CAAI Transactions on Intelligence Technology

WILEY

The Institution of Engineering and Technology

ORIGINAL RESEARCH OPEN ACCESS

MRPose: Multi-Robot Relative 6D Pose Estimation for Unseen Objects From RGB Images

Jiale Ren[1] | Mingxin Tan[1] | Mengyuan Liu[1] | Yingxuan You[2] | Hong Liu[1]

[1]State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School, Shenzhen, China | [2]Computer Vision Lab, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

Correspondence: Hong Liu (hongliu@pku.edu.cn)

Received: 4 January 2026 | **Revised:** 11 June 2026 | **Accepted:** 5 August 2026

Keywords: 6D object pose estimation | cross-view feature interaction | relative pose estimation

ABSTRACT

Relative pose estimation under the query–reference paradigm has emerged as a practical alternative for estimating the 6D pose of unseen objects without relying on CAD models or extensive annotations. However, many existing approaches remain difficult to deploy in practice, as their reliance on geometric matching makes robustness highly sensitive to matching quality. In this paper, we present MRPose, a framework for relative 6D object pose estimation under a multi-robot setting, where query and reference views are independently captured by different robots. Without requiring CAD models, shared calibration across robots or pose annotations, MRPose directly estimates the full relative transformation between unseen objects from RGB images. Built upon the power of vision foundation models, MRPose introduces a multiplex feature learning strategy that jointly exploits patch-level appearance correspondence and sparse geometric cues through attention-based cross-view feature aggregation. Instead of relying on traditional geometric solvers such as PnP or essential matrix estimation, the proposed framework directly regresses the full relative 6D pose in an end-to-end manner. Extensive experiments on the LINEMOD and YCB-Video benchmarks demonstrate that MRPose compares favourably with existing RGB-based relative pose estimation methods. Real-world experiments further validate its applicability in practical scenarios.

1 | Introduction

6D object pose estimation, which aims to recover an object's 3D rotation and translation from visual observations, is a fundamental problem in computer vision with wide-ranging applications in robotic manipulation [1], scene understanding [2] and augmented reality [3]. Early research primarily focused on instance-level [4–7] pose estimation, where accurate CAD models and detailed annotations are assumed to be available. Subsequent category-level methods [8–11] relaxed this requirement by learning canonical representations shared across object instances. More recently, generalisable pose estimation [12–14] has been studied to further extend pose estimation to unseen objects.

Despite this progress, most existing pose estimation approaches still rely on strong assumptions regarding object geometry or reference information. CAD-based methods require accurate object models that are often unavailable in realistic environments, whereas reference view-based methods typically assume access to carefully collected or densely sampled images of the target object. These assumptions limit the applicability of pose estimation in settings where object geometry is unknown,

Jiale Ren and Mingxin Tan contributed equally to this study.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2026 The Author(s). *CAAI Transactions on Intelligence Technology* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology and Chongqing University of Technology.

reference observations are sparse or visual data are acquired opportunistically rather than in a controlled manner.

When pose estimation is treated as an isolated vision task, these constraints may be acceptable. However, as pose estimation is embedded into interactive and embodied perception systems, the availability of object models and canonical references can no longer be taken for granted. Instead of assuming access to complete geometric priors or exhaustive reference observations, perception systems must operate under partial, evolving and distributed visual evidence.

This observation naturally motivates relative object pose estimation, which focuses on predicting the transformation between two observations of the same object rather than estimating its absolute pose in a fixed coordinate system. By operating directly on image pairs, relative pose estimation avoids the need for explicit CAD models and dense reference collections, while reducing dependence on carefully calibrated multi-view setups, and therefore provides a more flexible formulation under realistic constraints. Relative pose estimation is particularly relevant in collaborative perception settings, such as multi-robot systems, where visual observations are independently captured and exchanged among agents. In these scenarios, objects are often observed from arbitrary viewpoints by different robots, and the ability to estimate relative pose between observations becomes essential for sharing spatial information and enabling coordinated interaction. Importantly, such settings further challenge traditional assumptions, as observations may come from heterogeneous sensors and lack shared calibration. For example, a robot may observe an object and acquire a reference view during exploration or prior operation. When another robot later encounters the same object from a different viewpoint, it is often unnecessary to re-estimate the object pose from scratch. Instead, estimating the relative pose between the new observation and the shared reference allows the second robot to quickly align its perception with existing spatial information. Such a mechanism enables efficient knowledge transfer across robots and supports scalable collaboration without requiring shared calibration or object-specific models.

Recent work has demonstrated that relative pose can be inferred from pairs of RGB images without relying on CAD models. However, most existing methods are still designed under a single-agent assumption. They often depend on geometric solvers such as PnP or essential matrix estimation, rely on carefully engineered correspondence pipelines or restrict the output to partial pose components, such as rotation only. Moreover, some approaches introduce additional learned models or optimisation stages to establish correspondences, which further increase computational complexity and inference latency. As a result, these methods are not well suited for collaborative scenarios where observations are independently acquired and full 6D pose information is required.

In this paper, we propose MRPose, a framework for multi-robot relative 6D object pose estimation from RGB images. Given a query and a reference image captured by different robots from arbitrary viewpoints, MRPose directly estimates the full relative transformation between object observations from independently captured RGB images. The proposed framework leverages vision foundation models and introduces a multiplex feature learning strategy that combines patch-level appearance correspondence with sparse geometric cues. An attention-based feature aggregation module is further employed to enable effective cross-view feature interaction for accurate pose regression. Extensive experiments on standard benchmarks demonstrate that MRPose achieves state-of-the-art performance among RGB-based relative pose estimation methods. In addition, real-world multi-robot experiments validate the practicality of the proposed approach, highlighting its potential for deployment in collaborative perception systems. The main contributions are summarised as follows:

1. A multi-robot relative 6D object pose estimation formulation is introduced, enabling direct estimation of the full relative pose without relying on CAD models, shared calibration across robots or pose annotations.
2. A multiplex feature learning framework is presented to jointly exploit patch-level appearance correspondence and sparse geometric cues, together with an attention-based feature aggregation module that facilitates effective cross-view feature interaction and end-to-end pose regression.
3. Extensive evaluations on standard benchmarks, along with real-world multi-robot experiments, demonstrate the effectiveness, robustness and practical applicability of the proposed approach in collaborative perception scenarios.

2 | Related Work

2.1 | Instance-Level Object Pose Estimation

Instance-level object pose estimation methods primarily assume that CAD models of target objects are available and focus on constructing reliable 2D-3D correspondences. Early work such as PoseCNN [4] introduces an end-to-end network that separately regresses translation and rotation from RGB images. Building upon this regression paradigm, YoLoPose V2 [15] employs attention mechanisms to improve feature extraction, significantly enhancing regression-based pose prediction. Other approaches reformulate the task from different perspectives: CDPN [16] predicts dense coordinate maps to establish pixel-wise correspondences for rotation estimation, whereas SSD-6D [5] treats pose estimation as a classification problem by discretising the rotation space. More recently, T6D-Direct [17] leverages the DETR framework [18], framing pose estimation as a set prediction problem that jointly infers object instances and their poses. These methods exhibit strong performance when object CAD models are available but inherently lack generalisation beyond predefined object sets.

2.2 | Category-Level Object Pose Estimation

Although instance-level methods achieve strong performance when accurate CAD models of target objects are available, their reliance on object-specific geometry fundamentally limits their applicability to scenarios involving unseen objects. To overcome this dependency and extend pose estimation to broader real-world settings, category-level approaches aim to generalise across intra-class variations without requiring instance-specific models. NOCS [8] introduces a normalised canonical coordinate space in which different shapes within a category can be

aligned, enabling consistent 2D-3D correspondence learning. Building on this representation, SPD [14] integrates shape priors through deformation fields to better capture intra-category geometric variations, thus improving canonical coordinate prediction. Transformer-based approaches [19], such as SGPA [9], further extend this idea by incorporating cross-modal reasoning over RGB, depth and shape context. Meanwhile, methods such as DualPoseNet [11] bypass canonical spaces and directly regress category-level poses using a dual-branch architecture. Although these category-level methods improve generalisation across object instances within the same category, they still rely on canonical representations, shape priors or category annotations, which limits their applicability to scenarios involving arbitrary unseen objects.

2.3 | Unseen Object Pose Estimation

Although category-level methods broaden the applicability of pose estimation by learning canonical spaces or deformation priors, they still require category annotations or structural assumptions that limit generalisation to unseen objects. To overcome these constraints, recent research has shifted towards generalisable 6D pose estimation, which aims to infer object poses without relying on predefined categories or instance-level models. Existing approaches can be broadly divided into two paradigms: methods that rely on explicit CAD or geometric models to establish pose constraints [20, 21] and methods that leverage reference images of the target object without requiring precise geometric representations [22, 23]. However, CAD-based methods fundamentally depend on the availability of accurate object geometry, whereas reference-view-based approaches often require carefully collected reference images or dense viewpoints, limiting their robustness and scalability in unconstrained real-world scenarios. Recent methods such as NOPE [24], UNOPose [25] and Any6D [26] are also related, but they assume either object-centric reference–query inputs or a single RGB-D reference or anchor image, rather than our RGB-only multi-robot relative pose setting. Recent 3D foundation-model-based approaches, such as MASt3R [27] and Pos3R [28], further show that 3D-consistent features can provide more robust cross-view correspondences for pose estimation. These models are a promising direction for improving MRPose under large viewpoint changes. In the current pipeline, however, we adopt DINOv2 because it supports both global retrieval for image-pair generation and patch-level matching within a unified RGB-only feature space. Nevertheless, directly replacing DINOv2 with these models is not straightforward in our pipeline because MRPose also requires an image-level representation for efficient query–reference retrieval before patch-level pose regression.

2.4 | Relative Object Pose Estimation

While generalisable unseen object pose estimation alleviates the reliance on category labels and instance-level models, many existing methods still depend on explicit geometric constraints, dense reference views, or carefully engineered correspondence pipelines, which limits their applicability in realistic and collaborative settings. In contrast, relative object pose estimation offers a more flexible formulation by focussing on the relationship between observations rather than absolute object poses. Relative object pose estimation aims to predict the transformation between two observations of the same object, rather than estimating its absolute pose in a global coordinate frame. By operating on image pairs, this formulation avoids the need for explicit CAD models and reduces dependence on carefully calibrated multi-view setups, and has therefore attracted increasing attention in recent years.

Early approaches recover relative pose by establishing dense or sparse correspondences between image pairs and solving geometric constraints using PnP or essential matrix estimation. Methods such as OnePose [29], OnePose++ [30] and Gen6D [31] follow this paradigm by learning feature correspondences from RGB images and recovering pose through geometric solvers. Although effective, these approaches typically require carefully engineered correspondence pipelines or multiple reference views, which limits their scalability in unconstrained settings.

More recent work explores learning-based formulations for relative pose estimation. Among them, POPE [32] is the closest prior work to ours at the level of the high-level problem setting, since it also considers CAD-free pose estimation from a query image and a single reference image of an unseen object. However, POPE is developed for a single-agent promptable setting and relies on a correspondence-and-geometry pipeline with LoFTR-based local matching and geometric post-processing. More generally, many such approaches are designed under a single-agent assumption and do not explicitly consider scenarios where observations are independently captured by different robots. Overall, the problem of multi-robot relative 6D object pose estimation remains underexplored. Existing methods are not designed to handle independently acquired observations, heterogeneous sensing conditions or the need for efficient full-pose estimation in collaborative perception settings. This gap motivates the framework proposed in this paper.

2.5 | Object Localisation for Unseen Pose Estimation

Most object pose estimation pipelines operate on object-centric cropped images rather than full-scene observations. In instance-level and category-level settings, this preprocessing step can usually be handled by a trained object detector or segmenter because the target instances or categories are known during training. In unseen object pose estimation, however, such category-specific detection is no longer directly applicable. As a result, many existing methods either assume the ground-truth mask is available as known input or rely on category-agnostic front-end modules such as CNOS [33]. CNOS combines segmentation proposals with DINOv2 global matching against CAD-rendered template libraries and provides a strong baseline for novel object segmentation. Our setting is related but simpler: Given a query image and a nearby-view full reference image, we only need to crop from the reference image the object corresponding to the query observation. Therefore, our image-pair generation stage does not solve general novel object segmentation over a large template library but instead performs a simplified retrieval task that serves the downstream relative pose estimation pipeline.

3 | Methods

3.1 | Problem Definition

This work considers the problem of relative 6D object pose estimation in a multi-robot, multi-view perception setting, where different robots independently observe the same physical object from arbitrary viewpoints. Importantly, the target object instance is unseen during training, meaning that the specific object observed at test time does not appear in the training set. In addition, CAD models are also assumed to be unavailable.

Given a pair of RGB images $\langle \mathcal{I}_Q, \mathcal{I}_R \rangle$, where $\mathcal{I}_Q$ denotes a query image captured by one robot and $\mathcal{I}_R$ denotes a reference image captured by another robot, the task is to estimate the relative pose $\xi = (\mathbf{R}_{QR}, \mathbf{t}_{QR}) \in \mathrm{SE}(3)$ of the object observed in $\mathcal{I}_Q$ with respect to the same object observed in $\mathcal{I}_R$. Here, $\mathbf{R}_{QR} \in SO(3)$ and $\mathbf{t}_{QR} \in \mathbb{R}^3$ represent the relative rotation and translation between the two object observations, respectively.

To formalise this setting, we introduce three coordinate systems: the object-centric coordinate frame $\mathcal{F}_O$, the reference camera frame $\mathcal{F}_{C_R}$ and the query camera frame $\mathcal{F}_{C_Q}$. Let $\mathbf{T}_R^O \in \mathrm{SE}(3)$ denote the rigid transformation from the object frame $\mathcal{F}_O$ to the reference camera frame $\mathcal{F}_{C_R}$ and $\mathbf{T}_Q^O \in \mathrm{SE}(3)$ denote the transformation from $\mathcal{F}_O$ to the query camera frame $\mathcal{F}_{C_Q}$. Although neither $\mathbf{T}_R^O$ nor $\mathbf{T}_Q^O$ is directly observable due to the absence of CAD models and absolute pose annotations, their relative transformation can be expressed as

$$\mathbf{T}_Q^R = \mathbf{T}_Q^O \left(\mathbf{T}_R^O\right)^{-1}, \tag{1}$$

which corresponds exactly to the desired relative pose ξ.

Under this formulation, the reference image $\mathcal{I}_R$ serves as a visual anchor acquired during prior observation, encoding implicit object-level spatial information. Once the relative pose $\mathbf{T}_Q^R$ is estimated, the object pose in the query camera frame can be consistently aligned with the reference observation. This property enables relative pose estimation to be naturally applicable to multi-robot collaborative perception scenarios, where object observations collected by one robot can be transferred and reused by another robot operating from a different viewpoint.

3.2 | Framework

Given a query image $\mathcal{I}_Q$ and a reference image $\mathcal{I}_R$ captured by different robots, the framework first identifies the target object and establishes coarse correspondences between the two observations. The resulting image pairs and keypoint correspondences are then fed into the proposed multiplex feature learning module, which jointly encodes appearance and geometric information. An attention-based feature aggregation module further facilitates cross-view feature interaction, and the aggregated features are finally used to regress the relative 6D pose ξ through dedicated rotation and translation heads. Figure 1 provides a structural overview of the proposed framework, whereas Algorithm 1 summarises the overall processing pipeline of MRPose. Details of each component are described in the following sections.

ALGORITHM 1 | MRPose: Multi-robot relative 6D object pose estimation

Input: Query image $\mathcal{I}_Q$, reference image $\mathcal{I}_R$
Output: Relative 6D object pose ξ
Image Pair Generation; // Section 3.3
  obtain segmented reference images $\left\{\mathcal{I}_{R_i}^{\mathrm{SAM}}\right\}_{i=1}^{k} = \mathrm{EfficientSAM}(\mathcal{I}_R)$;
  extract global features of $\mathcal{I}_Q$ and $\left\{\mathcal{I}_{R_i}^{\mathrm{SAM}}\right\}$ using DINOv2;
  retrieve the most similar segmented image $\mathcal{I}_S = \mathrm{argmax}_{\mathcal{I}_{R_i}^{\mathrm{SAM}}} \cos\left(\mathcal{I}_Q, \mathcal{I}_{R_i}^{\mathrm{SAM}}\right)$;
  extract patch-level features $\mathcal{F}_Q = \left\{\mathbf{f}_i^Q\right\}$ and $\mathcal{F}_S = \left\{\mathbf{f}_j^S\right\}$ from $\mathcal{I}_Q$ and $\mathcal{I}_S$;
  `// mutual NN in feature space`
  establish cross-view correspondences $\mathcal{C} = \mathrm{MNN}(\mathcal{F}_Q, \mathcal{F}_S)$;
Multiplex Feature Learning; // Section 3.4
  sample N matched patch pairs from $\mathcal{C}$;
  construct patch-level appearance representations $\mathbf{F}_Q$, $\mathbf{F}_S$ and $\mathbf{F}_\mathcal{I} = \mathrm{Concat}(\mathbf{F}_Q, \mathbf{F}_S)$;
  construct point-based geometric representation $\mathbf{F_P}$ from matched keypoints $\langle \mathbf{P}_Q, \mathbf{P}_S \rangle$;
  `// attention-based feature aggregation`
  apply cross-attention to aggregate appearance and geometric features, $\langle \mathbf{F}'_\mathcal{I}, \mathbf{F}'_\mathbf{P} \rangle = \mathrm{CrossAttn}(\mathbf{F}_\mathcal{I}, \mathbf{F_P})$;
  fuse the refined features by concatenation, $\mathbf{F} = \mathrm{Concat}(\mathbf{F}'_\mathcal{I}, \mathbf{F}'_\mathbf{P})$;
Pose Estimation; // Section 3.5
  predict relative rotation $\widehat{\mathbf{R}}$ using a rotation head;
  predict relative translation $\widehat{\mathbf{t}}$ by regressing t_z and recovering (t_x, t_y) from the 2D bounding box;
  form relative pose $\xi = (\widehat{\mathbf{R}}, \widehat{\mathbf{t}})$;
return ξ

3.3 | Image Pair Generation

The reference image $\mathcal{I}_R$ is first fed into a pretrained segmentation model, EfficientSAM [34], to obtain a set of segmented images $\left\{\mathcal{I}_{R_m}^{\mathrm{Seg}}\right\}_{m=1}^{k}$:

$$\left\{\mathcal{I}_{R_m}^{\mathrm{Seg}}\right\}_{m=1}^{k} = \mathrm{EfficientSAM}(\mathcal{I}_R), \tag{2}$$

where k denotes the number of segmented regions produced by the segmentation model for the given reference image. Since this number depends on the image content and the segmentation outcome, k may vary across different reference images. Here, m indexes the segmented candidate regions generated in the image-pair generation stage.

Then, using the feature extractor of DINOv2 [35], we extract the [CLS] token from $\mathcal{I}_Q$ and from each candidate segment in $\left\{\mathcal{I}_{R_m}^{\mathrm{Seg}}\right\}_{m=1}^{k}$ and denote the resulting global descriptors by $\mathbf{g}_Q$ and $\left\{\mathbf{g}_{R_m}^{\mathrm{Seg}}\right\}_{m=1}^{k}$, respectively. These global image features capture the

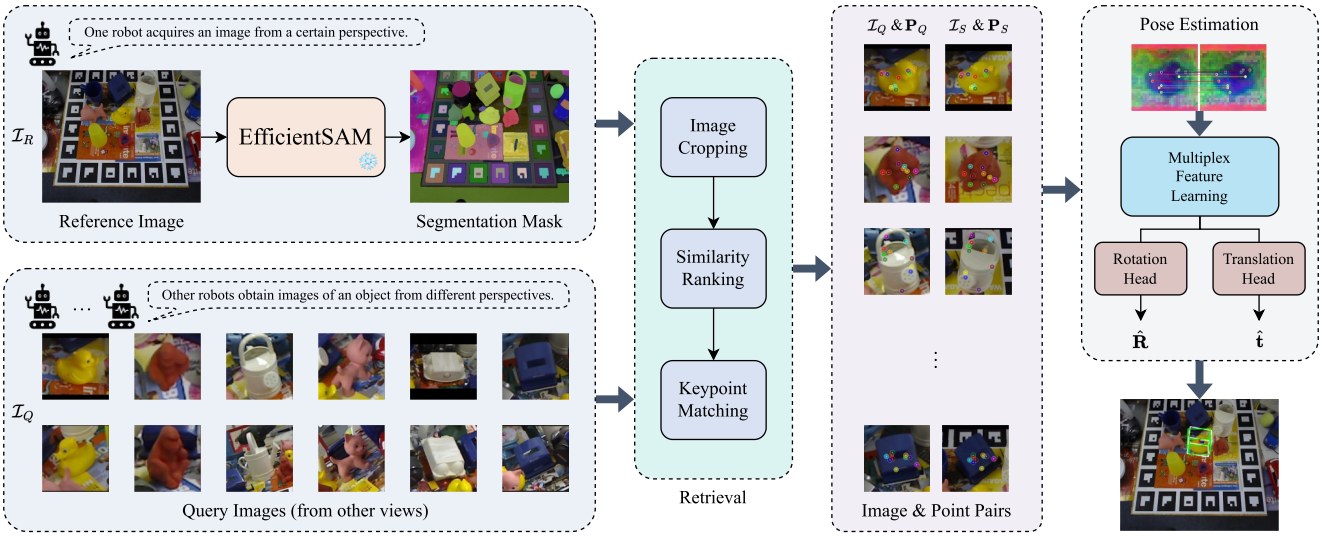


FIGURE 1 | Overview of MRPose. Given an RGB reference image and several RGB query images as input, MRPose first performs retrieval on the segmented reference and query images. This process identifies corresponding images and points of the specific object in the reference and query images, which are then fed into the multiplex feature learning module. Finally, the extracted features are directly input into the translation and rotation heads, enabling the model to compute the relative 6D pose.

overall properties and structure of the image. It should be noted that the DINOv2 representation in MRPose serves a dual role: Besides providing patch-level features for downstream correspondence extraction, its [CLS] token is also used as a global descriptor to retrieve, from the candidate segments in the full reference image, the crop corresponding to the query observation. We then calculate the cosine similarity between $\mathbf{g}_Q$ and each element in $\left\{\mathbf{g}_{R_m}^{\text{Seg}}\right\}_{m=1}^{k}$. The cosine similarity between $\mathbf{g}_Q$ and the mth candidate descriptor $\mathbf{g}_{R_m}^{\text{Seg}}$ is computed as follows:

$$\cos(\theta_m) = \frac{\mathbf{g}_Q \cdot \mathbf{g}_{R_m}^{\text{Seg}}}{\left\|\mathbf{g}_Q\right\| \cdot \left\|\mathbf{g}_{R_m}^{\text{Seg}}\right\|}. \tag{3}$$

By computing the cosine similarity for each candidate descriptor in $\left\{\mathbf{g}_{R_m}^{\text{Seg}}\right\}_{m=1}^{k}$, we obtain a similarity vector of dimension $1 \times k$, given by

$$\text{cosine_similarity} = \{\cos(\theta_m)\}_{m=1}^{k}. \tag{4}$$

Afterwards, based on the computed cosine similarity, select image $\mathcal{I}_S$ with the highest similarity to $\mathcal{I}_Q$ from $\left\{\mathcal{I}_{R_m}^{\text{Seg}}\right\}$. Similar to CNOS [33], this stage performs segmentation-based global retrieval, but here the retrieval is only from a nearby-view full reference image to obtain the crop corresponding to the query image.

Once a suitable image pair $\langle \mathcal{I}_Q, \mathcal{I}_S \rangle$ is retrieved, we extract patch-level features using the ninth transformer layer of DINOv2. Specifically, each image is decomposed into a set of patch tokens, yielding the patch feature sets:

$$\begin{aligned} \mathcal{F}_Q &= \text{DINOv2}(\mathcal{I}_Q) = \left\{\mathbf{f}_i^Q\right\}_{i=1}^{N_Q}, \\ \mathcal{F}_S &= \text{DINOv2}(\mathcal{I}_S) = \left\{\mathbf{f}_j^S\right\}_{j=1}^{N_S}, \end{aligned} \tag{5}$$

where $\mathbf{f}_i^Q$ and $\mathbf{f}_j^S$ denote the feature embeddings of the ith and jth patches in the $\mathcal{I}_Q$ and $\mathcal{I}_S$ respectively. Here, i and j denote the query-side and reference-side patch indices, respectively.

To establish reliable cross-view correspondences, we perform mutual nearest neighbour (MNN) matching in the DINOv2 feature space. The resulting correspondence set is defined as

$$\begin{aligned} \mathcal{C} &= \text{MNN}(\mathcal{F}_Q, \mathcal{F}_S), \\ &= \left\{(i,j) | i = \text{argmin}_{i'} \|\mathbf{f}_{i'}^Q - \mathbf{f}_j^S\|, j = \text{argmin}_{j'} \|\mathbf{f}_i^Q - \mathbf{f}_{j'}^S\|\right\}, \end{aligned} \tag{6}$$

which enforces mutual consistency between matched patch pairs and filters out ambiguous one-sided correspondences.

Equivalently, this process yields a correspondence set $\mathcal{C} = \{(i,j)\}$, where each element denotes a mutually matched patch pair between $\mathcal{I}_Q$ and $\mathcal{I}_S$. Let $K = |\mathcal{C}|$ denote the number of matched patch pairs obtained from MNN matching, which may vary across different image pairs. This mutual consistency constraint effectively filters out ambiguous one-sided correspondences and provides a sparse yet reliable set of cross-view matches.

Given $\mathcal{C}$, we map each matched patch index pair (i,j) to the pixel coordinates of the corresponding patch centres, forming the keypoint set $\langle \mathbf{P}_Q, \mathbf{P}_S \rangle \in \mathbb{R}^{n \times 2}$. Figure 2 illustrates an example of the resulting patch-level correspondences.

3.4 | Multiplex Feature Learning

In traditional methods, the PnP algorithm can be directly used to solve object pose when accurate correspondences and camera parameters are available. However, in our setting, the viewpoints are unknown and potentially heterogeneous, making classical geometric solvers inapplicable. Therefore, we adopt a learning-based approach to directly estimate the relative pose.

This module takes as input the outputs of the image pair generation module, including the retrieved image pair $\langle \mathcal{I}_Q, \mathcal{I}_S \rangle$ and the matched keypoint pairs $\langle \mathbf{P}_Q, \mathbf{P}_S \rangle \in \mathbb{R}^{n\times 2}$, which are processed through two complementary data streams encoding appearance and geometric information.

The detailed architecture of the multiplex feature learning module, together with the subsequent attention-based feature aggregation and pose prediction heads, is illustrated in Figure 3.

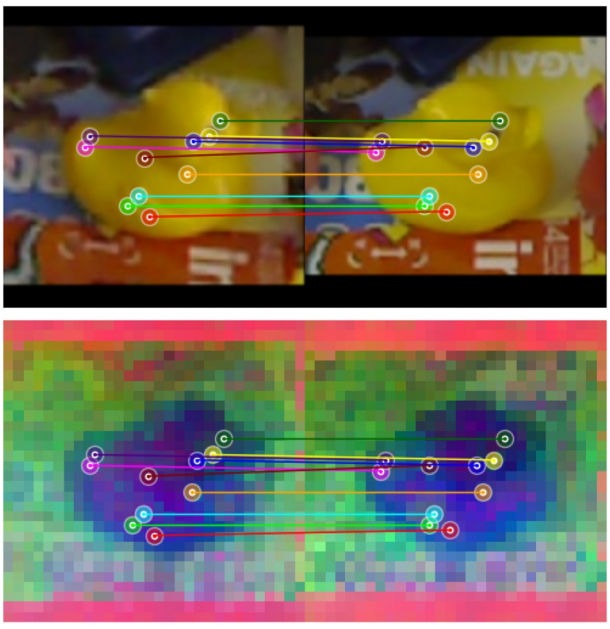


FIGURE 2 | Visualisation of keypoint matching. Top are the matched $\mathcal{I}_Q$ and $\mathcal{I}_S$ and bottom are the features extracted by DINOv2 after PCA dimensionality reductions.

For the image pair $\langle \mathcal{I}_Q, \mathcal{I}_S \rangle$, we first identify the DINOv2 patch tokens corresponding to the matched keypoints obtained from Equation (6). Since the number of matched patches K varies across image pairs, we randomly sample N patches from these candidates to construct a fixed-length representation. This strategy preserves keypoint-supporting regions that encode cross-view correspondence while avoiding unnecessary computation over all patch embeddings. After MNN matching, the proposed pipeline directly regresses the relative pose and therefore does not include an additional RANSAC-based geometric fitting stage. Under uncalibrated and potentially heterogeneous viewpoints, standard residuals such as reprojection error are not directly available for reliable RANSAC filtering. Robustness to noisy correspondences is instead provided by the combination of bidirectional MNN filtering, random correspondence sampling and the subsequent attention-based aggregation module, which adaptively suppresses inconsistent matches.

Given the correspondence set $\mathcal{C} = \{(i,j)\}$, we construct correspondence-aligned patch feature subsets by indexing the original patch feature sets $\mathcal{F}_Q$ and $\mathcal{F}_S$:

$$\begin{aligned} \mathcal{F}_Q^{\mathcal{C}} &= \left\{ \mathbf{f}_i^Q | (i,j) \in \mathcal{C} \right\}, \\ \mathcal{F}_S^{\mathcal{C}} &= \left\{ \mathbf{f}_j^S | (i,j) \in \mathcal{C} \right\}. \end{aligned} \tag{7}$$

These features are concatenated along the feature dimension to form fixed-length appearance representations,

$$\begin{aligned} \mathbf{F}_Q &= \mathrm{Concat}\left(\mathcal{F}_Q^{\mathcal{C}}\right), \\ \mathbf{F}_S &= \mathrm{Concat}\left(\mathcal{F}_S^{\mathcal{C}}\right), \end{aligned} \tag{8}$$

and further combined into an image-level feature:

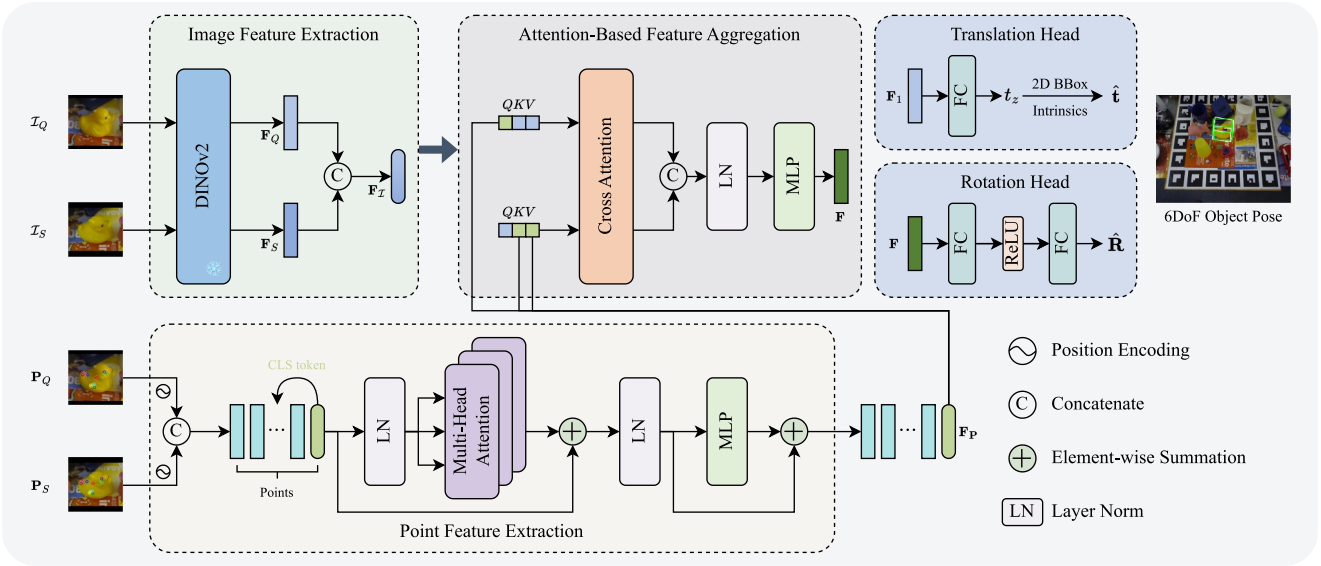


FIGURE 3 | Structure of the pose estimation network. Two input images from the previous module are processed by a pretrained DINOv2 to extract feature vectors, which are then concatenated to form $\mathbf{F}_{\mathcal{I}}$. Simultaneously, two sets of input points from the previous module undergo position encoding and are processed through layers, including multi-head attention and MLP layer, to produce the point feature vector $\mathbf{F}_{\mathbf{P}}$. Cross-attention is then applied to $\mathbf{F}_{\mathcal{I}}$ and $\mathbf{F}_{\mathbf{P}}$, and the results $\mathbf{F}$ are fed into the rotation head. $\mathbf{F}_1$ is fed into the translation head. The outputs of these two prediction heads together constitute the 6D pose of the object.

$$\mathbf{F}_{\mathcal{I}} = \mathrm{Concat}(\mathbf{F}_Q, \mathbf{F}_S). \tag{9}$$

In parallel, the matched keypoints provide complementary geometric cues for accurate relative pose estimation. Using the same N correspondences sampled in the image branch, each matched pair is represented by concatenating the 2D pixel coordinates of the query and reference keypoints, that is, $\left(x_i^Q, y_i^Q, x_i^S, y_i^S\right)$. Stacking these coordinate tuples yields $\mathbf{P}_{\text{sampled}} \in \mathbb{R}^{N\times 4}$. The 4D coordinate tuple of each matched pair is then mapped by a standard sine-cosine positional encoding to $\mathbf{P}_{\text{embedded}} \in \mathbb{R}^{N\times C}$, where C denotes the embedding dimension of the geometric token. These embedded tokens are processed by a transformer, treating each embedded point as an independent token. The resulting tokens are flattened and passed through an MLP to produce the geometric feature representation $\mathbf{F_P}$.

The module is implemented as a standard multi-head cross-attention block. Specifically, $\mathbf{F}_{\mathcal{I}}$ and $\mathbf{F_P}$ are alternately used as queries and key–value pairs in a bidirectional cross-attention mechanism: $\mathbf{F}_{\mathcal{I}}$ queries $\mathbf{F_P}$ to inject geometric cues into the image branch, whereas $\mathbf{F_P}$ queries $\mathbf{F}_{\mathcal{I}}$ to inject appearance cues into the geometric branch. The two aggregated outputs are then concatenated for subsequent pose regression. When $\mathbf{F}_{\mathcal{I}}$ is used as the query, the aggregated feature is given by

$$\mathbf{F}'_{\mathcal{I}} = \mathrm{Softmax}\left(\frac{(\mathbf{W}_Q\mathbf{F}_{\mathcal{I}})(\mathbf{W}_K\mathbf{F_P})^T}{\sqrt{d_k}}\right)(\mathbf{W}_V\mathbf{F_P}), \tag{10}$$

When $\mathbf{F_P}$ is used as the query, the aggregated feature is given by

$$\mathbf{F}'_{\mathbf{P}} = \mathrm{Softmax}\left(\frac{(\mathbf{W}_Q\mathbf{F_P})(\mathbf{W}_K\mathbf{F}_{\mathcal{I}})^T}{\sqrt{d_k}}\right)(\mathbf{W}_V\mathbf{F}_{\mathcal{I}}), \tag{11}$$

Finally, the aggregated appearance and geometric features are concatenated to form the fused representation:

$$\mathbf{F} = \mathrm{Concat}(\mathbf{F}'_{\mathcal{I}}, \mathbf{F}'_{\mathbf{P}}), \tag{12}$$

which captures both cross-view appearance correspondence and spatial geometric relationships. This fused feature $\mathbf{F}$ serves as the input to the subsequent pose estimation module.

3.5 | Pose Estimation

Our goal is to predict the full 6D pose, adopting different prediction strategies for rotation and translation.

For rotation, we avoid using traditional PnP algorithms and instead let the neural network directly learn the rotation from the multiplex features $\mathbf{F}$ obtained in the previous stage. To this end, we design a rotation prediction head composed of an MLP with two linear layers and ReLU activation functions. This head directly outputs the relative rotation between the query and reference images, represented using a continuous 6D rotation representation [36].

For translation, we follow common practice in RGB-based pose estimation [5, 37] and decouple the estimation along the camera viewing direction from the in-plane components. Specifically, the translation head directly regresses the relative displacement t_z along the camera z-axis between the query and reference views using the reference-image patch features. The remaining in-plane components (t_x, t_y) are then recovered from the 2D object location in the image plane via camera intrinsics. Together, (t_x, t_y, t_z) forms the predicted relative translation $\hat{\mathbf{t}}$ yielding the full relative translation between the two views.

This modular approach allows the rotation and translation heads to specialise in their respective tasks. The rotation head is dedicated to accurately capture the object's angular orientation, whereas the translation head focuses on estimating its spatial displacement. By leveraging the feature information extracted in the earlier stages of the pipeline, this design ensures precise and reliable object 6D pose estimation. Consequently, this approach represents a significant improvement over monolithic estimation strategies, offering more precise and reliable pose predictions in real-world applications.

3.6 | Loss Functions

Given the ground-truth translation along the camera viewing direction $t_{z,\text{gt}}$ and the predicted translation $\hat{t}_z$, the translation loss is defined as

$$L_{\text{trans}} = \left|\hat{t}_z - t_{z,\text{gt}}\right|. \tag{13}$$

For rotation, both the prediction and supervision are defined in the 6D rotation space. Specifically, the network predicts a 6D rotation representation $\hat{\mathbf{R}} \in \mathbb{R}^6$, and the ground-truth rotation is also represented as $\mathbf{R}_{\text{gt}} \in \mathbb{R}^6$. The rotation loss is defined as

$$L_{\text{rot}} = \left\|\hat{\mathbf{R}} - \mathbf{R}_{\text{gt}}\right\|_2. \tag{14}$$

The overall objective is given by:

$$L = \lambda_1 L_{\text{trans}} + \lambda_2 L_{\text{rot}}, \tag{15}$$

where λ_1 and λ_2 are weighting coefficients.

4 | Experiments and Discussions

4.1 | Datasets

In this paper, the two most commonly used datasets in the field, Linemod [38] and YCB-Video (YCB-V) [39], were utilised to evaluate the performance of the proposed method. Specifically, to ensure a fair comparison, we followed the setup of POPE and selected the same image pairs from these datasets.

4.2 | Implementation Details

When training MRPose on the two datasets mentioned earlier, we used the Adam optimiser for 500 epochs with a learning rate of 1e-5, weight decay of 1e-5 and batch size of 8, executed on a single GeForce RTX 3090. The loss weights were set to $\lambda_1 = 10$ and $\lambda_2 = 1$ to keep the loss functions in the same order of magnitude. During the feature extraction and keypoint matching stage, we used the pretrained *dinov2_vits14* as DINOv2 backbone. For each image pair, the number of keypoint pairs N and the encoding dimension of the point coordinates L were set to 10 and 1024, respectively.

It should be noted that POPE does not involve a training phase. Although our method includes a training process, its objective remains the generalisable estimation of unseen object poses, which fundamentally differs from traditional instance-level pose estimation. Therefore, in all the experiments below, our method adopts a cross-training and testing strategy: training on YCB-V and testing on Linemod and vice versa. This ensures that the objects encountered during testing never appear in the training set.

4.3 | Evaluation Metrics

We report the 15° accuracy, 30° accuracy and median error for rotation, as well as the median error for translation, following the standard practice for relative object pose estimation [40]. The accuracy metrics indicate the percentage of estimates with errors below these thresholds, whereas the median error reports the median of the errors between all predictions and the ground truth.

In addition, our method is capable of predicting translation, and ADD-(S) [38] is also reported in the experimental section, which considers the error between point clouds of objects, with a specific focus on symmetric objects. The calculation formula is as follows:

$$\text{ADD-(S)} = \begin{cases} \frac{1}{|M|}\sum_{x_1 \in M} \min_{x_2 \in M} \left\| \left(\tilde{R}x_1 + \tilde{t}\right) - (Rx_2 + t) \right\|, & \text{if symmetry,} \\ \frac{1}{|M|}\sum_{x \in M} \left\| \left(\tilde{R}x + \tilde{t}\right) - (Rx + t) \right\|, & \text{otherwise,} \end{cases} \tag{16}$$

where M denotes the set of 3D model points.

4.4 | Comparison With State-of-the-Art Methods

Our method takes only RGB images as inputs and estimates the relative pose between different viewpoints. We evaluate MRPose on the Linemod and YCB-Video datasets using standard rotation and translation metrics commonly adopted in relative pose estimation. The quantitative results are summarised in Table 1.

As shown in Table 1, MRPose consistently outperforms existing methods on both datasets across all reported metrics. In particular, on the Linemod dataset, MRPose improves Acc30 for rotation by 22.9% compared to the strongest baseline. These results demonstrate the robustness of our approach under challenging conditions such as cluttered backgrounds, partial occlusions and large viewpoint variations.

Although MRPose is designed to predict relative pose between image pairs, the absolute object pose in the camera coordinate system can be recovered by composing the estimated relative pose with the pose of the reference image. Based on this observation, we further evaluate our method under the ADD-(S) metric and compare it with instance-level and reference-view-based pose estimation methods, as reported in Table 2. Among these methods, PVNet is supervised and trained with dataset-specific annotations, whereas POPE estimates only relative rotation and cannot predict translation. Following common practice, methods marked with * additionally use the ground-truth translation of the query image to enable a fair comparison.

Under this evaluation protocol, MRPose achieves consistently better ADD-(S) performance than POPE across most object categories and remains competitive with instance-level methods, despite not being trained on the Linemod dataset. These results further validate the strong generalisation capability of MRPose and its effectiveness for full 6D pose estimation without relying on object-specific training data.

We note that POPE remains slightly stronger on a few categories in the fair-comparison rows marked with *, such as ape, duck, eggbox and iron. We attribute this mainly to POPE's explicit correspondence-and-geometry pipeline with LoFTR-based local matching and geometric post-processing, which can be advantageous when pose estimation depends heavily on very precise local correspondences or when object appearance is more ambiguous. By contrast, MRPose adopts an end-to-end regression formulation, which yields better overall average performance, but may be less favourable in a few such cases.

4.5 | Ablation Study

4.5.1 | Effectiveness of Feature and Keypoint Pair Interactive Learning

The coordinates of keypoints and image features both contain information relevant to object rotation. To further analyse their

TABLE 1 | Comparison with other methods on the Linemod and YCB-video datasets.

		Metric			
Dataset	**Method**	**R: Acc15 (↑)**	**R: Acc30 (↑)**	**R: Median. Err (↓) (degree)**	**t: Median. Err (↓) (cm)**
Linemod	Gen6D [31]	0.096	0.364	—	—
	LoFTR [41]	0.324	0.562	—	—
	POPE [32]	0.483	0.770	15.531	—
	Ours	**0.568**	**1.000**	**13.594**	5.484
YCB-video	Gen6D [31]	0.077	0.232	—	—
	LoFTR [41]	0.478	0.686	—	—
	POPE [32]	0.544	0.801	14.302	—
	Ours	**0.612**	**1.000**	**11.849**	6.260

Note: Best results in each dataset block are highlighted in bold. Translation error is reported whenever available in the source comparison; a — indicates that the corresponding translation result is not reported in the POPE paper under the evaluation setting considered here.

TABLE 2 | Comparison with other methods on the Linemod dataset under the ADD-(S) metric.

Method	Data-specific training	Reference views	All categories												
			Ape	Benchvise	Camera	Can	Cat	Driller	Duck	Eggbox	Glue	Holepuncher	Iron	Lamp	Phone	Avg
BB8	✓	—	27.9	62.0	40.1	48.1	45.2	58.6	32.8	40.0	27.0	42.4	67.0	39.9	35.2	43.6
Tekin	✓	—	21.6	81.8	36.6	68.8	41.8	63.5	27.2	69.6	80.0	42.6	75.0	71.1	47.7	56.0
Self6D	✓	—	38.9	75.2	36.9	65.6	57.9	67.0	19.6	99.0	94.1	15.5	77.9	68.2	50.1	58.9
PVNet	✓	—	43.7	99.9	86.7	95.5	79.3	96.4	52.6	99.2	95.7	81.9	98.9	99.3	92.4	86.3
Gen6D	✗	200	—	62.1	45.6	—	40.9	48.8	16.2	—	—	—	—	—	—	—
Onepose	✗	200	11.8	92.6	**88.1**	77.2	47.9	74.5	34.2	71.3	37.5	54.9	89.2	87.6	60.6	63.6
Onepose++	✗	200	31.2	**97.3**	88.0	**89.8**	70.4	**92.5**	42.3	**99.7**	48.0	69.7	**97.4**	**97.8**	**76.0**	**76.9**
POPE*	✗	1	**58.1**	67.8	66.8	66.4	68.5	68.9	**56.3**	74.1	64.1	65.4	81.3	73.9	63.1	67.3
Ours*	✗	1	57.1	83.7	69.4	71.7	**86.0**	84.9	47.1	62.5	**78.4**	**70.9**	75.7	75.0	64.1	71.3
Ours	✗	1	55.1	83.7	62.9	69.6	84.2	86.8	47.1	58.3	**78.4**	65.5	73.0	75.0	61.5	69.3

Note: Methods marked with * additionally use the ground-truth translation of the query image to enable a fair comparison. Among methods without data-specific training (the lower block), the best and second-best results in each column are highlighted in bold and underlined, respectively.

roles in pose regression and validate the effectiveness of our proposed attention-based feature aggregation module, we designed a set of comparative experiments using different features as inputs to the rotation prediction head. The features considered include image features only (OI), keypoint coordinates only (OP), concatenated image features and keypoints (IP-Cat.) and attention-based feature aggregation (IP-Attn.). These experiments were evaluated using the same metrics as in the previous section on the Linemod dataset, with results shown in Table 3. The results indicate that IP-Attn. achieves the best performance under most conditions, demonstrating that our aggregation strategy effectively integrates the two modalities, thereby enhancing the accuracy of pose estimation through their complementary contributions.

4.5.2 | *Number of Keypoints*

In our method, the number of keypoints N extracted from image feature patches is clearly an important hyperparameter. To investigate its impact, we trained models with different values of N and evaluated them on the Linemod dataset using the Acc15 metric. The results, shown in Figure 4, indicate that accuracy improves as the number of keypoints increases, eventually stabilising around $N = 10$. Based on this observation, we select $N = 10$ as a trade-off between accuracy and computational cost.

4.6 | Runtime Analysis

To evaluate the computational efficiency of the proposed method in practical applications, we conduct a detailed runtime analysis of the entire inference pipeline. Table 4 reports the average runtime of MRPose and POPE at each key stage. All experiments are conducted under the same hardware configuration to ensure a fair comparison.

During feature extraction, MRPose and POPE use the same DINOv2 backbone in our implementation and therefore have the same measured feature-extraction runtime. For segmentation, both methods use category-agnostic segmentation as a front end, but the measured runtime differs because the two pipelines invoke this stage with different processing strategies under our implementation: POPE performs a more expensive prompt/proposal generation process, whereas MRPose uses the segmentation output only to construct the reference crop for the query–reference image pair. Thus, the segmentation numbers in Table 4 should be interpreted as end-to-end implementation-level timings rather than as a comparison of different segmentation backbones. In the matching stage, MRPose is significantly faster than POPE. POPE relies on dense feature matching and geometric constraint solving, which leads to a high computational cost. In contrast, MRPose performs matching only on a small set of selected keypoint patches, substantially reducing the matching overhead.

For pose estimation, POPE requires additional geometric optimisation, resulting in relatively high latency, whereas MRPose directly regresses the relative pose using a lightweight neural network. As a result, the pose estimation stage of MRPose incurs negligible computational cost, requiring only approximately

TABLE 3 | Ablation study on feature selection and fusion strategies for relative pose estimation on Linemod and YCB-video.

Metric	Linemod				YCB-video			
	OI	OP	IP-Cat.	IP-Attn.	OI	OP	IP-Cat.	IP-Attn.
R: Acc15 (↑)	0.496	0.558	0.562	**0.568**	0.609	0.605	0.611	**0.612**
R: Acc30 (↑)	0.932	0.924	0.951	**1.000**	0.801	**1.000**	**1.000**	**1.000**
R: Median. Err (↓) (degree)	13.824	15.175	13.679	**13.594**	12.094	12.109	11.973	**11.849**
t: Median. Err (↓) (cm)	5.571	5.508	5.498	**5.484**	6.518	6.519	6.315	**6.260**

Note: Bold values indicate the best result(s) for each metric on each dataset.

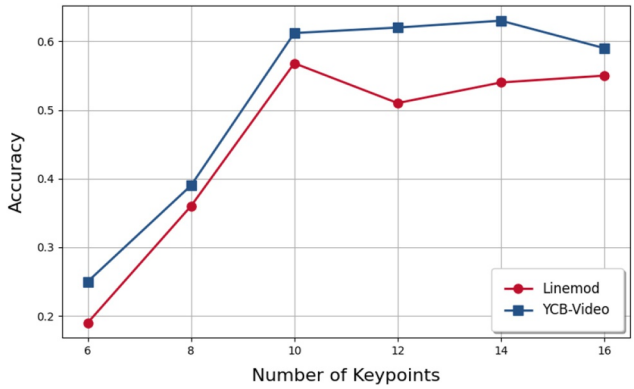


FIGURE 4 | On the Linemod dataset, the metric Acc15 changes with the number of keypoints N.

TABLE 4 | Runtime analysis of different methods.

Method	Runtime per stage (ms)				
	Seg.	Feat. Ext.	Matching	Pose estimation	Total
POPE	901.92	25.56	166.22	121.73	1215.43
Ours	26.86	25.56	70.24	0.09	122.75

0.09 ms. Overall, under our current implementation and evaluation protocol, MRPose shows a lower reported total runtime while maintaining high estimation accuracy, suggesting its potential for multi-robot collaborative scenarios where computational efficiency is important.

4.7 | Qualitative Results

To further demonstrate the effectiveness of our method, Figure 5 presents a comparison of the visual results on two datasets. It is evident that our method can estimate the object's pose more accurately than POPE. In challenging scenarios (such as the driller in Figure 5 and the c_box in Figure 6), the pose results predicted by POPE show significant deviations from the ground truth poses. In contrast, our method effectively estimates the 6D object poses, which convincingly demonstrates the effectiveness of our approach.

Despite the strong performance observed in most scenarios, our method may fail when the viewpoint difference between the query and reference images becomes excessively large. As illustrated in Figure 7, extreme viewpoint changes lead to significant appearance variation and limited overlap of discriminative visual patterns, which in turn degrades the quality of patch-level correspondences. In such cases, the mutual nearest neighbour matching becomes unreliable, resulting in insufficient or inaccurate cross-view feature associations and consequently erroneous relative pose estimation. This failure mode highlights an inherent limitation of appearance-based relative pose estimation methods under severe viewpoint discrepancies.

A current limitation concerns multi-instance scenes such as bin-picking setups [42, 43]: If multiple visually similar or identical

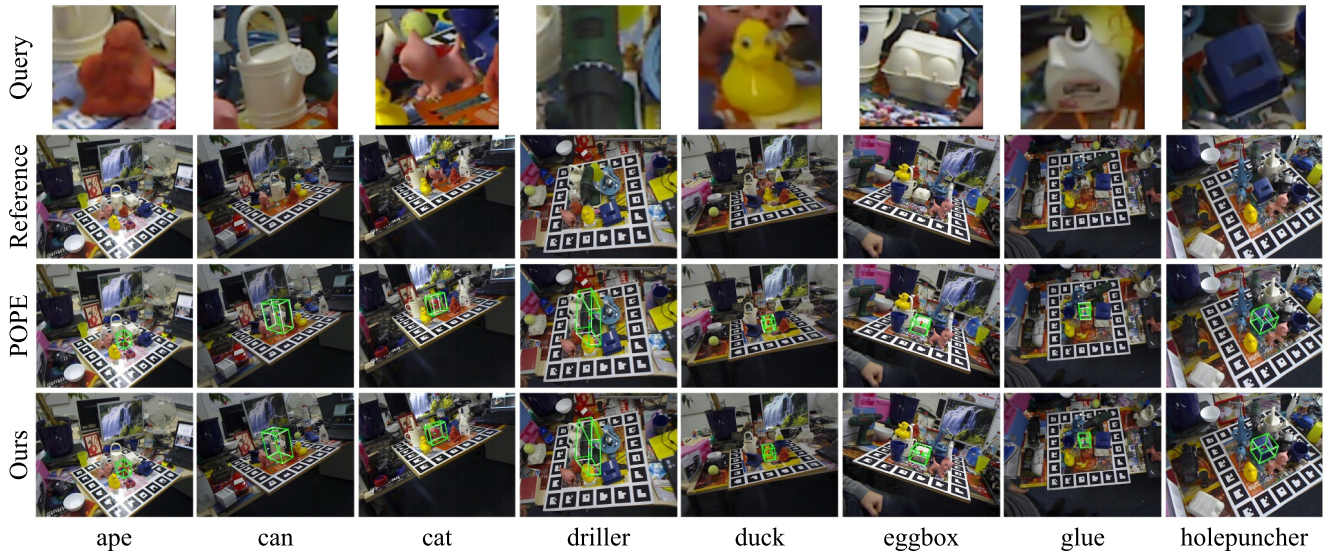


FIGURE 5 | Visualisation results of our method on the Linemod dataset. For each reference image, the object pose is visualised by composing the ground-truth pose of the query image with the relative pose estimated by MRPose, enabling consistent cross-view pose projection. In the visualisation results, green bounding boxes indicate the ground-truth object poses, whereas white bounding boxes denote the poses predicted by our model. The reference-view visualisation is obtained without using ground-truth pose annotations of the reference image.

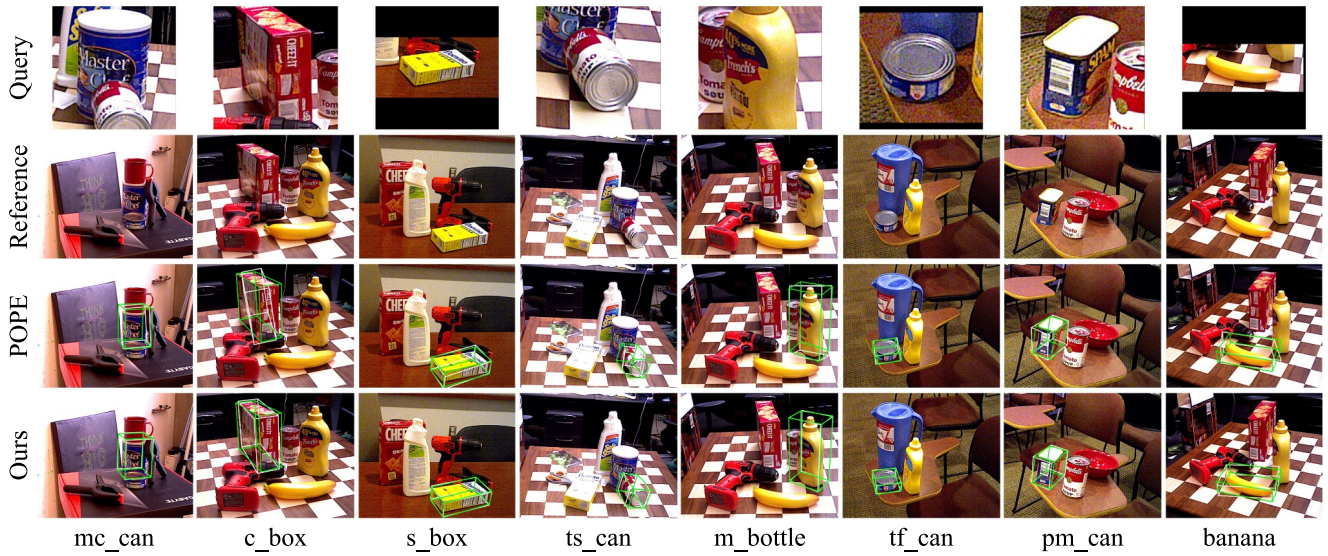


FIGURE 6 | Visualisation results of our method on the YCB-video dataset. For each reference image, the object pose is visualised by composing the ground-truth pose of the query image with the relative pose estimated by MRPose, allowing the object pose to be projected consistently across different viewpoints. In the visualisation results, green bounding boxes indicate the ground-truth object poses, whereas white bounding boxes denote the poses predicted by our model. The reference-view visualisation is obtained without using ground-truth pose annotations of the reference image.

instances appear in the reference image, the DINOv2-based retrieval in image-pair generation may associate the query with the wrong instance. This ambiguity occurs before pose estimation and will be addressed in future work through more explicit instance disambiguation.

4.8 | Real World Results

To validate the practicality of MRPose in real-world multi-robot scenarios, we conduct experiments using a dual-robot setup based on two DashGo D1 mobile robots, as illustrated in Figure 8a. To emulate heterogeneous robotic platforms, one robot is equipped with an Intel RealSense D435I camera, whereas the other uses an iPhone 16 Pro mobile phone camera. The two robots independently observe the same target object from different viewpoints, with an average relative viewpoint difference of approximately 30°, reflecting realistic cross-robot perception conditions. The captured RGB images are transmitted between the two robots through direct communication between their onboard control computers over the same local area network (LAN), allowing us to measure the end-to-end data transmission latency under practical network conditions without relying on shared calibration or synchronised sensing. For the heterogeneous-camera setup, the

Intel RealSense D435I intrinsics are obtained from the device interface, whereas the iPhone intrinsics are read during image acquisition through ARKit.

In this setting, the image captured by one robot is treated as the reference view, which serves as the canonical observation of the target object, whereas the image from the other robot is used as the query. For visualisation, the estimated relative transformation is applied to the object coordinate system associated with the reference view, resulting in a pose representation that appears as an absolute pose in the common reference frame. In our experiments, the average data transmission time between the two robots is measured to be 0.1395 s, indicating that the proposed pipeline can operate under realistic communication latency.



FIGURE 7 | Failure cases under extreme viewpoint changes. Each group shows a failure example. The first row presents the $\mathcal{I}_Q$ (a) and the selected $\mathcal{I}_S$ (b). The second row shows the pose estimation results, where green boxes denote ground-truth poses and white boxes indicate the predicted poses. Large viewpoint differences cause limited visual overlap and unreliable patch-level correspondences, leading to inaccurate relative pose estimation.

Qualitative results are shown in Figure 8b. Despite the viewpoint discrepancy and heterogeneous sensing conditions, MRPose produces stable and accurate relative pose estimates, leading to consistent geometric alignment across different object instances. These results demonstrate that MRPose can robustly handle cross-robot appearance variations and real-world communication delays, supporting its applicability in practical multi-robot collaborative perception systems.

5 | Conclusion

This paper presents MRPose, a unified framework for multi-robot relative 6D object pose estimation that operates without CAD models or pose annotations. By adopting a query–reference paradigm, MRPose enables robots with heterogeneous sensors to estimate relative object poses from independently acquired RGB observations, without shared calibration or synchronised sensing. The proposed multiplex feature learning and attention-based aggregation allow accurate and robust pose prediction without reliance on geometric solvers or handcrafted correspondence pipelines.

Extensive benchmark evaluations demonstrate consistent improvements over existing RGB-based relative pose estimation methods and strong generalisation to unseen objects. Real-world multi-robot experiments further confirm the practicality of

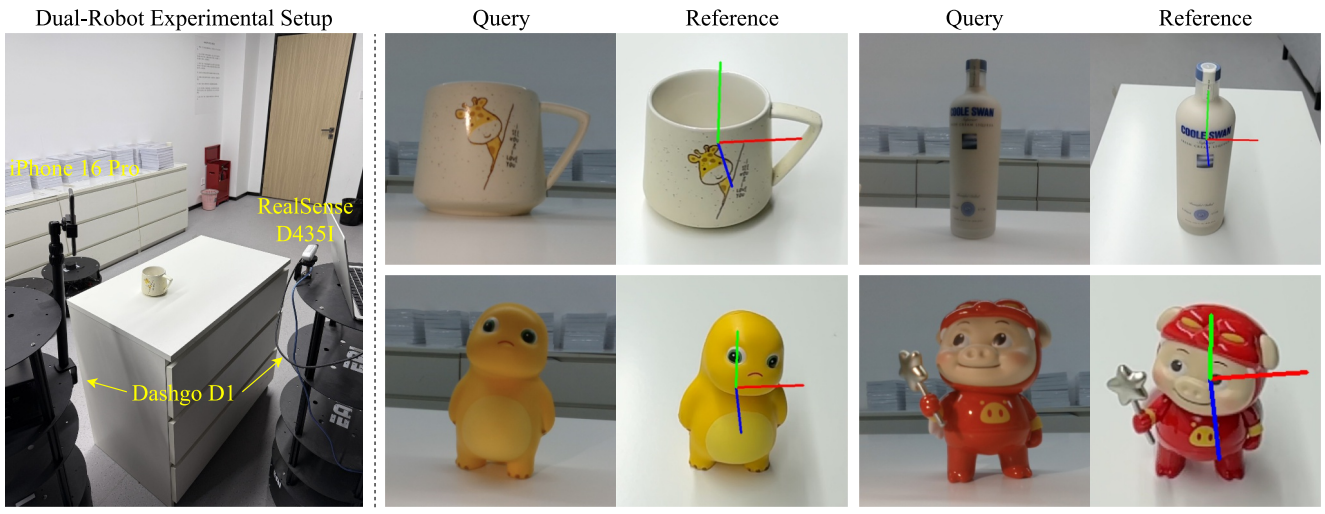


FIGURE 8 | Real-world experimental results of the proposed MRPose framework. (a) A dual-robot experimental setup, where the left robot is equipped with a mobile phone camera and the right robot uses an Intel RealSense camera to capture RGB images of the same target object from different viewpoints. (b) Qualitative visualisation of relative pose estimation results between the two robot observations. The estimated object poses are visualised using 3D coordinate axes, illustrating the relative pose consistency across different viewpoints without relying on ground-truth annotations.

MRPose under heterogeneous sensing and realistic communication latency, highlighting collaborative multi-view perception as a promising paradigm for deployable robot vision systems in unstructured environments.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62373009).

Conflicts of Interest

Hong Liu is an Executive Editor-in-Chief for the journal, and was not involved in peer review process or the decision to publish this article. The authors declare that they have no conflict of interest.

Data Availability Statement

The data that support the findings of this study are openly available in https://bop.felk.cvut.cz/challenges/, reference number [33, 34].

References

1. S. Wang, M. N. Nikolić, T. L. Lam, Q. Gao, R. Ding, and T. Zhang, "Robot Manipulation Based on Embodied Visual Perception: A Survey," *CAAI Transactions on Intelligence Technology* 10, no. 4 (2025): 945–958, https://doi.org/10.1049/cit2.70022.

2. Y. Chen, S. Huang, T. Yuan, S. Qi, Y. Zhu, and S.-C. Zhu, "Holistic++ Scene Understanding: Single-View 3d Holistic Scene Parsing and Human Pose Estimation With Human-Object Interaction and Physical Commonsense," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), 8648–8657.

3. Y. Zhao, S. Zhang, W. Zhao, Y. Wei, and J. Peng, "Augmented Reality System Based on Real-Time Object 6d Pose Estimation," in *Proceedings of the 2023 2nd International Conference on Image Processing and Media Computing (ICIPMC)* (IEEE, 2023), 27–34.

4. Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A Convolutional Neural Network for 6d Object Pose Estimation in Cluttered Scenes," preprint arXiv:1711.00199 (2017).

5. W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "Ssd-6d: Making Rgb-Based 3d Detection and 6d Pose Estimation Great Again," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2017), 1521–1529.

6. C. Song, J. Song, and Q. Huang, "Hybridpose: 6D Object Pose Estimation Under Hybrid Representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 431–440.

7. Y. Bukschat and M. Vetter, "Efficientpose: An Efficient, Accurate and Scalable End-to-End 6d Multi Object Pose Estimation Approach," preprint arXiv:2011.04307 (2020).

8. H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized Object Coordinate Space for Category-Level 6d Object Pose and Size Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), 2642–2651.

9. K. Chen and Q. Dou, "Sgpa: Structure-Guided Prior Adaptation for Category-Level 6d Object Pose Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2021), 2773–2782.

10. D. Chen, J. Li, Z. Wang, and K. Xu, "Learning Canonical Shape Space for Category-Level 6d Object Pose and Size Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 11973–11982.

11. J. Lin, Z. Wei, Z. Li, S. Xu, K. Jia, and Y. Li, "Dualposenet: Category-Level 6d Object Pose and Size Estimation Using Dual Pose Network With Refined Learning of Pose Consistency," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)* (2021), 3560–3569.

12. Z. Fan, Z. Song, Z. Wang, et al., "Acr-Pose: Adversarial Canonical Representation Reconstruction Network for Category Level 6d Object Pose Estimation," in *Proceedings of the 2024 International Conference on Multimedia Retrieval (ICMR)* (2024), 55–63.

13. T. Hou, A. Ahmadyan, L. Zhang, J. Wei, and M. Grundmann, "Mobilepose: Real-Time Pose Estimation for Unseen Objects With Weak Shape Supervision," preprint arXiv:2003.03522 (2020).

14. M. Tian, M. H. Ang, and G. H. Lee, "Shape Prior Deformation for Categorical 6d Object Pose and Size Estimation," in *Proceedings of the European Conference on Computer Vision (ECCV)* (2020), 530–546.

15. A. S. Periyasamy, A. Amini, V. Tsaturyan, and S. Behnke, "Yolopose v2: Understanding and Improving Transformer-Based 6d Pose Estimation," *Robotics and Autonomous Systems* 168 (2023): 104490, https://doi.org/10.1016/j.robot.2023.104490.

16. Z. Li, G. Wang, and X. Ji, "Cdpn: Coordinates-Based Disentangled Pose Network for Real-Time Rgb-Based 6-dof Object Pose Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), 7678–7687.

17. A. Amini, A. S. Periyasamy, and S. Behnke, "T6d-Direct: Transformers for Multi-Object 6d Pose Direct Regression," in *Proceedings of the DAGM German Conference on Pattern Recognition (GCPR)* (Springer, 2021), 530–544.

18. M. Zheng, P. Gao, R. Zhang, et al., "End-to-End Object Detection With Adaptive Clustering Transformer," preprint arXiv:2011.09315 (2020).

19. A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," in *Neural Information Processing Systems (NIPS)* (2017), 5998–6008.

20. J. Chen, Z. Zhou, M. Sun, et al., "Zeropose: Cad-Prompted Zero-Shot Object 6d Pose Estimation in Cluttered Scenes," *IEEE Transactions on Circuits and Systems for Video Technology* 35, no. 2 (2024): 1251–1264, https://doi.org/10.1109/TCSVT.2024.3482439.

21. J. Lin, L. Liu, D. Lu, and K. Jia, "Sam-6d: Segment Anything Model Meets Zero-Shot 6d Object Pose Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 27906–27916.

22. Y. He, Y. Wang, H. Fan, J. Sun, and Q. Chen, "Fs6d: Few-Shot 6d Pose Estimation of Novel Objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 6814–6824.

23. B. Wen, W. Yang, J. Kautz, and S. Birchfield, "Foundationpose: Unified 6d Pose Estimation and Tracking of Novel Objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 17868–17879.

24. T. Nguyen, M. N. Bui, C. Nguyen, et al., "Nope: Novel Object Pose Estimation From a Single Image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024).

25. X. Liu, G. Wang, R. Zhang, C. Zhang, F. Tombari, and X. Ji, "Unopose: Unseen Object Pose Estimation With an Unposed rgb-d Reference Image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025), 22023–22034.

26. T. Lee, B. Wen, M. Kang, G. Kang, I. S. Kweon, and K.-J. Yoon, "Any6d: Model-Free 6d Pose Estimation of Novel Objects," in

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025), 11633–11643.

27. V. Leroy, Y. Cabon, and J. Revaud, "Grounding Image Matching in 3d With mast3r," in *Proceedings of the European Conference on Computer Vision (ECCV)* (2024), 71–91.

28. W. Deng, D. Campbell, C. Sun, et al., "Pos3r: 6D Pose Estimation for Unseen Objects Made Easy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025), 16818–16828.

29. J. Sun, Z. Wang, S. Zhang, et al., "Onepose: One-Shot Object Pose Estimation Without Cad Models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 6825–6834.

30. X. He, J. Sun, Y. Wang, D. Huang, H. Bao, and X. Zhou, "Onepose++: Keypoint-Free One-Shot Object Pose Estimation Without Cad Models," *Advances in Neural Information Processing Systems* 35 (2022): 35103–35115, https://doi.org/10.52202/068431-2544.

31. Y. Liu, Y. Wen, S. Peng, et al., "Gen6d: Generalizable Model-Free 6-Dof Object Pose Estimation From Rgb Images," in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022), 298–315.

32. Z. Fan, P. Pan, P. Wang, Y. Jiang, D. Xu, and Z. Wang, "Pope: 6-Dof Promptable Pose Estimation of Any Object in Any Scene With One Reference," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 7771–7781.

33. V. N. Nguyen, T. Groueix, G. Ponimatkin, V. Lepetit, and T. Hodan, "Cnos: A Strong Baseline for Cad-Based Novel Object Segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops* (2023), 2134–2140.

34. Y. Xiong, B. Varadarajan, L. Wu, et al., "Efficientsam: Leveraged Masked Image Pretraining for Efficient Segment Anything," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 16111–16121.

35. M. Oquab, T. Darcet, T. Moutakanni, et al., "Dinov2: Learning Robust Visual Features Without Supervision," preprint arXiv: 2304.07193 (2023).

36. Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the Continuity of Rotation Representations in Neural Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), 5745–5753.

37. V. N. Nguyen, Y. Hu, Y. Xiao, M. Salzmann, and V. Lepetit, "Templates for 3d Object Pose Estimation Revisited: Generalization to New Objects and Robustness to Occlusions," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 6771–6780.

38. S. Hinterstoisser, V. Lepetit, S. Ilic, et al., "Model Based Training, Detection and Pose Estimation of Texture-Less 3d Objects in Heavily Cluttered Scenes," in *Proceedings of the Asian Conference on Computer Vision (ACCV)* (2013), 548–562.

39. B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The Ycb Object and Model Set: Towards Common Benchmarks for Manipulation Research," in *Proceedings of the International Conference on Advanced Robotics (ICAR)* (2015), 510–517.

40. W. Goodwin, S. Vaze, I. Havoutis, and I. Posner, "Zero-Shot Category-Level Object Pose Estimation," in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022), 516–532.

41. J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-Free Local Feature Matching With Transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 8922–8931.

42. J. Yang, Y. Gao, D. Li, and S. L. Waslander, "Robi: A Multi-View Dataset for Reflective Objects in Robotic Bin-Picking," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2021), 9788–9795.

43. J. Huang, J. Liang, J. Hu, et al., "Xyz-Ibd: A High-Precision Bin-Picking Dataset for Object 6d Pose Estimation Capturing Real-World Industrial Complexity," preprint arXiv:2506.00599 (2025).