HOSt3R: Keypoint-free Hand-Object 3D Reconstruction from RGB images

Anilkumar Swamy[1,2] Vincent Leroy[1] Philippe Weinzaepfel[1]
Jean-Sébastien Franco[2] Grégory Rogez[1]

[1]NAVER LABS Europe [2]Inria centre at the University Grenoble Alpes

Abstract

Hand-object 3D reconstruction has become increasingly important for applications in human-robot interaction and immersive AR/VR experiences. A common approach for object-agnostic hand-object reconstruction from RGB sequences involves a two-stage pipeline: hand-object 3D tracking followed by multi-view 3D reconstruction. However, existing methods rely on keypoint detection techniques, such as Structure from Motion (SfM) and hand-keypoint optimization, which struggle with diverse object geometries, weak textures, and mutual hand-object occlusions, limiting scalability and generalization. As a key enabler to generic and seamless, non-intrusive applicability, we propose in this work a robust, keypoint detector-free approach to estimating hand-object 3D transformations from monocular motion video/images. We further integrate this with a multi-view reconstruction pipeline to accurately recover hand-object 3D shape. Our method, named HOSt3R, is unconstrained, does not rely on pre-scanned object templates or camera intrinsics, and reaches state-of-the-art performance for the tasks of object-agnostic hand-object 3D transformation and shape estimation on the SHOWMe benchmark. We also experiment on sequences from the HO3D dataset, demonstrating generalization to unseen object categories.

1. Introduction

Understanding and reconstructing hand–object interactions in 3D is a key challenge with broad applications in robotics, augmented/virtual reality (AR/VR), and human–computer interaction. Whether enabling natural user interfaces, immersive experiences, or safe object manipulation in collaborative settings, accurate 3D reconstruction of both the hand and the object is essential. To address this challenge, we propose a two-stage pipeline that involves hand-object transformation estimation and multi-view reconstruction, designed to achieve robust hand-object reconstruction from monocular video or images capturing rigid hand-object motion (see Figure 1).

Figure 1. **Qualitative Hand-Object reconstructions on the HO3D dataset.** The first column is an RGB frame of the input sequence, followed by 3 different views of the reconstructed hand-object shape using HOSt3R, our proposed method.

Traditional methods for hand-object 3D reconstruction often rely on multi-stage processes, including parametric hand model prediction or keypoint detection, and Structure from Motion (SfM). Although effective in controlled settings, these approaches face significant limitations when applied to complex, real-world scenarios, particularly when dealing with occlusions, dynamic environments, uniform textures and diverse object shapes. Thus, there is a pressing need for more robust and scalable solutions that can handle these challenges without the constraints of template-based

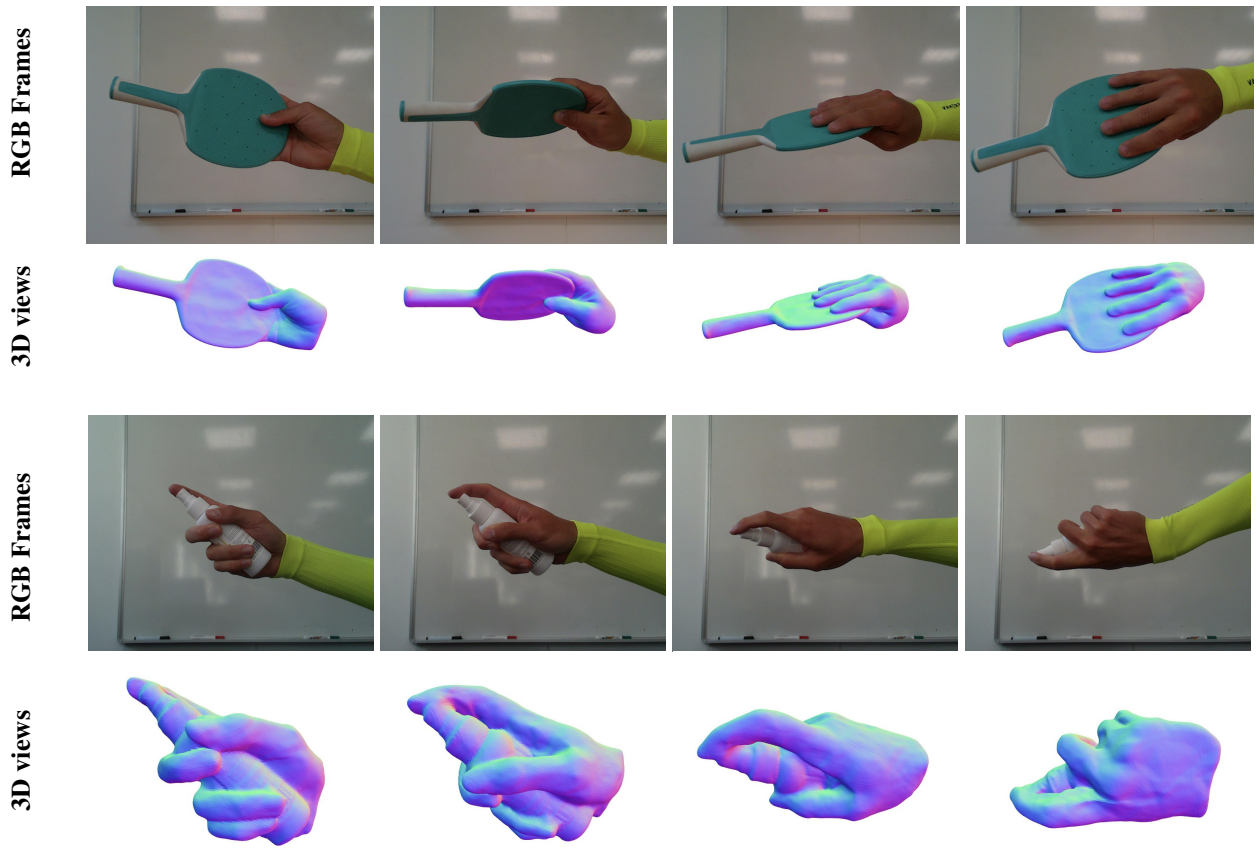

Figure 2. **Keypoint-free hand-object 3D reconstruction**: Given a monocular video sequence of a hand-object motion with an unknown object, our method reconstructs high-fidelity 3D hand-object surfaces. Each row shows one sequence of the SHOWMe dataset: the input image followed by three views of the reconstructed normals. Best viewed in color.

or keypoint-dependent techniques. In this work, we address these challenges by proposing a novel approach to hand-object reconstruction without keypoint detection that offers improved scalability and generalization.

Existing hand-object reconstruction methods often assume the availability of known object templates [12, 15, 16, 18, 20, 23, 40–42], limiting their applicability in in-the-wild scenarios where such templates may not be available. Other approaches [4, 5, 13, 19] do not require known object templates but are trained on datasets containing a limited number of objects, resulting in poor generalization for unseen objects. More recent methods [42, 43] attempt to overcome this limitation by learning object shape priors across six object categories and using these priors for hand-object shape reconstruction. However, while these methods improve generalization on novel objects, the reconstructed shapes tend to lack high fidelity.

Several recent studies [11, 17, 31, 32] have demonstrated the potential of a two-stage pipeline for hand-object reconstruction, using keypoint-based 6DoF hand and object tracking followed by multiview reconstruction using implicit neural representations. The quality of 6DoF tracking is critical for achieving a detailed 3D hand-object reconstruction. For example, [17] uses 3D hand keypoints for 6DoF tracking, while [11, 31] employ poses derived from structure-from-motion pipelines. In contrast, in [32], we introduced a robust hand-object transformation estimation technique that performs well on challenging scenarios, such as small, uniformly textured, or occluded objects, but requires fine-tuning when applied to new datasets or varying environments.

In this work, we propose a keypoint-free framework for joint hand–object 3D reconstruction from monocular hand–object motion videos. Our approach, termed HOSt3R, is inspired by recent advances in scene-level reconstruction and is tailored to the challenges of hand–object interactions. HOSt3R is designed to address key limitations of existing methods such as failure in the presence of untextured objects, mutual hand–object occlusion, and variations in background or camera intrinsics, which often hinder keypoint-based pipelines [11, 17, 31, 32].

We begin by estimating dense 3D pointmaps (*i.e.*, 3D

points for every pixel) for pairs of input images. Using these pointmaps, we compute the relative poses between image pairs, and then apply pose averaging across all pairs to recover global hand–object transformations. These transformations are used to initialize a neural implicit model, which jointly optimizes hand–object shape and motion using differentiable volumetric techniques.

Building upon DUSt3R [36], we adapt its two-stage pointmap estimation and alignment strategy to the more complex setting of hand–object scenes. While DUSt3R achieved breakthrough results in 3D scene reconstruction, its reliance on scene graph optimization introduces high memory requirements, making it impractical for large datasets or long video sequences. We address this limitation by focusing on pairwise point cloud estimation for hand–object pixels, followed by pose averaging, which enables our method to scale efficiently without the memory overhead of full scene graph optimization. By combining the pairwise estimation network with a pose averaging framework, we can scale to longer sequences. Finally, we incorporate the transformations computed in a multi-view reconstruction pipeline to further enhance the accuracy and detail of the recovered hand-object shapes, see examples in Fig. 2.

In summary, our key contributions are as follows:

1. We propose a keypoint-free framework for hand-object transformation estimation that is robust to a variety of objects and camera parameter changes.
2. We integrate the estimated hand-object transformations within a multi-view reconstruction pipeline to achieve template-free hand-object 3D shape reconstruction.
3. We benchmark the performance of our hand-object reconstruction method on the SHOWMe benchmark.
4. We also demonstrate the generalization ability of the proposed framework on the HO3D dataset.

2. Related work

Hand-Object reconstruction methods. Hand-object (HO) reconstruction from a single RGB image or monocular video presents significant challenges due to mutual occlusion between the hand and object, complex motion, and variability in object shape and appearance. The availability of depth information or a known 3D object models can facilitate shape estimation. Early works [1, 24, 34, 35, 44] used RGB-D or multi-view inputs to simplify the reconstruction process, but recent advances have focused on using monocular RGB images to achieve similar outcomes.

Hand-object reconstruction approaches generally fall into two categories: parametric model-based techniques and implicit representation-based methods. Parametric model-based approaches [3, 14, 20, 22, 26, 28] typically rely on predefined object templates or category-specific models to estimate hand and object poses. Some methods combine parametric models with implicit representations [4, 42] to enhance reconstruction detail. For example, [42] assumes known 3D templates and employs Signed Distance Functions (SDFs) to reconstruct hand and object shapes with greater fidelity. Similarly, [27] builds separate models for hand and object shapes, but this approach requires camera calibration and online training, limiting its use to known object templates.

In contrast, implicit representation-based methods [19, 27] represent shapes as continuous functions, allowing more flexible and expressive reconstructions. For instance, [4] reconstructs generic hand-held objects without relying on a pre-defined templates, though the method struggles to generalize across diverse object geometries. These methods face limitations in shape generalization, as they often require training on specific object sets [19].

Monocular video-based methods have also gained attention for joint hand-object reconstruction. For example, [15] leverages photometric consistency over time to improve accuracy, while [16] explores optimization-based approaches. [22] uses spatial-temporal consistency to select pseudo-labels for self-training. However, a key limitation of these methods is the reliance on object template meshes during inference, which effectively reduces the reconstruction task to a 6-DoF pose estimation problem. In contrast, our method eliminates the need for object templates, enabling the reconstruction of arbitrary hand-object shapes with high precision.

Keypoint-based hand-object transformation. Most hand-object reconstruction methods from multiple RGB images [11, 17, 31, 43] follow a two-stage pipeline: (i) hand-object transformation estimation and (ii) shape estimation from the obtained transformations. Any method that uses either detected 2D hand keypoints [25, 29, 37] or salient keypoints [30] on the image to estimate hand-object transformations is considered keypoint-based.

Several works [17, 31, 43] employ off-the-shelf hand keypoint detectors [29, 37] to compute initial hand–object transformations. These approaches perform well when the hand remains visible (typically in sequences involving small objects) but they fail when the hand is occluded by larger objects. Other methods [11, 31] rely on Structure-from-Motion (SfM) pipelines based on detected image keypoints.

While effective for scenes with textured or feature-rich objects, this strategy fails in the presence of small or uniformly textured objects due to a lack of salient features. In contrast to these keypoint-based approaches, we propose a method that does not rely on keypoint detection. Instead, we estimate dense 3D pointmaps, where every pixel is associated with a 3D point, and solve a 2D-to-3D matching problem. This enables robust pose estimation even in the absence of distinctive keypoints.

Keypoint-free hand-object transformation. An emerging direction in hand–object reconstruction is to directly regress camera parameters [38, 39] as initialization for a global photometric optimization [31, 32]. This approach generally offers better performance and robustness than keypoint-based methods, in scenes with small or uniformly textured objects. However, direct pose regression remains challenging and prone to inaccuracies, as it must disentangle complex interactions between camera intrinsics, extrinsics, and the underlying 3D scene structure [36]. To address this, we draw inspiration from [36] and tackle pose prediction via pointmap regression: a simple overparameterization of these three quantities that enables easy training and inference, offering strong robustness and generalization capabilities.

3. Rigid transformation estimation framework

3.1. Pairwise pointmap estimation network

Pointmap. A pointmap $X \in \mathbb{R}^{W\times H\times 3}$ is a pixelwise prediction where each prediction represents a 3D point in space. When associated with its corresponding RGB image I of resolution $W \times H$, this pointmap establishes a direct and unique mapping between the image pixels and the 3D points in the scene. Specifically, for each pixel in the image, indexed by coordinates (i, j), there is a corresponding 3D point $X_{i,j}$, such that every pixel $I_{i,j}$ in the image is uniquely linked to a 3D point $X_{i,j}$. Formally, this can be expressed as $I_{i,j} \leftrightarrow X_{i,j}$ for all pixel coordinates $i, j \in \mathbb{N}^{W\times H}$. This mapping assumes that each camera ray intersects with exactly one 3D point, which implies that cases involving translucent or semi-transparent surfaces, where a camera ray might pass through multiple 3D points, are not considered in this scenario. The pointmap, therefore, provides a straightforward representation of the 3D structure of the scene, with each pixel in the image corresponding to a unique location in 3D space. This assumption simplifies the interpretation and processing of the pointmap.

To train the network with strong supervision, we need ground-trtuh pointmap for every input image. To that end, we consider a camera with intrinsic parameters defined by the matrix $K \in \mathbb{R}^{3\times 3}$. Given this matrix, the pointmap X of the observed scene can be directly computed from the ground-truth depth map $D \in \mathbb{R}^{W\times H}$, where W and H are the width and height of the image, respectively. The relationship between the pointmap X and the depth map D is given by:

$$X_{i,j} = K^{-1} \begin{bmatrix} iD_{i,j} \\ jD_{i,j} \\ D_{i,j} \end{bmatrix}. \tag{1}$$

Here, $(i, j) \in \mathbb{N}^{W\times H}$ represent the x-y pixel coordinates in the image, and X is expressed in the camera's coordinate frame.

Further, we denote $X^{n,m}$ as the pointmap X^m from camera n expressed in the reference frame of image m. This transformation is described by:

$$X^{n,m} = P_m {P_n}^{-1} H(X^n), \tag{2}$$

where $P^m, P^n \in \mathbb{R}^{3\times 4}$ are the world-to-camera pose matrices for views m and n, respectively, and $H : (x, y, z) \to (x, y, z, 1)$ is the homogeneous coordinate mapping.

Pointmap estimation network. We build a pairwise pointmap estimation network f that takes two RGB input images $I_1, I_2 \in \mathbb{R}^{W\times H\times 3}$ as input and produces two corresponding pointmaps $X^{1,1}, X^{2,1} \in \mathbb{R}^{W\times H\times 3}$, along with associated confidence maps $C^{1,1}, C^{2,1} \in \mathbb{R}^{W\times H}$. It is important to note that both pointmaps are expressed within the same reference frame as I_1 to ensure consistency across generated outputs. The network f's architecture is inspired by [36, 38] so that we can benefit from the pretraining of both [38] and [36]. [38] is trained for cross-view image completion and [36] is trained for 2D to 3D matching problem. We follow the same architecture for pairwise pointmap estimation as shown in Fig. 3. Networks f is composed of two symmetrical branches, one of each image comprising an image encoder, decoder, and then the regression module called "Head", a sequence of MLP layers. The two input images are first divided into an equal number of patches of size 16×16. Then these patches are processed through a shared ViT encoder [10] to compute patch embedding (also called as 'token') representations E_1 and E_2:

$$E_1 = Encoder(I_1), \quad E_2 = Encoder(I_2). \tag{3}$$

The decoder is a transformer network with cross-attention followed by self-attention. In self-attention, each token attends to every other token of the same view and in cross-attention, a token from the first view attends to every other token from the second view. The decoder module is comprised of blocks of individual decoders and information is always shared between the two image decoders:

$$Z_i^1 = \text{DecoderBlock}_i^1\left(Z_{i-1}^1, Z_{i-1}^2\right), \tag{4}$$
$$Z_i^2 = \text{DecoderBlock}_i^2\left(Z_{i-1}^2, Z_{i-1}^1\right), \tag{5}$$

for $i = 1, \ldots, B$ for a decoder with B blocks and initialized with encoder tokens $Z_0^1 := E^1$ and $Z_0^2 := E^2$. Here, $\text{DecoderBlock}_i^v\left(Z^1, Z^2\right)$ denotes the i-th block in branch $v \in \{1, 2\}$, Z^1 and Z^2 are the input tokens, with D^2 the tokens from the other branch. Finally, each image decoder block output is then fed to a "Head" MLP network to regress

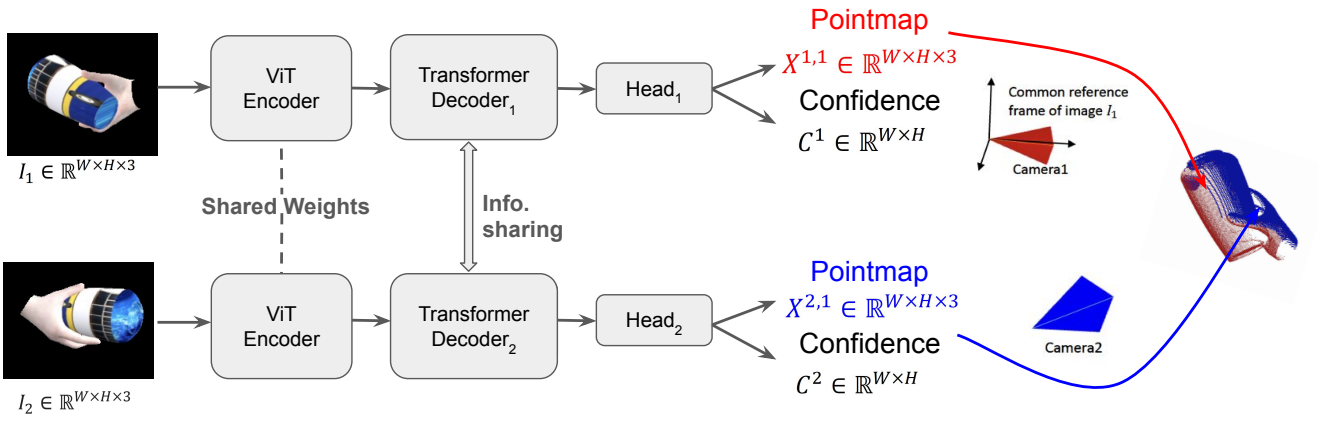

Figure 3. **Pairwise pointmaps estimation network.** Given two views of the same hand-object pair (I_1, I_2), the network processes both views through a shared Vision Transformer (ViT) encoder. Each view's features are decoded using distinct decoders that share mutual information. These features are then passed through the 'Head' module to predict the pointmap X and the confidence score map C for each view. The pointmaps for both views are predicted within the coordinate system of the first view's image, and the network is trained by minimizing the error between the ground truth and the predicted pointmaps.

a pointmap and an associated confidence map:

$$X^{1,1}, C^{1,1} = \text{Head}^1\left(Z_0^1, \ldots, Z_B^1\right), \tag{6}$$

$$X^{2,1}, C^{2,1} = \text{Head}^2\left(Z_0^2, \ldots, Z_B^2\right). \tag{7}$$

Training Objective. The network f is trained using 3D points regression loss with confidence aware terms.

It builds upon a standard regression loss ℓ_{regr} where we use the Euclidean distance between the ground-truth and predicted pointmaps. Let us denote the ground-truth pointmaps as $\bar{X}^{1,1}$ and $\bar{X}^{2,1}$, with two corresponding sets of hand-object pixels $\mathcal{D}^1$, $\mathcal{D}^2 = \{i^1, \ldots, i^L\}$ on which the ground-truth is defined. The regression loss for a hand-object mask M, pixel i in view $v \in \{1, 2\}$ is defined as below:

$$\ell_{\text{regr}}(v, i) = M_i \cdot \left| \frac{X_i^{v,1}}{z} - \frac{\bar{X}_i^{v,1}}{\bar{z}} \right|. \tag{8}$$

M_i indicates whether the pixel belongs to the hand-object ($M_i = 1$) or to the background ($M_i = 0$). If $M_i = 0$ then pixel i is ignored for the loss calculation. We normalize the predicted and ground-truth pointmaps by scaling factors to handle scale ambiguity between the predictions and ground truth pointmaps. The scaling factors for ground truth and predictions are $z = \text{norm}(X^{1,1}, X^{2,1})$ and $\bar{z} = \text{norm}(\bar{X}^{1,1}, \bar{X}^{2,1})$, respectively. This basically represents pointmap as the distance of all pointmaps from the origin:

$$\text{norm}(X^1, X^2) = \frac{1}{|\mathcal{D}^1| + |\mathcal{D}^2|} \sum_{v\in\{1,2\}} \sum_{i\in\mathcal{D}^v} \|X_i^v\|. \tag{9}$$

To make ℓ_{reg} confidence aware, we enable the joint prediction of score for each pixel which indicates the confidence of the pointmap prediction. The final loss is the confidence weighted regression loss from Eq. (8) for all hand-object pixels:

$$\ell_{\text{conf}} = \sum_{v\in\{1,2\}} \sum_{i\in\mathcal{D}^v} C_i^{v,1} \ell_{\text{regr}}(v, i) - \alpha \log C_i^{v,1}, \tag{10}$$

Here, $C_i^{v,1}$ represents the confidence score for the i-th pixel prediction, while α serves as a hyper-parameter controlling the regularization term. To guarantee that the confidence score remains strictly positive, it is typically defined as $C_i^{v,1} = 1 + \exp(C_i^{v,1}) > 0$.

3.2. Relative pose computation from pointmaps

The aligned pointmaps from the two views have several useful properties: they are expressed in a common coordinate system, are spatially aligned, and maintain pixel-level correspondence. As a result, the estimated pointmaps can be used to compute the relative pose between the two views. Given two images I_1 and I_2 with corresponding estimated pointmaps $X^{1,1}$ and $X^{2,1}$(in I_1 coordinate system), we can compute the relative pose between the two views as follows:

1. Compute Focal Length from Depth ($X_z^{1,1}$):

$$f = \text{estimateFocalLength}(X_z^{1,1}). \tag{11}$$

2. Set Principal Point at the Center:

$$(c_x, c_y) = \left(\frac{W}{2}, \frac{H}{2}\right), \tag{12}$$

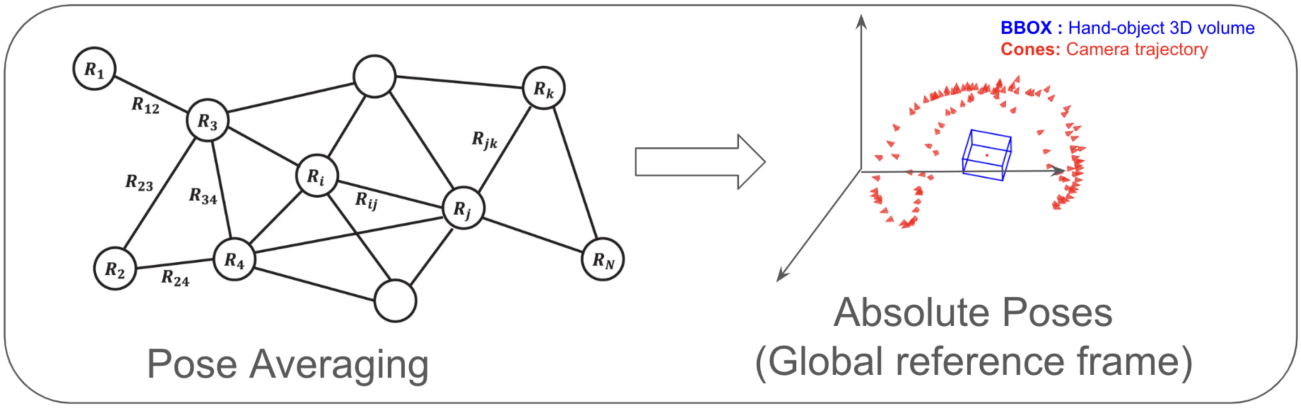

Figure 4. **Pose averaging** is a process of estimating absolute poses from a set of relative poses. In the graph, each node R_k represents absolute transformation (rotation and translation) to be optimized, edges represent the measured relative transformation R_{ij}. Pose averaging over this graph yields global absolute hand-object transformations of all nodes, plotted as red camera cones in a single coordinate frame.

where W and H represent the width and height of the image, respectively.

3. Solve PnP with RANSAC: To estimate the relative pose between the two views, solve the PnP (Perspective-n-Point) problem using RANSAC:

$$[R, t] = \text{PnP\_RANSAC}\left(\text{pixel\_coords}(H, W), X^{2,1}, K\right), \tag{13}$$

where:

$$K = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{14}$$

Here, R and t represent the relative rotation and translation between the two images I_1 and I_2.

3.3. Pose averaging on relative poses

The relative pose computed using the output of network f is in an arbitrary local coordinate system. However, for 3D hand-object reconstruction from an input hand-object motion sequence, we need to compute global hand-object absolute transformations. To this end, we create a pairwise graph from a set of images $I_1, I_2, ..., I_N$ for a given sequence. We first construct a connectivity graph $G(V, E)$ where N images form vertices V and each edge $e = (n, m) \in E$ indicates that images I_n and I_m share hand-object visual information. To create a graph, we first create all possible image pairs (a fully connected graph) and then use the classifier from [32] to filter out the invalid image pairs to reduce the fully connected graph to a sparse graph. The idea is to retain the image pair edges that share enough hand-object visual information. The process of estimating rotation and translation for this type of problem is separable [9], and we explain how to apply this to the camera pose problem as follows.

Rotation Averaging. Given a sequence of frames $\{I_1, I_2, ...I_N\}$, we estimate and refine the relative poses, then construct a directed pairwise graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where the N frames correspond to the graph's vertices, and each edge $e = (i, j) \in \mathcal{E}$ encodes the relative rotational relationship between frames I_i and I_j. To compute the global rotations from this graph, we apply the Shonan rotation averaging method [9], which frames the problem as a factor graph. In this graph, each node represents an unknown absolute rotation to be determined, and the edges, or factors, correspond to the previously estimated relative rotations, which are noisy. The objective is to minimize the sum of the Frobenius norms between the predicted and the measured relative rotations:

$$\min_{R \in SO(p)^n} \sum_{(i,j)\in\mathcal{E}} k_{ij} \|R_j - R_i \hat{R}_{ij}\|_F^2, \tag{15}$$

from $SO(3)$ to $SO(p)$ where $p > 3$ until the termination, k_{ij} is the concentration parameter for assumed noise model.

Translation Averaging. Translation averaging involves estimating the absolute translations from the previously obtained absolute rotations $\hat{R}_{i=1..N}$ and the relative translations $\hat{t}_{ij}$. To achieve this, we employ the Gaussian Factor Graph optimization framework [8], which minimizes the sum of squared Euclidean distances between the estimated translations (t_i, t_j) and the measured relative translations t_{ij}, while also enforcing constraints based on the relative translation measurements:

$$\min_{t_0, t_1, ..., t_N} \sum_{(i,j)\in\mathcal{E}} \left\| R_i \cdot \hat{t}_{ij} - (t_j - t_i) \right\|^2. \tag{16}$$

The optimization problem is subsequently addressed using a linear system solver from GTSAM [7], with the anchor

Method	Rigid Transformation Error				
	Rot Error ↓	Trans Error ↓	Det. Rate (%) ↑	@(15cm&15°)(%) ↑	@(30cm&30°)(%) ↑
DOPE [37] + fixed hand pose	28.9†	0.23†	99.0	30.2	69.1
DOPE [37] + median filtering	28.7	0.22	**100.0**	30.4	69.2
DOPE [37] + PoseBERT [2]	28.0	0.22	**100.0**	19.5	58.2
COLMAP [30]	**15.9**†	**0.06**†	78.3	**59.5**	67.2
SHOWMe [32]	20.9	0.12	**100.0**	46.0	80.0
HOSt3R (ours)	18.2	0.08	**100.0**	50.5	**86.3**

Table 1. **Rigid transformation estimation comparison.** The 'Rot. error' is the geodesic distance expressed in degrees with the ground-truth rigid transformation. The 'Trans error' is the MSE. † means the metrics are computed for the frames for which the pose is successfully recovered.

factor set by fixing the first pose's position at the origin. An illustration of the pose averaging process is provided in Fig. 4.

4. Experimental results

In this section, we first describe how the estimated hand–object transformations are integrated into a multi-view stereo (MVS) pipeline to reconstruct the 3D shape of the hand and object. We then detail the datasets used for training and evaluation, and finally present quantitative and qualitative results for both hand–object transformation and 3D shape estimation.

Integration with multi-view reconstruction. We estimate pairwise relative poses and perform pose averaging across a hand–object scene graph to obtain global transformations in a common coordinate system. These transformations are then integrated into the multi-view 3D hand–object reconstruction pipeline introduced in [32]. Specifically, we use the estimated rigid transformations as initialization for an implicit neural representation method, which reconstructs both the surface geometry and color of the hand–object scene. During reconstruction, we also refine the transformations through joint optimization to correct for any initial inaccuracies. For each sequence, we sample 60 evenly spaced frames, *e.g.* selecting every 15th frame in a 900-frame video, to perform the 3D reconstruction.

Datasets. We initialize our pairwise pointmap estimation network with the pre-trained weights of DUSt3R [36], originally trained on indoor scene datasets. We then fine-tune the network on a synthetically generated dataset. For synthetic data generation, we adapt the pipeline from ObMan [13], which produces parametric hand (MANO) grasp poses for approximately 2.7K everyday object models across 8 object categories. We modify this pipeline to generate multi-view data with varied camera intrinsics, extrinsics, and hand–object occlusion ratios. The resulting dataset includes multi-view RGB images, camera intrinsics and extrinsics, and depth maps. Pointmaps are derived from depth maps and intrinsics using Equation 1. A depiction of the generated multi-view dataset is provided in the supplementary material.

Quantitative results. Following the evaluation protocol of [31, 32] we evaluate both rigid transformation estimation and joint hand-object shape reconstruction. We adopt the same experimental setup in terms of data splits and the number of frames used for evaluation. We report HOSt3R's rigid transformation errors on the SHOWMe dataset, using the same evaluation table as in [32] (see Tab. 1). Our method outperforms the SHOWMe baseline by a margin of 2.7° in rotation error and 4.0cm in translation error. In addition, our method successfully recovers transformations for all sequences, achieving a 100% detection rate. Furthermore, HOSt3R improves over SHOWMe by 4.5% in the number of frames with translation error under 15cm and rotation error under 15°. It also achieves the highest percentage of frames with both translation and rotation errors below 30cm and 30°, respectively. While slightly underperforming COLMAP in some metrics, our method is significantly more robust, achieving a 100% detection rate and 86.3% correct frames under the 30cm / 30° threshold.

We then evaluate hand–object reconstruction on the SHOWMe benchmark, following the protocol introduced in [31, 32]. In Tab. 2, we report standard metrics including Accuracy, Completion, and F-score. HOSt3R achieves the best overall performance among all baselines and is on par with the method proposed in [32]. Qualitative results are shown in Fig. 5 for several SHOWMe sequences; additional results are provided in the supplementary material. The reconstructed hand–object geometry is highly detailed and consistent across a variety of grasps, object shapes, sizes, and textures, demonstrating the robustness and generalization capability of the proposed HOSt3R approach.

Generalization. Our proposed method demonstrates strong generalization to unseen hand–object sequences

Rigid Transform	Recon. Method	Rec. rate (%) ↑	Acc.† (cm) ↓	Comp.† (cm) ↓	Acc. ratio @5mm (%) ↑	Comp. ratio @5mm (%) ↑	Fscore @5mm (%) ↑
GT	IHOI [42]	87.3	0.79	1.34	41.7	37.8	39.3
GT	VH [21]	93.7	0.42	0.65	67.3	61.6	63.6
GT	FDR [33]	95.8	0.35	0.49	75.8	72.0	73.5
GT	HHOR [17]	**100.0**	**0.35**	**0.32**	**81.1**	**83.8**	**82.3**
DOPE [37]	FDR [33]	92.7	1.02	3.18	31.7	15.7	20.0
COLMAP [30]	FDR [33]	76.0	0.64	0.79	39.3	36.2	37.6
COLMAP [30]	HHOR [17]	72.9	0.65	0.74	40.9	42.1	41.3
SHOWMe [32]	HHOR [17]	**100.0**	0.61	0.62	55.6	56.0	55.6
HOSt3R (ours)	HHOR [17]	**100.0**	**0.58**	**0.59**	**56.4**	**57.0**	**56.4**

Table 2. **Hand-object reconstruction evaluation** using ground-truth and estimated rigid transformations. † means that the metrics are obtained by computing on the reconstructed mesh only, the failing ones are not taken into account, making direct comparisons between different methods unfair. DOPE refers to the variant 'DOPE + fixed hand pose' from Table 1.

from different datasets. We validate this capability on sequences from the HO3D dataset, which features novel objects, hand shapes, backgrounds, and hand–object motions. We present qualitative results on HO3D sequences that align with our experimental settings, specifically, those with rigid hand–object motion and a comparable number of frames, as shown in Fig. 1. Our method successfully reconstructs detailed hand–object shapes without requiring fine-tuning or access to camera intrinsics. However, some parts of the hand, particularly the fingers, are not fully recovered, as illustrated in the last row and last column of Fig. 1. This limitation is primarily due to insufficient viewpoint coverage of the fingers in the input frames, which remains a common challenge in multi-view reconstruction.

5. Discussion

We present HOSt3R, a novel method for robust hand–object transformation and 3D reconstruction that overcomes key limitations of prior work. By integrating a DUSt3R-inspired pairwise relative pose estimation network within a pose averaging framework, our approach achieves robustness to camera variations, appearance changes, and occlusions, without relying on keypoint detectors. This design also alleviates memory constraints in large-scale reconstructions. On the SHOWMe dataset, our method significantly reduces pose errors, achieving a 100% detection rate and 86.3% accuracy under the 30cm & 30° error threshold, even in challenging conditions. It generalizes well across object types, grasp styles, and scenes, and offers improved geometric fidelity compared to existing baselines. Qualitative results on the HO3D dataset further demonstrate strong generalization, achieved without fine-tuning or access to camera intrinsics. While minor limitations remain, particularly in recovering fine finger details under sparse viewpoints, future work could incorporate diffusion-based shape priors [6] to improve reconstruction quality in highly occluded regions.

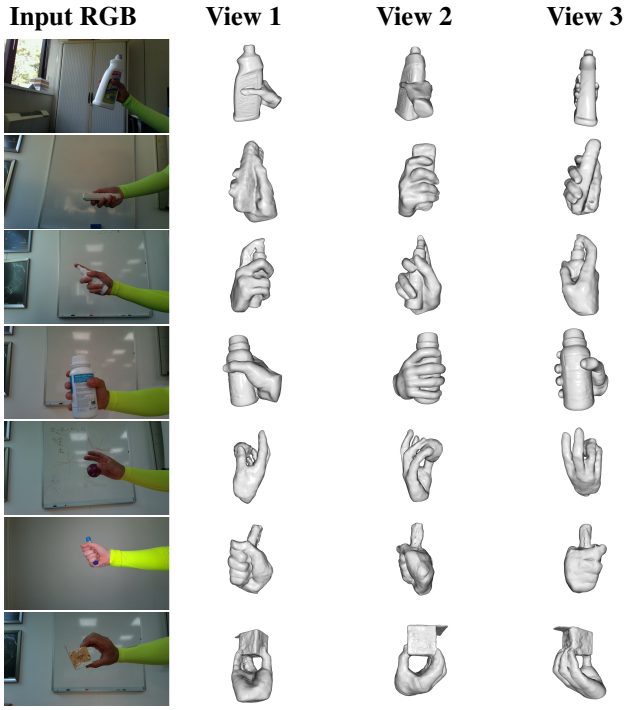

Figure 5. **Qualitative Hand-object reconstructions on sequences from the SHOWMe dataset.** Each row shows one sequence: the first image is the RGB input, followed by three views of the reconstructed hand-object shape using our method.

References

[1] Luca Ballan, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012. 3

[2] Fabien Baradel, Romain Brégier, Thibault Groueix, Philippe Weinzaepfel, Yannis Kalantidis, and Grégory Rogez. Posebert: A generic transformer module for temporal 3d human modeling. *IEEE Trans. PAMI*, 2022. 7

[3] Zhe Cao, Ilija Radosavovic, Angjoo Kanazawa, and Jitendra Malik. Reconstructing hand-object interactions in the wild. In *ICCV*, 2021. 3

[4] Zerui Chen, Yana Hasson, Cordelia Schmid, and Ivan Laptev. Alignsdf: Pose-aligned signed distance fields for hand-object reconstruction. In *ECCV*, 2022. 2, 3

[5] Zerui Chen, Shizhe Chen, Cordelia Schmid, and Ivan Laptev. gsdf: Geometry-driven signed distance functions for 3d hand-object reconstruction. In *CVPR*, 2023. 2

[6] Wencan Cheng, Hao Tang, Luc Van Gool, and Jong Hwan Ko. Handdiff: 3d hand pose estimation with diffusion on image-point cloud. In *CVPR*, 2024. 8

[7] Frank Dellaert and GTSAM Contributors. borglab/gtsam, 2022. 6

[8] Frank Dellaert and Michael Kaess. *Factor Graphs for Robot Perception*. Foundations and Trends in Robotics, 2017. 6

[9] Frank Dellaert, David M Rosen, Jing Wu, Robert Mahony, and Luca Carlone. Shonan rotation averaging: Global optimality by surfing so (p)ˆ n so (p) n. In *ECCV*, 2020. 6

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 4

[11] Zicong Fan, Maria Parelli, Maria Eleni Kadoglou, Xu Chen, Muhammed Kocabas, Michael J Black, and Otmar Hilliges. Hold: Category-agnostic 3d reconstruction of interacting hands and objects from video. In *CVPR*, 2024. 2, 3

[12] Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent Lepetit. Handsformer: Keypoint transformer for monocular 3d pose estimation ofhands and object in interaction. *CVPR*, 2022. 2

[13] Yana Hasson, Gül Varol, Dimitris Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019. 2, 7

[14] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019. 3

[15] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *CVPR*, 2020. 2, 3

[16] Yana Hasson, Gül Varol, Cordelia Schmid, and Ivan Laptev. Towards unconstrained joint hand-object reconstruction from rgb videos. In *3DV*, 2021. 2, 3

[17] Di Huang, Xiaopeng Ji, Xingyi He, Jiaming Sun, Tong He, Qing Shuai, Wanli Ouyang, and Xiaowei Zhou. Reconstructing hand-held objects from monocular video. In *SIGGRAPH Asia*, 2022. 2, 3, 8

[18] Shijian Jiang, Qi Ye, Rengan Xie, Yuchi Huo, and Jiming Chen. Hand-held object reconstruction from rgb video with dynamic interaction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12220–12230, 2025. 2

[19] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. In *3DV*, 2020. 2, 3

[20] Mia Kokic, Danica Kragic, and Jeannette Bohg. Learning to estimate pose and shape of hand-held objects from rgb images. In *IROS*, 2019. 2, 3

[21] Vincent Leroy, Jean-Sébastien Franco, and Edmond Boyer. Volume sweeping: Learning photoconsistency for multi-view shape reconstruction. *IJCV*, 2021. 8

[22] Shaowei Liu, Hanwen Jiang, Jiarui Xu, Sifei Liu, and Xiaolong Wang. Semi-supervised 3d hand-object poses estimation with interactions in time. In *CVPR*, 2021. 3

[23] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Generalized feedback loop for joint hand-object pose estimation. *IEEE Trans. PAMI*, 2019. 2

[24] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *ICCV*, 2011. 3

[25] JoonKyu Park, Yeonguk Oh, Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Handoccnet: Occlusion-robust 3d hand mesh estimation network. In *CVPR*, 2022. 3

[26] Neng Qian, Jiayi Wang, Franziska Mueller, Florian Bernard, Vladislav Golyanik, and Christian Theobalt. HTML: A Parametric Hand Texture Model for 3D Hand Reconstruction and Personalization. In *ECCV*. 2020. 3

[27] Wentian Qu, Zhaopeng Cui, Yinda Zhang, Chenyu Meng, Cuixia Ma, Xiaoming Deng, and Hongan Wang. Novel-view synthesis and pose estimation for hand-object interaction from sparse views. In *ICCV*, 2023. 3

[28] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM ToG*, 2017. 3

[29] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration. In *ICCVW*, 2021. 3

[30] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *CVPR*, 2016. 3, 7, 8

[31] Anilkumar Swamy, Vincent Leroy, Philippe Weinzaepfel, Fabien Baradel, Salma Galaaoui, Romain Brégier, Matthieu Armando, Jean-Sebastien Franco, and Grégory Rogez. SHOWMe: Benchmarking Object-agnostic Hand-Object 3D Reconstruction. In *ICCVW*, 2023. 2, 3, 4, 7

[32] Anilkumar Swamy, Vincent Leroy, Philippe Weinzaepfel, Fabien Baradel, Salma Galaaoui, Romain Brégier, Matthieu Armando, Jean-Sebastien Franco, and Grégory Rogez. Showme: Robust object-agnostic hand-object 3d reconstruction from rgb video. *CVIU*, 2024. 2, 4, 6, 7, 8

[33] Briac Toussaint, Maxime Genisson, and Jean-Sébastien Franco. Fast gradient descent for surface capture via differentiable rendering. In *3DV*, 2022. 8

[34] Dimitrios Tzionas, Abhilash Srikantha, Pablo Aponte, and Juergen Gall. Capturing hand motion with an rgb-d sensor, fusing a generative model with salient points. In *GCPR*, 2014. 3

[35] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing Hands in Action using Discriminative Salient Points and Physics Simulation. *IJCV*, 2016. 3

[36] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3d vision made easy. In *CVPR*, 2024. 3, 4, 7

[37] Philippe Weinzaepfel, Romain Brégier, Hadrien Combaluzier, Vincent Leroy, and Grégory Rogez. DOPE: Distillation of part experts for whole-body 3D pose estimation in the wild. In *ECCV*, 2020. 3, 7, 8

[38] Philippe Weinzaepfel, Vincent Leroy, Thomas Lucas, Romain Brégier, Yohann Cabon, Vaibhav Arora, Leonid Antsfeld, Boris Chidlovskii, Gabriela Csurka, and Revaud Jérôme. CroCo: Self-Supervised Pre-training for 3D Vision Tasks by Cross-View Completion. In *NeurIPS*, 2022. 4

[39] Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *ICCV*, 2023. 4

[40] Jane Wu, Georgios Pavlakos, Georgia Gkioxari, and Jitendra Malik. Reconstructing hand-held objects in 3d from images and videos. *arXiv preprint arXiv:2404.06507*, 2024. 2

[41] Lixin Yang, Xinyu Zhan, Kailin Li, Wenqiang Xu, Jiefeng Li, and Cewu Lu. Cpf: Learning a contact potential field to model the hand-object interaction. In *ICCV*, 2021.

[42] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3d reconstruction of generic objects in hands. In *CVPR*, 2022. 2, 3, 8

[43] Yufei Ye, Poorvi Hebbar, Abhinav Gupta, and Shubham Tulsiani. Diffusion-guided reconstruction of everyday hand-object interaction clips. In *ICCV*, 2023. 2, 3

[44] Hao Zhang, Yuxiao Zhou, Yifei Tian, Jun-Hai Yong, and Feng Xu. Single depth view based real-time reconstruction of hand-object interactions. *ACM ToG*, 2021. 3