Bi-Manual Joint Camera Calibration and Scene Representation

Haozhan Tang[1] Tianyi Zhang[1] Matthew Johnson-Roberson[1,2] Weiming Zhi[1]

***Abstract*—Robot manipulation, especially bimanual manipulation, often requires setting up cameras on different robot manipulators. Before robot manipulators can generate motion or even build representations of their environments, the cameras rigidly mounted to the robot need to be calibrated. Camera calibration is a cumbersome process involving collecting a set of images, with each capturing a pre-determined marker. In this work, we introduce the Bi-Manual Joint Calibration and Representation Framework (Bi-JCR). Bi-JCR enables dual robot manipulators, each with cameras mounted, to circumvent taking images of calibration markers. By leveraging 3D foundation models for dense, marker-free multi-view correspondence, Bi-JCR jointly estimates: (i) the extrinsic transformation from each camera to its end-effector, (ii) the inter-arm relative poses between manipulators, and (iii) a unified, scale-consistent 3D representation of the shared workspace, all from the same captured RGB image sets. The representation, jointly constructed from images captured by cameras on both manipulators, lives in a common coordinate frame and supports collision checking and semantic segmentation to facilitate downstream bimanual coordination tasks. We empirically evaluate the robustness of Bi-JCR on a variety of tabletop environments, and demonstrate its applicability on a variety of downstream tasks.**

I. Introduction

Robot manipulators with wrist-mounted cameras generally need to be meticulously calibrated offline to enable perceived objects to be transformed into the robot's coordinate frame. This is done via a procedure known as hand-eye calibration, where the manipulator is moved through a set of poses and take images of a known calibration marker, such as a checker board or AprilTag [1]. Traditional hand-eye calibration methods focus on a single "eye-in-hand" camera and rely on external markers to compute the rigid transform between camera and end-effector. When extended to two independently moving arms, these approaches must be repeated separately for each arm, and then a secondary step is required to fuse the two coordinate frames. In this work, we tackle the problem of calibrating of dual manipulators with wrist-mounted cameras without using any calibration markers. Here, we assume that the poses of the cameras relative to the end-effectors, along with the relative poses of the manipulator bases are unknown and require estimation.

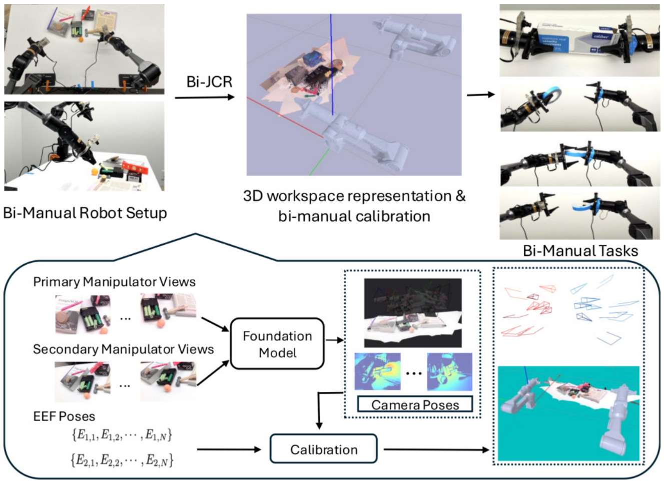

Bi-JCR
Bi-Manual Robot Setup
3D workspace representation & bi-manual calibration
Bi-Manual Tasks
Primary Manipulator Views
...
Secondary Manipulator Views
...
Foundation Model
...
Camera Poses
EEF Poses
$\{E_{1,1}, E_{1,2}, \cdots, E_{1,N}\}$
$\{E_{2,1}, E_{2,2}, \cdots, E_{2,N}\}$
Calibration

Fig. 1: We tackle a bi-manual setup, where the extrinsics of both cameras and the relative poses of the robot bases to one another are unknown. Bi-JCR solves to recover all three transformations.

Here, we propose a framework called Bi-manual Joint Representation and Calibration (Bi-JCR) that simultaneously builds a representation of the environment and calibrates both cameras for dual-manipulators with wrist-mounted cameras. Bi-JCR uses the same set of images for both calibrating the camera and constructing the environment representation, thereby avoiding the need to calibrate offline with markers. Bi-JCR leverages modern *3D foundation models* to efficiently estimate an unscaled representation along with unscaled camera poses, from a set of images captured by the dual manipulators. Then, by considering the forward kinematics of each arm, we formulate a joint scale recovery and dual calibration problem which can subsequently be solved via gradient descent on a manifold of transformation matrices. By optimizing across a single calibration problem defined using images from both arms, Bi-JCR simultaneously solves for each hand-eye transform, aligns the two robot base frames, recovers a missing scale factor, and directly yields the rigid transform between the two camera sensors. This enables immediate fusion of visual data across both viewpoints for bimanual manipulation, without reliance on external markers or depth sensors.

We empirically evaluate Bi-JCR and demonstrate its ability to accurately calibrate cameras on both manipulators, and produce a dense and size-accurate representation of the environment that can be transformed into the workspace coordinate frame. We leverage the representation into downstream manipulation and to execute successful grasps and bi-manual hand-overs. Concretely, our contributions include:

- The Bi-manual Joint Calibration and Representation (Bi-JCR) method that leverages 3D foundation models to build an environment representation built from wrist-mounted cameras, while calibrating the cameras;
- The formulation of a novel optimization problem that recovers camera transformations on both manipulators, the relative pose between the manipulators, and a scale factor to obtain metric scale from representation;
- Rigorous evaluation on real-world data, and the evaluation of performance of downstream tasks in a real-world bi-manual setup.

[1] Robotics Institute, Carnegie Mellon University, USA.
[2] College of Connected Computing, Vanderbilt University, USA.

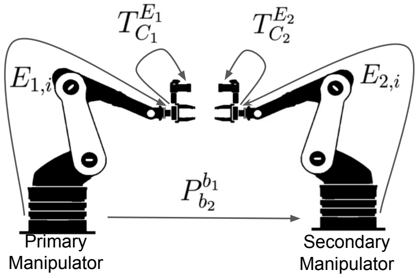


Fig. 2: The transformations between each arm, along with their cameras, are shown here. The transformations $T_{C_1}^{E_1}$, $T_{C_2}^{E_2}$, and $P_{b_2}^{b_1}$ are all unknown and will be recovered via Bi-JCR.

II. Related Work

Hand–Eye Calibration: Decades-old closed-form solvers [2]–[4] remain the de facto standard because they are fast and provably correct under marker-based conditions without noise. Yet in modern labs, the very assumptions they rely on,static checkerboards, perfectly segmented corners, independent wrist motion, are routinely violated. Recent learning-based variants regress the transform directly from images [5], but demand gripper visibility or prior CAD models and often degrade sharply outside the synthetic domain in which they are trained. Other end-to-end policies learning-based methods bypass calibration entirely, mapping pixels to torques [6], [7], but at the cost of losing an explicit transform that downstream planners and safety monitors still persist. Foundation model for calibration is also explored in [8], but has not been extended to the bi-manual setting. **Bi-manual Hand-Eye Calibration:** Extending single manipulator hand-eye calibration to the bi-manual setup is not trivial as it requires finding the pose of the secondary manipulator in the primary manipulator's frame. Previously, some methods relied on the secondary manipulator holding a checkerboard to perform bi-manual calibration [9], [10]. Recently, a graph-based method also uses Apritags [1] to calibrate multiple manipulators simultaneously [11]. **Scene Representation:** Bimanual manipulation requires reasoning about *shared* workspaces where two end-effectors and several movable objects compete for space. Classical metric maps such as occupation grids [12] and signed distance fields [13], [14] give fast binary or distance queries for collision checking, yet they discretize space and struggle to capture fine contact geometry in small parts. Continuous alternatives such as Gaussian process maps [15], kernel regressors [16], [17], and neural implicit surfaces [18], offer subvoxel accuracy. Learning methods that *directly* consume point clouds [19], [20] or integrate them into trajectory optimization [21] completely bypass explicit mapping, placing the computational burden on a powerful network. In vision, photorealistic NeRF-style scene encoders [22]–[24] also produce rich geometry and texture depending on Structure-from-Motion initialization [25]. **Foundation Models:** Large-scale models trained in web corpora, LLMs in NLP [26] and multimodal encoders in vision [27] have recently been extended to 3-D perception [28]–[31]. In robotics, these large pre-trained deep learning models are referred to as ”foundation models”, gaining increasing applications when used as black box functions for downstream tasks [32], [33], although the output of these models requires potential post-processing [34] before robotic tasks.

III. Bi-manual Joint Representation and Calibration

The proposed Bi-manual Joint Representation and Calibration (Bi-JCR) framework aims to solve the eye-to-hand calibration problem for both manipulators, and in the process, also recover the relative poses of the robot base. At the same time, we can recover a dense 3D representation of the tabletop scene, which can facilitate downstream manipulation. This process is done without relying on any camera pose information from external markers, such as checkerboards or AprilTags [1].

A. Problem Setup:

We consider two manipulators, with one designated as the *primary* manipulator and the other as the *secondary*, each equipped with an end-effector mounted low-cost RGB camera. A set of objects is arranged on a tabletop within their shared workspace. The rigid transformations from each camera to its corresponding end–effector are unknown and must be estimated. To collect data, we command each manipulator through a sequence of N distinct end–effector poses, capturing an RGB image at each pose. We denote the primary manipulator's poses by $\{E_{1,1}, E_{1,2}, \cdots, E_{1,N}\}$ with N corresponding RGB images $\{I_{1,1}, I_{1,2}, \cdots, I_{1,N}\}$. We denote the N poses for the secondary manipulator as $\{E_{2,1}, E_{2,2}, \cdots, E_{2,N}\}$, with corresponding RGB images $\{I_{2,1}, I_{2,2}, \cdots, I_{2,N}\}$.

Using only these end–effector poses and captured images, Bi-JCR will recover all of the following: The rigid transformation $T_{C_1}^{E_1}$ from Camera 1 to the primary end–effector; The rigid transformation $T_{C_2}^{E_2}$ from Camera 2 to the secondary end–effector; The scale factor λ aligning the foundation model's output frame with the real-world metric frame; The pose of the secondary base frame b_2 relative to the primary base frame b_1, denoted $P_{b_2}^{b_1}$; The transformation $T_w^{b_1}$ from the foundation model's output frame w to the primary base frame b_1; A metric-scale 3D reconstruction of the scene in the primary base frame b_1.

The transformations between the dual manipulators, along with their attached cameras, are illustrated in fig. 2. Here, we observe that only the forward kinematics of the manipulators, i.e. the transformation from the bases of the manipulators to their end-effector, is known. The relative position of the two manipulators are also initially unknown. Bi-JCR solves for all of the unknown transformations.

B. 3D Foundation Models in the Pipeline

The two image sets can be fed into a 3D foundation model to obtain a reconstruction of the scene in an arbitrary coordinate frame and scale. We recover both of the relative camera poses

$$\{P_{1,1}, \ldots, P_{1,N},\, P_{2,1}, \ldots, P_{2,N}\} \tag{1}$$

along with corresponding point sets containing the reconstruction,

$$\{\hat{X}_{1,1}, \ldots, \hat{X}_{1,N},\, \hat{X}_{2,1}, \ldots, \hat{X}_{2,N}\}, \tag{2}$$

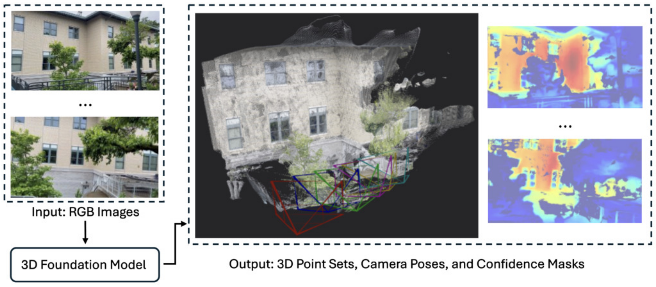


Fig. 3: 3D foundation model taking a set of RGB images, and output 3D point sets, camera poses and confidence maps.

where each point in a set corresponds to a pixel in the associated input image, and confidence values for each pixel can also be recovered. Pre-trained 3D foundation models are often used to extract structure from images of indoor scenes and building structures, and their application for table-top scenes has been under-explored. Example outputs from the model, DUSt3R [28], is given in fig. 3. Because the foundation model recovers geometry only up to an unknown scale, both the estimated camera poses and the aggregated point cloud are not expressed in real-world metric units. To resolve this scale ambiguity, we introduce a *scale factor* λ so that the pose of camera i on manipulator m in the real-world (or base) frame w becomes

$$P^{w}_{m,i}(\lambda) = \begin{bmatrix} R_{m,i} & \lambda\, t_{m,i} \\ 0 & 1 \end{bmatrix} \in \mathrm{SE}(3), \tag{3}$$

where we have $i = 1, \ldots, N$ and the index, $m \in \{1, 2\}$. In this expression, $R_{m,i} \in \mathrm{SO}(3)$ denotes the rotation and $t_{m,i} \in \mathbb{R}^3$ is the translation estimated by the foundation model. Next, defining the transform from the scaled foundation frame w to each manipulator's base frame b_m as $T^{b_m}_{w}$, the camera poses in the real-world base frames are

$$P^{b_m}_{m,i}(\lambda) = T^{b_m}_{w}\, P^{w}_{m,i}(\lambda), \quad i = 1, \ldots, N. \tag{4}$$

C. Solving for Initial Calibration Solution

In Bi-JCR, we seek to simultaneously solve for λ, $T^{b_1}_{w}$, and $T^{b_2}_{w}$ in the process of solving bi-manual hand-eye calibration for the two camera frame to base frame transformations $T^{E_1}_{C_1}$ and $T^{E_2}_{C_2}$. During the sequence of manipulator motions, the transformation between scaled camera poses and end effector poses for manipulator $m \in \{1, 2\}$ can be formulated as the classical hand-eye calibration equations [2]:

$$E_{m,i}{}^{-1} E_{m,i+1} T^{E_m}_{C_m} = T^{E_m}_{C_m} P^{b_m}_{m,i}{}^{-1}(\lambda) P^{b_m}_{m,i+1}(\lambda) \tag{5}$$

Now, we denote the transformation between poses by,

$$T^{E_{m,i+1}}_{E_{m,i}} = E_{m,i}{}^{-1} E_{m,i+1}, \tag{6}$$

$$T^{P^w_{m,i+1}}_{P^w_{m,i}}(\lambda) = P^{w}_{m,i}{}^{-1}(\lambda) P^{w}_{m,i+1}(\lambda). \tag{7}$$

Then, the hand-eye equations can be formulated as

$$T^{E_{m,i+1}}_{E_{m,i}} T^{E_m}_{C_m} = T^{E_m}_{C_m} T^{P^w_{m,i+1}}_{P^w_{m,i}}(\lambda). \tag{8}$$

Here we observe that eq. (8) admits the $AX = XB$ form of the classical hand-eye calibration problem, with the right-hand side dependent on the scale factor λ.

The first phase of Bi-JCR aims to obtain an initial solution for the scale and desired transformation, which we denote as λ', $T^{E_1}_{C_1}{}'$ and $T^{E_2}_{C_2}{}'$. Since the rotation component here are in $\mathbf{SO}(3)$, we can first solve for the rotation components of the transformations, which are invariant to the scale factor. This can be achieved via linear algebra on the manifold of rotation matrices by following [4]. We convert rotation components of $T^{E_{m,i+1}}_{E_{m,i}}$ and $T^{P^w_{m,i+1}}_{P^w_{m,i}}$ into the log map of $\mathbf{SO}(3)$ to its lie algebra ($\mathfrak{so}(3)$) where for some $R \in \mathbf{SO}(3)$. This gives,

$$\omega = \arccos\left(\frac{\mathrm{Tr}(R) - 1}{2}\right), \tag{9}$$

$$\mathrm{LogMap}(R) := \frac{\omega}{2\sin(\omega)} \begin{bmatrix} R_{3,2} - R_{2,3} \\ R_{1,3} - R_{3,1} \\ R_{2,1} - R_{1,2} \end{bmatrix} \in \mathfrak{so}(3). \tag{10}$$

where, $\mathrm{Tr}(\cdot)$ indicates the trace operator and the subscripts indicate the elements' indices in R. Then we can find best fit rotational components via:

$$R^{E_m}_{C_m}{}' = (M_m^{\top} M_m)^{-\frac{1}{2}} M_m^{\top}, \tag{11}$$

$$\text{where } M_m = \sum_{i=1}^{N-1} \mathrm{LogMap}(R^{E_{m,i+1}}_{E_{m,i}}) \otimes \mathrm{LogMap}(R^{P^w_{m,i+1}}_{P^w_{m,i}}),$$

The $\otimes$ is the outer product, and the matrix inverse square root can be computed efficiently via singular value decomposition.

Next, we solve the translation components along with the scale factor jointly, by minimizing the residuals of the similar scale recovery problem formulated in [8] for each arm, assuming that the scale factor is consistent for the results of each arm:

$$\text{SRP:} \quad \arg\min_{t^{E_m}_{C_m}{}', \lambda'} \sum_{i=1}^{N-1} ||Q_i t^{E_m}_{C_m}{}' - \mathbf{d}_i(\lambda')||_2^2, \tag{12}$$

$$\text{where } Q_i = I - R^{E_{m,i+1}}_{E_{m,i}}, \tag{13}$$

$$\text{and } \mathbf{d}_i(\lambda') = t^{E_{m,i+1}}_{E_{m,i}} - R^{E_m}_{C_m}{}'(\lambda) t^{P^w_{m,i+1}}_{P^w_{m,i}}. \tag{14}$$

Equation (12) can be solved via least-squares, and we obtain our solutions λ', $T^{E_1}_{C_1}{}'$ and $T^{E_2}_{C_2}{}'$, which can be further refined via gradient-based optimisation.

D. Refine Calibration through Gradient-based Optimization

We further refine the solutions via gradient descent to improve estimation. Here, we first rearrange Equation (8) into,

$$T^{E_{m,i+1}}_{E_{m,i}} T^{E_m}_{C_m} - T^{E_m}_{C_m} T^{P^w_{m,i+1}}_{P^w_{m,i}} = 0,$$

$$\text{for } i \in \{1, \cdots, N-1\}, m \in \{1, 2\}. \tag{15}$$

then we can solve the calibration problem by minimizing the difference between transformation matrices $T^{E_{m,i+1}}_{E_{m,i}} T^{E_m}_{C_m}$ and $T^{E_m}_{C_m} T^{P^w_{m,i+1}}_{P^w_{m,i}}$. Specifically, we define a cost function as,

$$\ell(\lambda, T^{E_1}_{C_1}, T^{E_2}_{C_2}) =$$

$$\sum_{m\in\{1,2\}} \Big(\frac{1}{N-1} \sum_{i=1}^{N-1} \alpha D_R(T^{E_m}_{C_m}) + (1-\alpha) D_t(\lambda, T^{E_m}_{C_m})\Big),$$

$$\text{where } D_R(T^{E_m}_{C_m}) = \arccos\big(\mathrm{tr}(RE^{m,i+1}_{m,i}{}^{\top} RC^{m,i+1}_{m,i}) - 1\big),$$

$$\text{and } D_t(\lambda, T^{E_m}_{C_m}) = \big\|te^{m,i+1}_{m,i} - tc^{m,i+1}_{m,i}\big\|_2, \tag{16}$$

where $\mathrm{tr}(\cdot)$ is the trace operator. We can then minimize the cost function via gradient descent [35] while constraining the rotation to be on the $\mathbf{SO}(3)$ manifold. Here, we use the results from the previous section as the initial solution. Furthermore, to ensure that the rotational components in $\mathbf{SO}(3)$ during the backpropagation process, we follow [36] and first pull each rotation into the Lie algebra with the logarithm map, then perform the gradient update on the resulting axis–angle vector in $\mathbb{R}^3$, and push it back onto the manifold via the exponential map. With the solutions that minimize the cost function, we can then obtain the world-to-base transformations $T_w^{b_1}$, and $T_w^{b_2}$ via

$$T_w^{b_m} = \underset{i \in \{1,\cdots,N-1\}}{\mathrm{AVG}_{\mathrm{SE3}}} \left(T_{E_{m,i}}^{E_{m,i+1}} T_{C_m}^{E_m} {T_{P_{m,i}^w}^{P_{m,i+1}^w}}^{-1} \right) \tag{17}$$

where $\mathrm{AVG}_{\mathrm{SE3}}$ is the average over a set of transformation matrices on $\mathbf{SE}(3)$, by considering the average of rotation and translation separately.

E. Obtaining Metric-Scale 3D Representation

Here, we seek to build a real-world metric scale 3D representation of the environment under the primary manipulator's frame. Following Equation (3), we first scale camera poses by λ to get $\{P_{1,1}^w, \cdots, P_{1,N}^w\}$ and $\{P_{2,1}^w, \cdots, P_{2,N}^w\}$, we can then get the calibrated metric scale camera poses by

$$P_{m,i}^{b_1} = T_w^{b_1} P_{m,i}^w, \text{ for } m \in \{1, 2\}, i \in \{1, \cdots, N\}. \tag{18}$$

Next, we also want to use the point sets from each arm, associated with each input image, $\{\hat{X}_{1,1}, \cdots, \hat{X}_{1,N}\}$ and $\{\hat{X}_{2,1}, \cdots, \hat{X}_{2,N}\}$ and their associated confidence maps $\{C_{1,1}, \cdots, C_{1,N}\}$ and $\{C_{2,1}, \cdots, C_{2,N}\}$ from the output of the foundation model to recover a rich and high-quality representation of the environment. We first use a confidence threshold to filter out points below this threshold in each $\hat{X}_{m,i}$. Then, we transform the points from the filtered point sets, $\{\mathbf{x}_i\}_{i=1}^{N_{pc}}$, to get a point cloud in real-world metric scale and primary manipulator's base frame, $\{\mathbf{x}_i^{b_1}\}_{i=1}^{N_{pc}}$ through

$$\{\mathbf{x}_i^{b_1} = T_w^{b_1}(\lambda \mathbf{x}_i), \text{ for } i \in \{1, \cdots, N_{pc}\}\}. \tag{19}$$

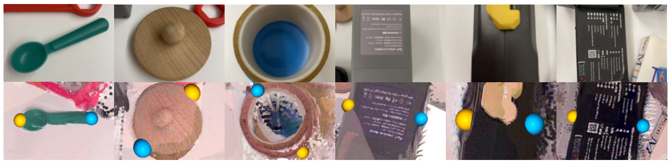
Furthermore, the pose of the secondary manipulator's base in the primary manipulator's base frame can be computed as

$$P_{b_2}^{b_1} = T_w^{b_1} {T_w^{b_2}}^{-1}. \tag{20}$$

The pose of the secondary manipulator's base in the primary manipulator's base frame enables us to compute end-effector poses for downstream tasks of both manipulators in a single unified frame. This facilitates downstream processes, such as object segmentation along with grasping generation, to operate.

IV. Empirical Evaluation

In this section, we rigorously evaluate our proposed Bi-Manual Joint Calibration and Representation (Bi-JCR) method. Our bi-manual setup consists of two AgileX Piper 6 degree-of-freedom manipulators, each with a low-cost USB webcam mounted on the gripper. We seek to answer the following questions: Can Bi-JCR produce correct hand-eye calibration for both arms, even when the number of images provided is low?Can Bi-JCR recover the scale accurately such that our representation's sizes match the physical world? Can high-quality 3D environment representations, in the correct coordinate frame, be built? Does Bi-JCR facilitate downstream bi-manual manipulation tasks?



(a) Spoon (b) Tea Lid (c) Tape (d) Battery (e) Toolbox (f) Joystick

Fig. 4: Visualization of the objects we used for scale validation in Table II. For each object, the top is their real-world appearances, and the bottom is the reconstructions. The blue and yellow dots specify the length measured for scale validation.

A. Eyes-to-Hands Calibration with Bi-JCR

Baselines: COLMAP-based pose estimation and Ray Diffusion. To assess the calibration quality of Bi-JCR, we compare against two alternatives. First, in the absence of high-contrast markers such as checkerboards or AprilTags [1], we use SfM via COLMAP [25] and apply the Park–Martin algorithm [4] to compute eye-to-hand transformations for each manipulator. Second, we evaluate Ray Diffusion [37], a sparse-view diffusion model trained on large datasets [38] that directly regresses camera poses.

Task and Metrics: We take images in three different environments: a scene on a darker light condition with 9 items (scene A), two others of which are under brighter light conditions with different sets of objects of 9 and 10 items respectively (scene B, scene C). Bi-JCR and the two baseline methods are evaluated with an increasing number of input images (4, 7 and 9 images per manipulator), then check whether the calibration has converged correctly by considering residual losses via Equation (15) with ground truth, obtained via Apriltags [1], on the right-hand side. Lower residual values indicate a higher degree of consistency.

Results: We tabulate our results in Table I. We observe that COLMAP often results in diverged calibration, as images where correspondence cannot be found are discarded. Whether the calibration has converged and the number of images used, from each manipulator, are also shown in Table I. Although Ray Diffusion registered all images, it registered them in an inconsistent way under this tabletop setup of cluttered, partially visible objects, causing the calibration optimizer to accumulate large errors. Our Bi-JCR method consistently produce smaller residual under both lower and higher number of views for both manipulators, showing remarkable image efficiency. We also observe a residual loss reduce trend as the number of images gradually increase, in comparison to the residual loss fluctuation in the other two baseline methods, which shows Bi-JCR's reliable precision gain with increasing number of views.

Here, we also visualize the aligned camera and end-effector poses after calibration via Bi-JCR in Figure 6. The end-effectors and outlined as U-shapes and cameras are represented by cones. Both end-effector poses and camera poses are transformed to the primary manipulator's base frame using the base to base transformation estimated by Bi-JCR. Primary manipulator end-effector and eye-to-hand transformed camera are colored in red, and the secondary manipulator's are colored

		Scene A			Scene B			Scene C		
	Images Per Manipulator	4 images	7 images	9 images	4 images	7 images	9 images	4 images	7 images	9 images
Bi-JCR (Ours)	Converged	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Residual δ_R	0.0769	0.0724	0.0668	0.0740	0.0617	0.0569	0.0743	0.0634	0.0612
	Residual $\delta_{\mathbf{t}}$	0.0461	0.0391	0.0378	0.0587	0.0351	0.0340	0.0424	0.0341	0.0373
	No. of Poses (Left)	4	7	9	4	7	9	4	7	9
	No. of Poses (Right)	4	7	9	4	7	9	4	7	9
COLMAP [25] + Calibration	Converged	×	×	×	✓	✓	✓	×	✓	✓
	Residual δ_R	NA	NA	NA	0.0865	0.0583	0.0584	NA	0.0591	0.0586
	Residual $\delta_{\mathbf{t}}$	NA	NA	NA	0.0317	0.0269	0.0274	NA	0.0277	0.0271
	No. of Poses (Left)	0	4	2	4	7	9	0	7	9
	No. of Poses (Right)	4	0	0	4	7	9	4	7	9
Ray Diffusion [37] + Calibration	Residual δ_R	0.4845	0.4288	0.7951	0.4948	0.3839	0.3634	0.4504	0.2610	0.2223
	Residual $\delta_{\mathbf{t}}$	0.2118	0.1416	0.1309	0.2067	0.2051	0.1637	0.2125	0.1858	0.1760
	No. of Poses (Left)	4	7	9	4	7	9	4	7	9
	No. of Poses (Right)	4	7	9	4	7	9	4	7	9

TABLE I. Quantitative evaluation on Bi-JCR's calibration residual error (δ_R and δ_t) against baseline methods. Lower residual indicates more accurate calibrations.

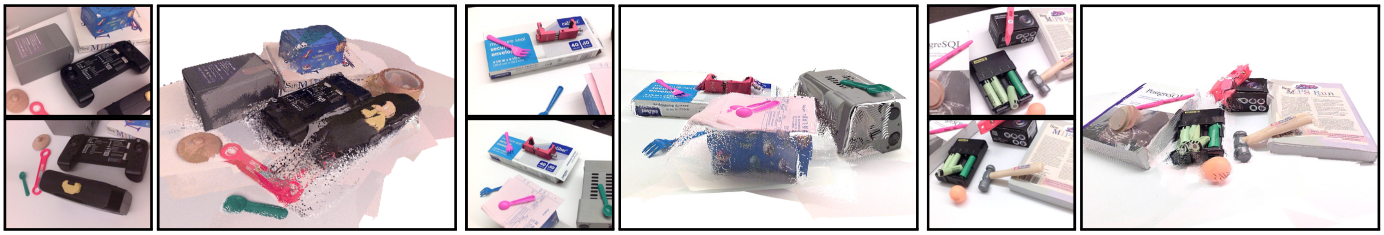


Fig. 5: Qualitative evaluations of the recovered 3D reconstructions, including scene A, scene B, scene C from left to right. The reconstruction of the scene is dense and geometrically accurate.

Object	Spoon	Tea Lid	Tape	Battery	Toolbox	Joystick
5 Images	4.90%	4.3%	2.73%	1.59%	10.90%	7.73%
8 Images	1.61%	0.26%	1.34%	1.10%	2.53%	2.98%

TABLE II. Percentage of error of object dimensions to compare the size of real-world objects against reconstructed objects. Within 8 images per manipulator, the percentage errors of reconstructed sizes are at most 2.98%, indicating accurate scale recovery.

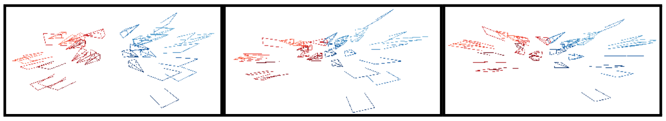


Fig. 6: Qualitative evaluation of the camera calibration in the three scenes. We observe that the cameras indicated as cones are aligned with the end-effector poses, indicating accurate calibration. The end-effector and camera poses of the left manipulator are colored in shades of red and those of the right in blue.

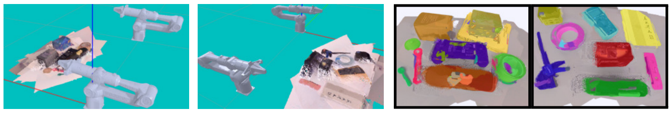
in blue. As shown in Figure 6, Bi-JCR successfully recovers eye-to-hand transformations that consistently align camera and end-effector poses for both primary and secondary manipulator across all scenes.

B. Accurate Metric Scale Recovery with Bi-JCR

Bi-JCR also reconstructs a 3D dense point cloud on a metric scale of the real-world environment. Here, we evaluate the accuracy of scale recovery by comparing the difference between the side length of the real-world object and the side length of the reconstructed objects with 5 and 8 images collected from each manipulator, as shown in Figure 4. The error, computed by

$$\text{err}_{obj} = \frac{|s_{\text{reconstructed}} - s_{\text{real world}}|}{s_{\text{real world}}}, \tag{21}$$

is reported in Table II. With only 8 images per manipulator, Bi-JCR is able to reduce the error to a median of 1.48% and at most 2.98%, which marks precise scale recovery giving real-world metric scale.



(a) Visualized in Pybullet. (b) Segmentated tabletop.

Fig. 7: Visualization of running segmentation algorithm in the real world metric scale reconstructed 3D representation from Bi-JCR, which allows various bi-manual downstream tasks such as joint grasping and passing.

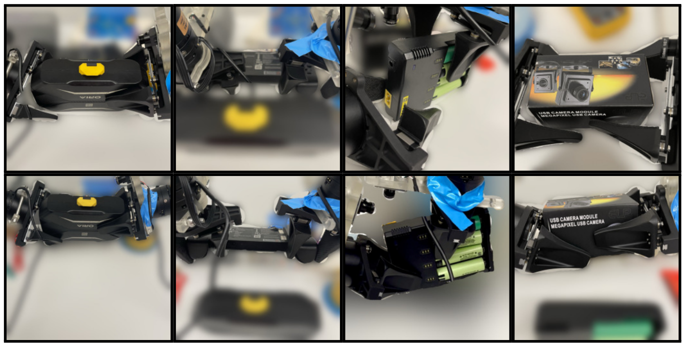


Fig. 8: Execution of the bi-manual joint grasping on heavier objects in the scene. Background blurred for greater clarity.

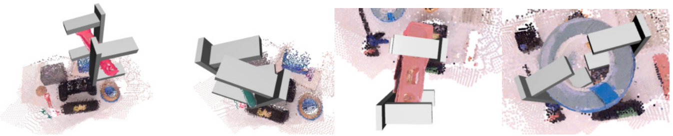


Fig. 9: Top: Selected objects (wrench, spoon, balance meter and tape) segmented; Bottom: Generated robot end-effector grasping poses for manipulator hand-overs.

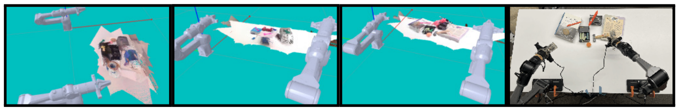


Fig. 10: Visualization of the base to base transformation recovery and transformation recovery of foundation model output frame to primary manipulator's base frame, including scene A, B, C, and the real world bi-manual bases setup (bottom right).

C. 3D Representation in Primary Manipulator's Base Frame

Besides scale, the quality of the 3D representation built by Bi-JCR is critical to downstream tasks. Here, we qualitatively assess the reconstructed 3D point cloud by visualizing it against images taken on the real world environment in Figure 5. We observe that Bi-JCR reconstructs the relative position and orientation of objects in the environment correctly, and the shape of each object is highly preserved. We further investigated whether representations can be accurately transformed into the robot's coordinate frame and the placement of the secondary manipulator base in the primary manipulator base frame. We inject the reconstruction, along with manipulator poses, into the PyBullet Simulator [39]. As shown in Figure 10, the relative pose of the two manipulators highly resembles the relative pose of the two manipulators in the real world, indicating the correct estimation of the pose of the secondary manipulator in the base frame of the primary manipulator. The table 3D reconstruction in simulation remains parallel to the bases of manipulators, and object orientations and positions are visually correct relative to the bases of manipulators, indicating high-quality both a 3D reconstruction and its accurate transformation into the primary manipulator's frame.

D. Bi-JCR Enables Bi-manual Downstream Tasks

To assess both the relative-pose estimation between the two manipulators' bases and the fidelity of the reconstructed 3D scene, we conduct two real-world bimanual tasks: (1) joint grasping of large objects and (2) passing of small objects. We begin by running Bi-JCR to recover the base-to-base transformation and reconstruct the 3D environment in simulation (fig. 7a). Next, we apply the 3D point-cloud segmentation algorithm from [40] to isolate each object on the tabletop (fig. 7b).

Joint grasping: We first select the cluster corresponding to the large objects: a toolbox, a controller, a battery pack and a box, as illustrated fig. 8. A grasp pose for the primary manipulator is computed in its own base frame using [41], and likewise for the secondary manipulator. The secondary grasp must then be transformed into its base frame. Finally, both end-effector poses are executed via inverse kinematics and joint control to perform the coordinated grasp.

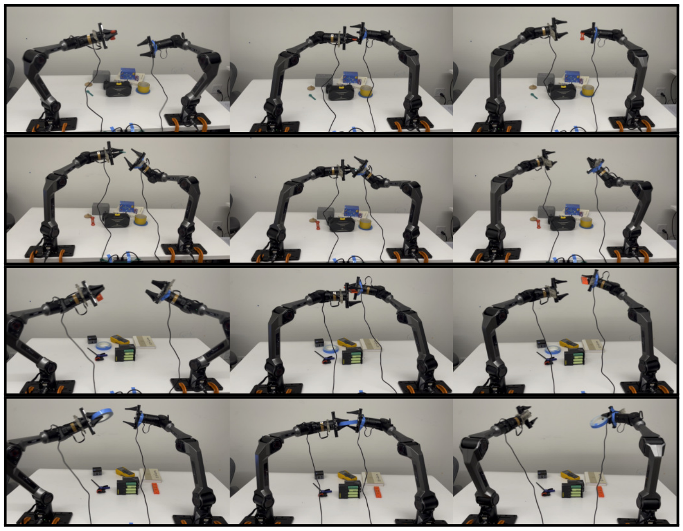


Fig. 11: We demonstrate passing of the objects: wrench, spoon, balance meter, tape from top to bottom.

Bimanual passing: We then focus on the small objects, a wrench, a spoon, a balance meter and a tape, shown in Fig. 9. After choosing a target transfer location in the primary manipulator's base frame, we translate the segmented point cloud to that location and generate synchronized "pass-in-air" poses for both arms. Again, the secondary end-effector pose is reprojected into its base frame. The primary manipulator loads the object onto its gripper and then successfully hands it off to the secondary manipulator at the specified location.

These experiments demonstrate that, given an accurate base-to-base pose estimate and a high-quality dense 3D reconstruction in the primary manipulator's frame, Bi-JCR reliably enables complex bimanual operations.

V. Conclusion

We introduced Bi-JCR, a unified framework for joint calibration and 3D representation in bimanual robotic systems with wrist-mounted cameras. By leveraging 3D foundation models, Bi-JCR eliminates the need for calibration markers and enables simultaneous estimation of camera extrinsics, inter-arm relative poses, and a shared, metric-consistent scene representation. Our approach unifies the calibration and perception processes using only RGB images, and supports downstream manipulation tasks such as grasping and object handover. Extensive real-world evaluations demonstrate Bi-JCR's robustness and its ability to generalize across diverse environments and foundation model backbones.

References

[1] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *IEEE International Conference on Robotics and Automation*, 2011.

[2] R. Y. Tsai and R. K. Lenz, "A new technique for fully autonomous and efficient 3d robotics hand/eye calibration," *IEEE Trans. Robotics Autom.*, 1988.

[3] R. Horaud and F. Dornaika, "Hand-eye calibration," *I. J. Robotic Res.*, 1995.

[4] F. Park and B. Martin, "Robot Sensor Calibration: Solving AX = XB on the Euclidean Group," *IEEE Transactions on Robotics and Automation*, 1994.

[5] E. Valassakis, K. Dreczkowski, and E. Johns, "Learning eye-in-hand camera calibration from a single image," in *Proceedings of the 5th Conference on Robot Learning*, 2022.

[6] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," in *Advances in Neural Information Processing Systems*, 2021.

[7] P. Florence, C. Lynch, A. Zeng, O. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson, "Implicit behavioral cloning," in *Conference on Robot Learning (CoRL)*, 2021.

[8] W. Zhi, H. Tang, T. Zhang, and M. Johnson-Roberson, "Unifying representation and calibration with 3d foundation models," *IEEE Robotics and Automation Letters*, 2024.

[9] L. Wu, J. Wang, L. Qi, K. Wu, H. Ren, and M. Q.-H. Meng, "Simultaneous hand–eye, tool–flange, and robot–robot calibration for comanipulation by solving the axb=ycz problem," *IEEE TRansactions on robotics*, vol. 32, no. 2, pp. 413–428, 2016.

[10] Z. Fu, J. Pan, E. Spyrakos-Papastavridis, X. Chen, and M. Li, "A dual quaternion-based approach for coordinate calibration of dual robots in collaborative motion," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4086–4093, 2020.

[11] Z. Zhou, L. Ma, X. Liu, Z. Cao, and J. Yu, "Simultaneously calibration of multi hand–eye robot system based on graph," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 5, pp. 5010–5020, 2023.

[12] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE Journal on Robotics and Automation*, 1987.

[13] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: a level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995.

[14] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, 2011.

[15] S. O'Callaghan, F. T. Ramos, and H. Durrant-Whyte, "Contextual occupancy maps using gaussian processes," in *2009 IEEE International Conference on Robotics and Automation*, 2009.

[16] F. Ramos and L. Ott, "Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent," *International Journal Robotics Research*, 2016.

[17] W. Zhi, R. Senanayake, L. Ott, and F. Ramos, "Spatiotemporal learning of directional uncertainty in urban environments with kernel recurrent mixture density networks," *IEEE Robotics and Automation Letters*, 2019.

[18] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[19] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[20] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.

[21] W. Zhi, I. Akinola, K. van Wyk, N. Ratliff, and F. Ramos, "Global and reactive motion generation with geometric fabric command sequences," in *IEEE International Conference on Robotics and Automation, ICRA*, 2023.

[22] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.

[23] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, 2022.

[24] T. Zhang, K. Huang, W. Zhi, and M. Johnson-Roberson, "Darkgs: Learning neural illumination and 3d gaussians relighting for robotic exploration in the dark," 2024.

[25] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," *CoRR*, 2023.

[27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, 2021.

[28] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "Dust3r: Geometric 3d vision made easy," in *CVPR*, 2024.

[29] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[30] V. Leroy, Y. Cabon, and J. Revaud, "Grounding image matching in 3d with mast3r," 2024.

[31] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, "Vggt: Visual geometry grounded transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[32] R. Bommasani and et al., "On the opportunities and risks of foundation models," *CoRR*, 2021.

[33] R. Firoozi, J. Tucker, S. Tian, A. Majumdar, J. Sun, W. Liu, Y. Zhu, S. Song, A. Kapoor, K. Hausman, B. Ichter, D. Driess, J. Wu, C. Lu, and M. Schwager, "Foundation models in robotics: Applications, challenges, and the future," *CoRR*, 2023.

[34] W. Zhi, H. Tang, T. Zhang, and M. Johnson-Roberson, "3d foundation models enable simultaneous geometry and pose estimation of grasped objects," *arXiv preprint arXiv:2407.10331*, 2024.

[35] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*, vol. 4. Springer, 2006.

[36] R. Brégier, "Deep regression on manifolds: a 3D rotation case study," 2021.

[37] J. Y. Zhang, A. Lin, M. Kumar, T.-H. Yang, D. Ramanan, and S. Tulsiani, "Cameras as rays: Pose estimation via ray diffusion," in *International Conference on Learning Representations (ICLR)*, 2024.

[38] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, "Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction," in *International Conference on Computer Vision*, 2021.

[39] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning." http://pybullet.org, 2016–2019.

[40] H. Tang, T. Zhang, O. Kroemer, M. Johnson-Roberson, and W. Zhi, "Graphseg: Segmented 3d representations via graph edge addition and contraction," *arXiv preprint arXiv:2504.03129*, 2025.

[41] A. Ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp pose detection in point clouds," *The International Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1455–1473, 2017.