint-based or rendering-based pipelines and require robot-specific training. In contrast, Calib-X is training-free and combines semantic cues from a foundation model with the known robot geometry through a render-and-match pipeline. Without any task-specific training, Calib-X achieves an overall AUC of 87.704, outperforming the best DREAM variant and CtRNet-X by 26.964 and 1.473, respectively. It also obtains the lowest mean ADD error of 0.012 m, while remaining competitive across different camera settings.

Evaluation on Panda Manipulation Dataset. The DREAM-real dataset only includes images in which the robot arm is fully visible. To better evaluate performance in practical manipulation scenarios, we further compare Calib-X with existing methods on the Panda Manipulation Dataset. This dataset contains more challenging cases where only part of the robot is visible and the robot may move partially outside the image frame. As shown in Table 3, Calib-X clearly outperforms CtRNet and CtRNet-X under single-frame estimation. In the robot-in-view setting, Calib-X achieves an AUC of 85.379 with a mean ADD error of 0.020 m. Under the more challenging robot-in-and-out setting, it still obtains an AUC of 69.977, showing its robustness to partial observations. When multiple frames are jointly considered, Calib-X further improves the AUC to 88.541 and 83.488 in the two settings, respectively, while reducing the mean ADD error to 0.011 m and 0.016 m. Furthermore, these results also show that Calib-X is not limited to cases where the whole robotic arm is observed. Calib-X provides reliable calibration under both full and partial robot visibility, and multi-frame estimation can further improves robustness.

4.4 Qualitative Results

Calibrated Results. To further validate the calibration results, we present qualitative comparisons on representative scenes from the DROID dataset in Figure 3. We observe that the robot model projected using the provided calibration from DROID dataset often shows noticeable misalignment with the observation, although the original extrinsics were obtained using calibration boards during data collection. CtRNet provides reasonable alignment when most of the robot is visible and the camera viewpoint is relatively regular. However, as shown in the third row of Figure 3, its performance de-

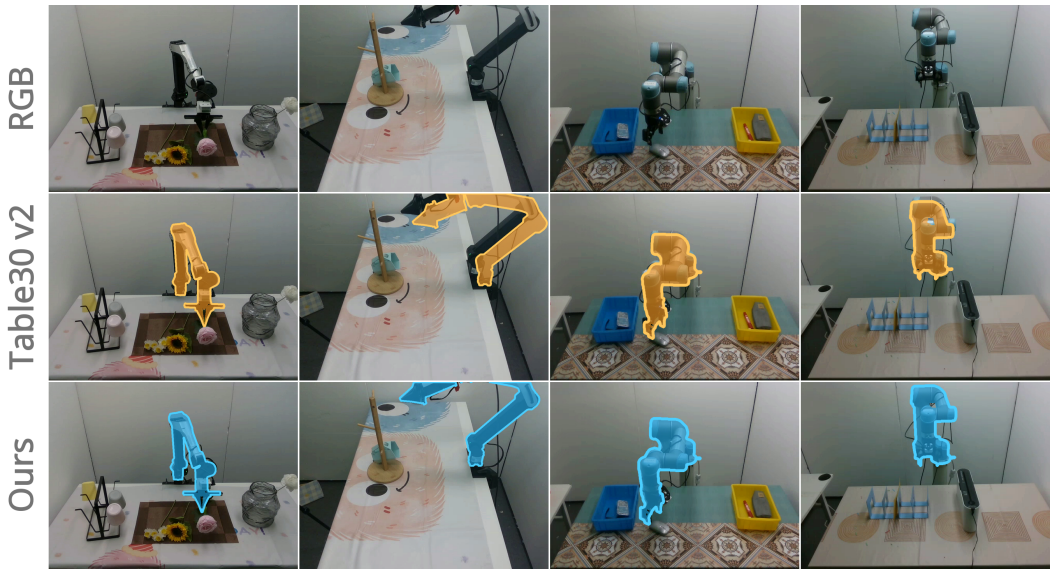


Figure 4: Generalization results on the RoboChallenge Table30v2 dataset. From top to bottom, we show the input RGB images, the robot models projected using the dataset-provided calibration, and the results of Calib-X. Calib-X achieves more accurate alignment on both ARX5 and UR5 under different robot configurations and camera viewpoints.

Variant	Robot in View		Robot In-and-Out	
	AUC ↑	Mean (m) ↓	AUC ↑	Mean (m) ↓
w/o SAM	83.374	0.037	65.285	0.124
with SAM	**85.559**	**0.016**	**66.415**	**0.097**

Table 4: Foreground-masking ablation on the Panda Manipulation Dataset. Each setting contains 900 frames (30 per episode), and both variants use three refinement iterations.

Matcher	AUC ↑	Mean (m) ↓
LightGlue	69.763	0.033
MASt3R	84.658	0.015
RoMa v1	83.354	0.017
RoMa v2	**86.145**	**0.014**

Table 5: Image-matcher ablation on 300 randomly sampled images from the Panda ORB subset of DREAM-real.

grades significantly when only a small part of the robot is visible or when the viewpoint is unusual, often resulting in large pose errors and incorrect alignment. In contrast, Calib-X consistently aligns the rendered model with the visible robot structure. As shown in the fourth row, our method remains effective even when only a limited part of the robot is visible. It also achieves accurate alignment in cluttered environments and under large viewpoint variations, as illustrated by the middle examples. These results further demonstrate that Calib-X can reliably estimate camera-to-robot poses beyond controlled settings where the robot is fully visible.

Coarse Initialization and Pose Refinement. We further visualize the two stages of Calib-X in Figure 5. Despite the large viewpoint differences and complex backgrounds, the coarse estimation stage can already locate the camera from the predefined reference views and provide a reliable initial pose. Starting from this initialization, the refinement stage re-renders the robot from the current estimate and establishes new correspondences with a smaller viewpoint gap. As shown in the last row, only a few iterations are required to further align the projected robot model with the observed robot. These examples show that feature matching is effective for both coarse camera pose initialization and fine-grained calibration.

4.5 Ablation Studies

Effect of Iterative Pose Refinement. We further investigate whether feature matching can be utilized not only for coarse pose initialization but also for fine-grained pose refinement. As shown in Table 2, the coarse estimate without refinement already yields competitive performance across

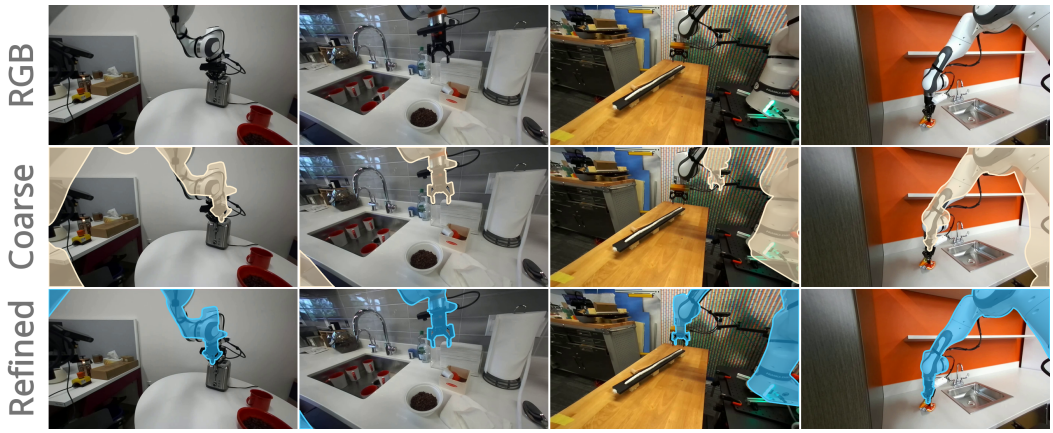


Figure 5: Visualization of the coarse initialization and pose refinement on random DROID scenes. The first row shows the input RGB images, while the second and third rows show the robot model projected using the coarse and refined camera poses, respectively. The coarse stage already provides a reasonable initialization, and the refinement stage further improves the alignment within a few iterations.

all three settings. We observe that even a single iteration of feature matching improves calibration accuracy, and additional iterations provide further gains. With three refinement iterations, Calib-X achieves results comparable to differentiable rendering on DREAM-real, reaching an AUC of 87.70 with a mean ADD error of 0.012 m. It also outperforms the baseline on both Panda Manipulation settings, especially in the more challenging robot-in-and-out scenario, where the AUC improves from 65.40 to 69.98 and the mean error decreases from 0.111 m to 0.091 m. Furthermore, iterative pose refinement is considerably more computationally efficient than differentiable rendering.

Effect of Foreground Masking. As shown in Table 4, SAM-based foreground masking consistently improves calibration accuracy in both settings by suppressing background interference, increasing AUC and reducing mean ADD error.

Effect of Image Matcher. As shown in Table 5, RoMa v2 achieves the best performance among all matchers, confirming that dense feature matching is more suitable than sparse local-feature matching for real-to-render calibration.

Generalization to other Robot Platforms. We further evaluate Calib-X on the RoboChallenge Table30v2 dataset using two additional robot platforms, ARX5 and UR5. As shown in Figure 4, the dataset-provided calibration in the second row shows noticeable misalignment under different robot configurations and camera viewpoints. In contrast, Calib-X produces more accurate alignment for both robot platforms, including side-view and near-frontal observations. Neither robot is used in the previous experiments, demonstrating that Calib-X can be directly applied to different robot appearances and kinematic structures without retraining.

5 Conclusion

In this work, we presented Calib-X, a training-free framework for camera-to-robot pose estimation. By leveraging a pretrained foundation model for sim-to-real feature matching, Calib-X estimates the camera pose from a single RGB image without requiring fiducial markers, pose annotations, or robot-specific retraining. The proposed pipeline provides both a reliable coarse initialization and an efficient iterative refinement, achieving accurate calibration across different viewpoints, partial robot observations, complex environments, and unseen robot platforms. Beyond improving calibration accuracy, Calib-X demonstrates that pretrained visual correspondence models can effectively bridge the gap between real-world observations of known robotic geometry and their simulated digital twin. This offers a practical alternative to calibration pipelines that rely on manual setup or task-specific training, and suggests a more general direction for adapting robotic perception systems across cameras, environments, and embodiments.

AI USE STATEMENT

In preparing this manuscript, we used generative AI tools to assist with language editing and LaTeX formatting, including the layout of figures and tables. The authors reviewed and revised all AI-assisted changes and verified the technical claims, numerical results, and citations against the underlying experiments and source materials. We take full responsibility for the final content of this paper.

REFERENCES

Andrea Caraffa, Davide Boscaini, Amir Hamza, and Fabio Poiesi. Freeze: Training-free zero-shot 6d pose estimation with geometric and vision foundation models. In *European Conference on Computer Vision*, pp. 414–431. Springer, 2024.

Podshara Chanrungmaneekul, Yiting Chen, Joshua T Grace, Aaron M Dollar, and Kaiyu Hang. Arccalib: Autonomous markerless camera-to-robot calibration via exploratory robot motions. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 17933–17940. IEEE, 2025.

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.

Weijian Deng, Dylan Campbell, Chunyi Sun, Jiahao Zhang, Shubham Kanitkar, Matt E Shaffer, and Stephen Gould. Pos3r: 6d pose estimation for unseen objects made easy. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16818–16828, 2025.

J Edstedt, D Nordström, Y Zhang, G Bökman, J Astermark, V Larsson, A Heyden, F Kahl, M Wadenbäck, and M Felsberg. Roma v2: Harder better faster denser feature matching. *arXiv preprint arXiv:2511.15706*, 2025.

Mark Fiala. ARTag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pp. 590–596. IEEE, 2005.

Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco J. Madrid-Cuevas, and Manuel J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

Ran Hao, Orhan Özgüner, and M. Cenk Çavuşoğlu. Vision-based surgical tool pose estimation for the da Vinci robotic surgical system. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1298–1305. IEEE, 2018.

Zhengdong Hong, Kangfu Zheng, and Linghao Chen. Easyhec++: Fully automatic hand-eye calibration with pretrained image models. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 816–823. IEEE, 2024.

Radu Horaud and Fadi Dornaika. Hand-eye calibration. *The International Journal of Robotics Research*, 14(3):195–210, 1995.

Jarmo Ilonen and Ville Kyrki. Robust robot-camera calibration. In *2011 15th International Conference on Advanced Robotics*, pp. 67–74. IEEE, 2011.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

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

Jens Lambrecht and Linh Kästner. Towards the usage of synthetic data for marker-less pose estimation of articulated robots in RGB images. In *2019 19th International Conference on Advanced Robotics*, pp. 240–247. IEEE, 2019.

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.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pp. 71–91. Springer, 2024.

David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

Jingpei Lu, Florian Richter, and Michael C. Yip. Pose estimation for robot manipulators via keypoint optimization and sim-to-real transfer. *IEEE Robotics and Automation Letters*, 7(2):4622–4629, 2022.

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

Yunchao Ma, Ze Chen, Erjin Zhou, Ziming Liu, Haowei Zhang, Kai Liu, and Haoqiang Fan. Table30 v2: Evaluating generalized models by real robots at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4461–4467, 2026.

Frank Michel, Alexander Krull, Eric Brachmann, Michael Ying Yang, Stefan Gumhold, and Carsten Rother. Pose estimation of kinematic chain instances via object coordinate regression. In *British Machine Vision Conference*, pp. 181.1–181.11, 2015.

Edwin Olson. AprilTag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pp. 3400–3407. IEEE, 2011.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, 2024.

Evin Pınar Örnek, Yann Labbé, Bugra Tekin, Lingni Ma, Cem Keskin, Christian Forster, and Tomas Hodan. Foundpose: Unseen object pose estimation with foundation features. In *European Conference on Computer Vision*, pp. 163–182. Springer, 2024.

Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anikait Singh, Anthony Brohan, et al. Open x-embodiment: Robotic learning datasets and RT-X models. *arXiv preprint arXiv:2310.08864*, 2023.

Tanner Schmidt, Richard A. Newcombe, and Dieter Fox. DART: Dense articulated real-time tracking. In *Robotics: Science and Systems*, volume 2, pp. 1–9, 2014.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Tutian Tang, Minghao Liu, Wenqiang Xu, and Cewu Lu. Kalib: Easy hand-eye calibration with reference point tracking. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 17948–17955. IEEE, 2025.

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.

Yiming Zuo, Weichao Qiu, Lingxi Xie, Fangwei Zhong, Yizhou Wang, and Alan L. Yuille. CRAVES: Controlling robotic arm with a vision-based economic system. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4214–4223, 2019.