HMDO : Markerless Multi-view Hand Manipulation Capture with Deformable Objects

Wei Xie* Zhipeng Yu* Zimeng Zhao Binghui Zuo Yangang Wang†

Southeast University, China

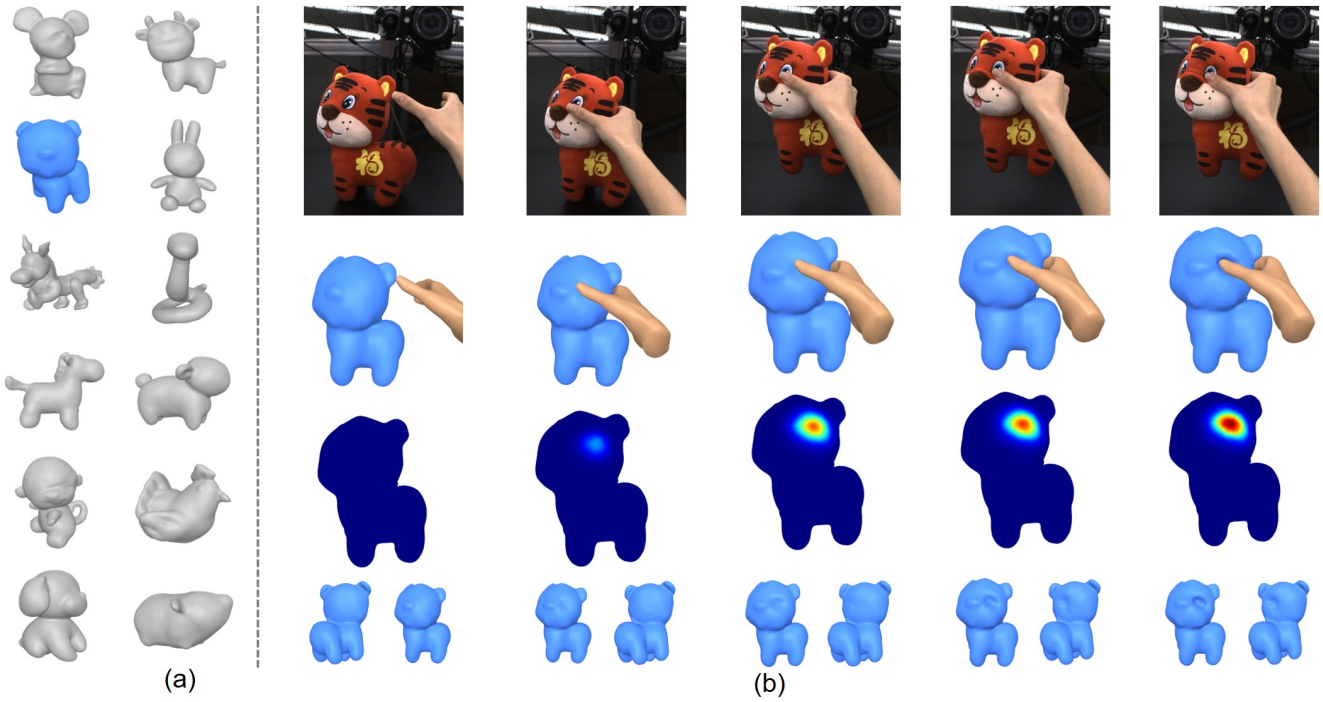


Figure 1. **HMDO dataset**. (a) 12 zodiacs as manipulated objects; (b) Data including multi-view synchronized images, 3D meshes and contact deformation maps.

Abstract

We construct the first markerless deformable interaction dataset recording interactive motions of the hands and deformable objects, called HMDO (Hand Manipulation with Deformable Objects). With our built multi-view capture system, it captures the deformable interactions with multiple perspectives, various object shapes, and diverse interactive forms. Our motivation is the current lack of hand and deformable object interaction datasets, as 3D hand and deformable object reconstruction is challenging. Mainly due to mutual occlusion, the interaction area is difficult to observe, the visual features between the hand and the object are entangled, and the reconstruction of the interaction area deformation is difficult. To tackle this challenge, we propose a method to annotate our captured data. Our key idea is to collaborate with estimated hand features to guide the object global pose estimation, and then optimize the deformation process of the object by analyzing the relationship between the hand and the object. Through comprehensive evaluation, the proposed method can reconstruct interactive motions of hands and deformable objects with high quality. HMDO currently consists of 21600 frames over 12 sequences. In the future, this dataset could boost the research of learning-based reconstruction of deformable interaction scenes.

***Keywords:** Deformable interaction, Markerless capture, Mulit-view dataset, Collaborative reconstruction.*

This work was supported in part by the National Natural Science Foundation of China (No. 62076061) and the Natural Science Foundation of Jiangsu Province (No. BK20220127). Wei Xie and Zhipeng Yu are co-first authors. Corresponding author: Yangang Wang. E-mail: yangangwang@seu.edu.cn. All the authors from Southeast University are affiliated with the Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Nanjing, China.

1. Introduction

Understanding the interactive motions of hands and objects is an important topic in computer vision and graphics due to its wide range of applications in virtual reality, augmented reality, robotics, etc. In recent years, thanks to the development of deep learning and the creation of several hand and rigid object interaction datasets [16, 8, 46, 6, 17, 24], the research on the reconstruction of hands and rigid objects from monocular images has developed rapidly. However, these recent methods fail in the reconstruction of hands and deformable objects interactions. Mainly due to the need to solve non-rigid deformations of closely interacting contact regions and the lack of datasets for hand and deformable object interactions. Therefore, it is necessary to break the limitation of the lack of datasets that record hands and deformable objects.

To facilitate the study of data-driven methods to address scenarios where hands and deformable objects closely interact, we construct the first multi-view deformable interaction dataset. We built a multi-view capture system, using 10 high-speed industrial cameras to synchronously capture the close interaction process between hands and deformable objects at high frame rates from different perspectives. However, creating annotations for interactive motions of hands and deformable objects is very challenging. Reconstructing hand and deformable object interactions is more complicated than reconstructing hands and rigid objects interactions, because we not only need to solve the rigid motions of deformable objects, but also recover the non-rigid motions. In addition, due to occlusion, the interaction area is difficult to observe, and the visual features between the hand and the object are entangled, which makes it very difficult to reconstruct the deformation of the contact regions.

Most existing solutions [27, 32, 15, 48, 25] adopt fusion-based methods to reconstruct deformable objects. Some works propose specific strategies for interaction problems [42, 44], they rely on extra depth cameras to record slow motions. Besides the depth dependence, these fusion-based methods [42, 44] to reconstruct interactions do not model the deformable objects explicitly, and could hardly obtain instance mesh sequences with time-invariant topology. Commonly, they could only tackle the interaction between single pair of hand-object. Other existing template-based methods [34, 28, 11, 30, 39, 38] have difficulty in handling scenes where hands and objects are closely interacting. In our captured data, the human hands interact more closely with the deformable objects. This means that the interaction area is difficult to observe, the hand state needs to be reconstructed simultaneously, and the visual features between the hand and the object are entangled.

Aiming at the closely interactive motion reconstruction of hands and deformable objects, we propose a template-based method to annotate the data we captured. The method jointly reconstructs hand pose, object pose, and object deformation from captured data. All manipulated dolls are scanned and repaired in advance for their digital meshes with impermeable and homeomorphic properties. Our hand surface model also has higher resolution than MANO [29] and can more accurately represent the contact regions. The topological consistency of the object and hand meshes facilitates surface deformation analysis in our future studies. In terms of reconstruction algorithm design, the information of each viewing perspective is fully utilized to reconstruct the accurate hand, and the collaboration between the hand and the object is considered to guide the object pose estimation and the object deformation. Furthermore, a top-down strategy is adopted in our framework, so it theoretically supports reconstructing the interaction between multiple hands and deformable objects. Through comprehensive evaluation, the proposed method can reconstruct interactive motions of hands and deformable objects with high quality.

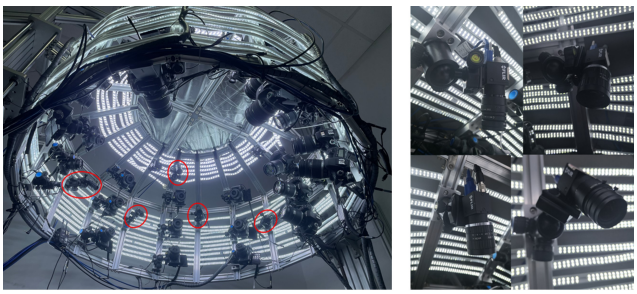
The main contributions of this work are summarized as follows.

- A markerless deformable interaction dataset recording interactive motions between hands and deformable dolls of various appearances and morphology. The dataset is publicly available on our website;
- A pipeline to reconstruct interactive motions of hands and deformable objects from multi-view data;
- An object deformation optimization algorithm under the guidance of hand and object collaboration.

2. Related Work

2.1. Hand and Object Interaction Datasets

In recent years, several datasets for hand and object interaction have already been proposed. [12] provided a dataset containing hand and object interactions, called FPHAB. They used a capture system consisting of magnetic sensors attached to the subject's hands and objects to obtain 3D annotations of the hands in RGB-D video sequences. However, since the magnetic sensors and the tape connecting them are visible, this changes the appearance of the hand in the color image. Hasson *et al*. [19] introduced ObMan, which provided a large dataset of synthetic images of hands grasped objects. HO-3D [16] used several RGB-D cameras to capture sequences and presented the first markerless dataset of color images with 3D annotations for both the hand and object. Following this, DexYCB [8] created a large-scale dataset of hands and rigid objects. Zhao *et al*. [46] proposed a hand-object interaction dataset containing physical properties and stability metrics. ContactPose [6] proposed to use a thermal camera to capture the contact map of objects to reflect the common contact regions of the grasped objects. Hampali *et al*. [24] created a dataset containing 3D annotations of objects manipulated



(a) (b)

Figure 2. **Hardware system.** (a) Multi-view synchronized capture system; (b) Industrial cameras.

by two hands. However, existing hand and object interaction datasets only include interactions between hands and rigid objects, and lack data recording the interactions of hands and deformable objects. We present HMDO, the first markerless dataset recording interactive motions between hands and deformable dolls of various appearances and morphologies.

2.2. Hand and Object Joint Reconstruction

Existing work mainly focuses on hand and rigid object reconstruction. Because rigid objects only have global degrees of freedom (DoFs), the interaction between hand and rigid objects is easier to model, capture, and reconstruct. With the popularity of learning-based methods in the fields of graphics and 3D vision, datasets recording hand-only [47, 13, 45], rigid-object-only [40, 20, 35, 26], and interaction between the two [12, 19, 6, 16, 8, 46, 24] have increased rapidly in both quantity and quality. This further promotes more literature to adopt data-driven methods to solve related problems. Hasson *et al*. [19] reconstructed the shape and pose of the hand-object through a unified network with extra synthetic data. [18] proposed a sparsely supervised learning method to reconstruct hand-object, exploiting the photometric consistency between sparsely supervised frames. Cao *et al*. [7] explored reconstructing hand-object interactions in the wild. Grady *et al*. [14] refine the estimated hand-object sate through contact prior learned from those datasets. Zhao *et al*. [46] makes the interaction state more stable by introducing the optimization process based on a physics engine. The datasets recording hand interactions with rigid objects also speed up the comparison and evolution of methods for other problems including grasping generation [23, 22] and manipulating planning [9, 43, 31].

2.3. Deformable Object Reconstruction

Precisely, rigid objects are only ideal approximations, while deformable objects are more common in daily life, *e.g.*backpacks, clothes, dolls, and even human bodies. Reconstructing such objects has always been a difficult problem in the field. As one of the mainstream solutions, template-based methods often build deformable objects as finite element models (FEM), which are determined by object-specific parameters and have high DoF in the calculation. Some researchers [39, 38] used sparse depth information to reconstruct deformation of single models. Some [34, 28] tried to generalize them to the scene containing weak hand-object interactions. Others [11, 30] obtain more deformation details by installing a depth camera and an extra force sensor on the robotic gripper at the same time. All these methods are difficult to generalize and apply to scenes where deformable objects interact closely with human hands. On the other hand, fusion-based methods [27, 32, 21, 15] abandon explicit modeling of objects and reconstruct the entire scene as a field with slow changes in depth and illumination. Some hybrid attempts [48, 41] divided the reconstruction process into two steps: online mesh template acquisition and real-time non-rigid reconstruction. Most of them take the human body, face, and hand as reconstruction objects, and do not consider the influence of occlusion. In recent years, some progresses [5, 4, 25] have been made when combining with learning-based technologies to find correspondences between two adjacent camera frames. And the studies [42, 44] specific to interaction problems to distinguish hands and objects from the scene. Nevertheless, most of the above methods still aim at real-time performance rather than high-quality, high-precision, and topology-consistent dataset preparation. Therefore, to our best knowledge, there exist no large-scale datasets recording the interaction between human hands and deformable objects. When the hand interacts with the deformable object, the deformation of the hand is much smaller than that of the object. Therefore, most methods use the rigid approximation of the hand to take the deformation of the object as the main contradiction. We also adopt this assumption.

3. Data Capture

3.1. System Configuration

We build a multi-view synchronized capture system to capture interactive motions of hands and deformable objects. The multi-view synchronized capture system uses hardware signals to trigger cameras shown in Fig. 2. The system includes three components: an industrial camera array, a synchronous signal generator, and data caching devices. Our camera array contains 10 high-speed industrial cameras. Each one is equipped with $8mm$ focal lens and produces 2048×1536 resolution images. The capturing frequency and framed amount can be adjusted through Bluetooth before capturing. When capturing each hand manipulation motion, the signal generator sends hardware trigger signals to each camera through the Ethernet patch cable. It controls each camera to shoot with a negligible delay. Once

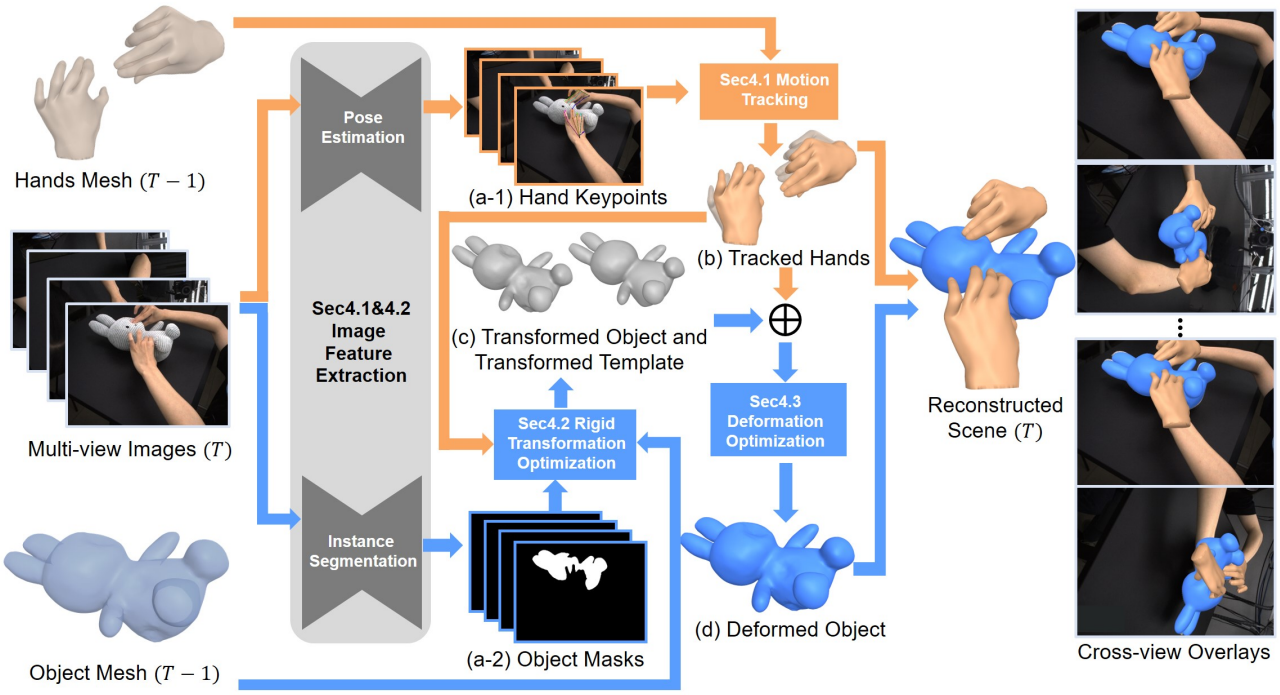


Figure 3. **Hand and deformable object reconstruction pipeline.** (a) Image features are first extracted from the current frame T, including the 2D poses of the hands and the masks of the object; (b) Hands motion tracking is performed based on image features and historical information; (c) The object global pose is estimated based on the current frame object masks and tracked hands, and the object mesh of the previous frame $T-1$; (d) The deformed object is obtained under the guidance of hand and object collaboration.

the capture is finished, the data from the camera is transferred to the caching devices. This solution is capable of capturing stable data with a frame rate up to 110 FPS.

3.2. Object Template Acquisition

As shown in Fig. 1, twelve zodiac dolls with various geometric and visual attributes are selected as the manipulation objects in our work. Although the doll comes from different manufacturers, we require that the elastic coefficient and density of the stuffing inside the doll should be as close as possible. In addition, during the data capture of hand-object interactions, we avoid pressure or tension that may cause plastic deformation for the dolls. We follow the steps to create a template object model for each doll. First, we adopt the fusion-based method [1] with a single RealSense435i device to capture a coarse object mesh, which shown as Fig. 5. Then a series of repairing processes including hole filling, isolated elements removal, and isotropic remeshing is adopted to guarantee that the whole object mesh is watertight and has genus 0 (homeomorphic to a sphere). After manual repair, these meshes with good geometric properties are used as template object models.

4. Hand-Deformable Object Reconstruction

An overview of our pipeline for reconstructing hand and deformable object interactions from multi-view motion data is shown in Fig. 3. Firstly, the method for hand tracking is described in Sec. 4.1. Then, the method for object global pose estimation is given in Sec. 4.2. In Sec. 4.3, we describe the optimization strategy for obtaining the deformable object by iteratively analyzing the relationship between hand and object. To facilitate the identification, for the object variables, the variables marked with hats superscript represent the results after global pose transformation, and the variables with tilde superscript represent the results after non-rigid deformation optimization. For the hand variables, the hand after optimization is represented by a tilde superscript.

4.1. Hand Motion Tracking

Hand pose estimation. First, we use the 2D pose estimation network [37] to estimate the 2D keypoints and confidences for each camera of the current frame. We then solve the following equation to obtain the 3D keypoints by utilizing camera parameters and the estimated 2D keypoints,

$$\underset{k}{\operatorname{argmax}} \sum_{n \in \mathcal{N}} \left\| [X_n]_{\times} \cdot \Pi_n \left(R_n k + T_n \right) \right\|, \tag{1}$$

where $[X_n]_{\times}$ is the skew matrix of the homogenous coordinate of X_n, which describes the 2D coordinate of hand keypoint in the n-th camera. k is the 3D hand keypoint. Π_n and $[R_n, T_n]$ are the intrinsic parameter and extrinsic

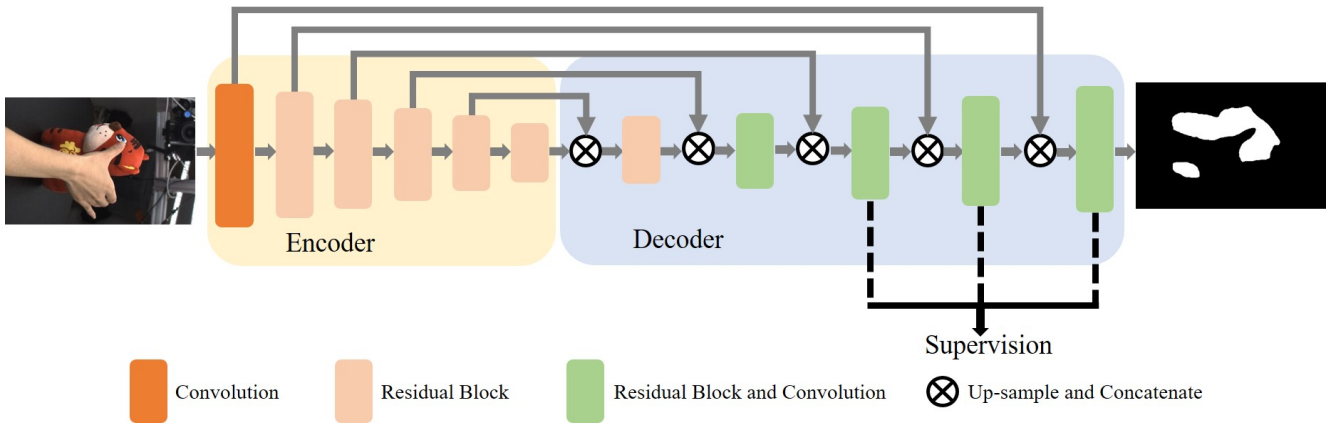


Figure 4. **Overview of object segmentation network.** We designed an object segmentation network, and the network used different scales of masks as supervision.

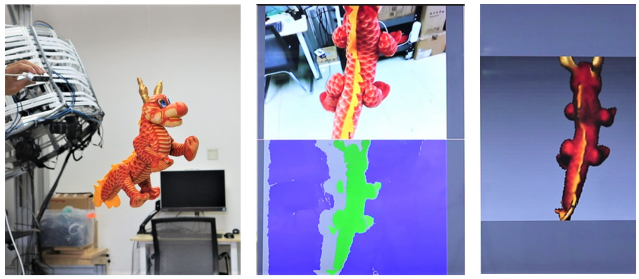


Figure 5. **Template acquisition.** (a) Reference object; (b) Scanning process; (d) Reconstructed coarse template.

parameter of the n-th camera, respectively. $\mathcal{N}$ is the set of valid camera views for this keypoint. It is worth noting that not all views have accurate 2D keypoints. For each camera, only when the confidence of the 2D keypoint is greater than a threshold, it will be regarded as a valid camera in the corresponding 3D keypoint calculation. In our experiment, the threshold was set to 0.6.

Hand mesh optimization. We utilize the multi-view system to pre-optimize our hand surface model to obtain personalized hand shape parameters of subjects. During hand and deformable object tracking, only the pose $\boldsymbol{\theta}$ of the hand is optimized. We minimize the errors of deformed skeleton joints and estimated keypoints in both 3D and 2D space,

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_{j} \left(\sum_{n \in \mathcal{N}} \|x_{j,n} - \mathcal{P}_n\left(f_j(\boldsymbol{\theta})\right)\| + \|k_j - f_j(\boldsymbol{\theta})\| \right), \tag{2}$$

Where $x_{j,n}$ is the j-th 2D keypoint in the n-th camera, and k_j is the j-th 3D keypoint. $f_j(\boldsymbol{\theta}) \in \mathbb{R}^{3\times 1}$ represents the 3D position of the j-th hand skeleton joint with the parameter $\boldsymbol{\theta}$, and $\mathcal{P}_n(\cdot)$ can be expressed as:

$$\mathcal{P}_n\left(f_j(\boldsymbol{\theta})\right) = R_n f_j(\boldsymbol{\theta}) + T_n. \tag{3}$$

After getting the hand pose $\boldsymbol{\theta}$ of the current frame, We use historical information to do a temporal smoothing filter to obtain the smoothed pose $\tilde{\boldsymbol{\theta}}$. We adopt the linear blend skinning to deform our hand surface model M_h. Specifically, for the i-th vertex v_h^i in the hand mesh, the deformed new position $\tilde{v}_h^i$ is computed as:

$$\tilde{v}_h^i = \sum_j \omega\left(v_h^i, f_j(\tilde{\boldsymbol{\theta}})\right)\left[R_j\left(v_h^i - f_j(\tilde{\boldsymbol{\theta}})\right) + f_j(\tilde{\boldsymbol{\theta}}) + t_j\right] \tag{4}$$

where $[R_j, t_j]$ is the rigid transformation of the j-th bone, which is only determined by the hand pose $\tilde{\boldsymbol{\theta}}$, the skinning weight $\omega\left(v_h^i, f_j(\tilde{\boldsymbol{\theta}})\right)$ is computed by heat-based method, which measures the influence of the j-th bone to the i-th vertex.

4.2. Object Pose Estimation

Object segmentation. We design an encoder-decoder network for object segmentation as shown in Fig. 4. We use different scales of masks as supervision. Through this network, the object makes $\{S_n\}$ of the current frame can be obtained. S_n represents the mask of the n-th view.

Object pose optimization. We use the genetic mutation algorithm to iteratively solve the global position of the object. In each iteration, we first select samples that are inherited to the next generation, and then we use the uniform distribution to simulate the mutation process based on the selected samples to obtain new samples for the next generation. Among them, we use the roulette method to determine the samples that can be inherited to the next generation. The smaller loss value of the sample, the greater the probability of inheritance to the next generation. The loss function for each sample is calculated as follows:

$$L = \sum_n \mathcal{D}(M(\boldsymbol{\alpha}), \tilde{M}_h, S_n) + \lambda_o \|\boldsymbol{\alpha}\|_2 \tag{5}$$

Where the first term is the reprojection error of the object, and the second term is the regularization term. $\boldsymbol{\alpha} \in \mathbb{R}^{6\times 1}$ is

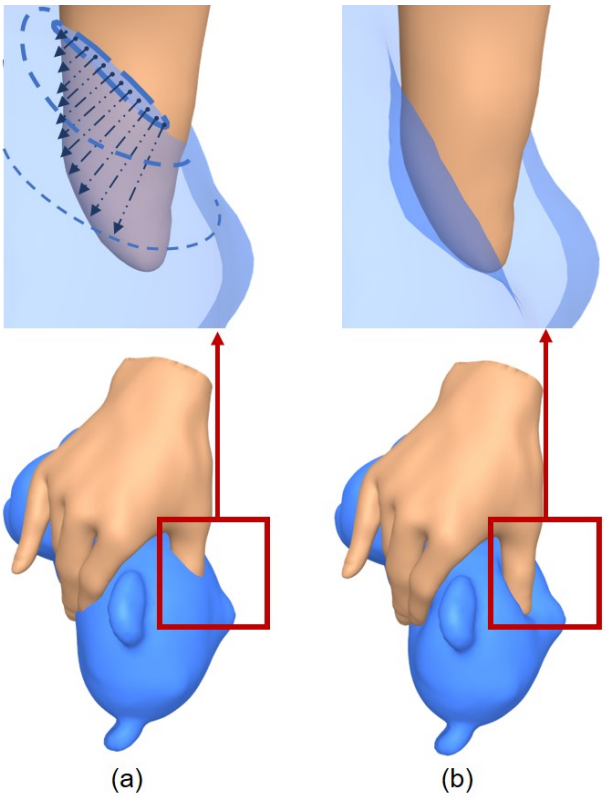


Figure 6. **Deformation schematic diagram.** (a) Result before deformation; (b) Result after deformation.

the features of one sample. The first three dimensions represent the rotation, where we use axis-angle representation to describe the global rotation. The last three dimensions represent the translation. The first term can be expressed as:

$$\mathcal{D}(M(\boldsymbol{\alpha}), \tilde{M}_h, S_n) = 1 - \frac{\mathcal{H}_n\left(\tilde{M}_h, M(\boldsymbol{\alpha})\right) \cap S_n}{\mathcal{H}_n\left(\tilde{M}_h, M(\boldsymbol{\alpha})\right) \cup S_n} \quad (6)$$

where, $\tilde{M}_h$ is the hand mesh of the current frame we obtained in 4.1, and $M(\boldsymbol{\alpha})$ represents the object mesh with the parameter $\boldsymbol{\alpha}$, i.e., global rigid transformation. $\mathcal{H}_n\left(\tilde{M}_h, M(\boldsymbol{\alpha})\right)$ is the rendered mask of the object mesh in the n-th camera that is not occluded by the hand mesh. We use the global pose of the previous frame as the initial one in our optimization.

Our goal is to minimize the overall loss, which is the sum of L for all samples. Through iterative optimization, we obtain the object global pose that converges to the optimal solution. We then use the historical information to perform a temporal smoothing filter on the optimized pose. Since our method is a global optimization strategy, it does not accumulate temporal errors.

4.3. Object Deformation Optimization

To model non-rigid deformation of deformable objects, we follow the representation in embedded deformation graphs [33]. The non-rigid deformation is represented by the affine transformation $\{A_s, t_s\}$ of uniformly selected vertices $\{g_s\}$ on the mesh, which are treated as nodes of the graph. For the vertex $\hat{v}_o$ in the globally transformed object template mesh, the new position of the deformation $\tilde{v}_o$ is computed as:

$$\tilde{v}_o = \sum_{g_s \in \mathcal{S}(\hat{v}_o)} \omega\left(\hat{v}_o, g_s\right)\left[A_s\left(\hat{v}_o - g_s\right) + g_s + t_s\right] \quad (7)$$

where $\mathcal{S}(\hat{v}_o)$ is the neighbor nodes of mesh vertex $\hat{v}_o$. $\omega(\hat{v}_o, g_s)$ is the deformation weights of node g_s to $\hat{v}_o$, which represents the influence of the node on the vertex. The definition of vertex neighbor nodes and the calculation of deformation weights refer to [33]. To get the transformation $\{A_s, t_s\}$ of all nodes in the deformation graph, we estimate them by optimizing the following function:

$$E = \lambda_1 E_{\text{cont}} + \lambda_2 E_{\text{silh}} + \lambda_3 E_{\text{temp}} + \lambda_4 E_{\text{rigid}} + \lambda_5 E_{\text{reg}}. \quad (8)$$

Contact term. E_{cont} is the contact term. By analyzing the relationship between the hand and the object, the object is deformed to match the contact, resulting in a reasonable manipulation. This term can be expressed as:

$$E_{\text{cont}} = \sum_i ||\tilde{v}_o^i - v_{target}^i||. \quad (9)$$

where $\tilde{v}_o^i$ is the deformed position of $\hat{v}_o^i$, and v_{target}^i is the target position of $\hat{v}_o^i$ by analyzing the relationship between the hand and the object. The object vertices target position are obtained by the following steps. After rigid transformation of the current frame object in Sec. 4.2, rays are emitted from the object to the hand for intersection detection. If penetration occurs, we record the vertex on the object, as well as the first intersection with the hand, which are marked as $(\hat{p}_o, \tilde{p}_h)$. The standard 3D Axis-Aligned Bounding Box [3] tree is used to speed up the process. We set the target position of these penetrated vertex $\{\hat{p}_o\}$ to $\{\tilde{p}_h\}$. The regions squeezed by the hand affect the surrounding regions. We employ a strategy based on geodesic distance and penetration depth to diffuse deformation around. The target positions of the remaining vertices can be expressed as:

$$v_{target} = \hat{v}_o - \frac{1}{N}\sum_{i=0}^{N} \mathcal{I}(\hat{v}_o, \hat{p}_o^i) * \vec{n} \quad (10)$$

where, $\hat{v}_o$ is the vertex position of the object before deformation, and N represents the number of penetrated vertex affecting vertex $\hat{v}_o$. $\vec{n}$ is the unit normal vector of vertex $\hat{v}_o$. $\mathcal{I}(\cdot)$ is the impact factor, which can be expressed as:

$$\mathcal{I}(\hat{v}_o, \hat{p}_o) = d\left(\hat{p}_o\right) * \exp(-\lambda_c * \mathcal{G}(\hat{v}_o, \hat{p}_o)) \quad (11)$$

where, $d(\hat{p}_o)$ represents the penetration depth of vertex $\hat{p}_o$, and $\mathcal{G}(\cdot)$ is the geodesic distance. Regarding the calculation of geodesic distance, we refer to [10]. In our experiments, when $\mathcal{I}(\cdot)$ is less than 0.02, we consider the vertex $\hat{v}_o$ are not affected by the vertex $\hat{p}_o$.

Silhouette term. The E_{silh} term constrains the projection of the object model under each camera perspective to be consistent with the contour of the observed images.

$$E_{\text{silh}} = \sum_{\hat{v}_o \in \mathcal{C}} \sum_{n \in \mathcal{N}(\hat{v}_o)} \left\| \mathcal{P}_n(\tilde{v}_o) - c_{\hat{v}_o,n} \right\| \tag{12}$$

where $\mathcal{C}$ includes all the vertices that have corresponding contours in the observed images. $\mathcal{N}(\hat{v}_o)$ records all camera numbers for which $\hat{v}_o$ has corresponding contour points in the observed camera perspective. $c_{\hat{v}_o,n}$ is the corresponding contour point of vertex $\hat{v}_o$ under the observed image of n-th camera. $\tilde{v}_o$ is the deformation result of $\hat{v}_o$, and $\mathcal{P}_n(\tilde{v}_o)$ is the 2d projection of $\tilde{v}_o$. For methods of finding the matching 2d pixels on the image plane and retrieving the 3D position of 2D image coordinates, we refer to [36].

Temporal smoothing term. The temporal smoothing term E_{temp} encourages smooth deformation from frame to frame, which can be expressed as:

$$E_{\text{temp}} = \sum_i ||\tilde{v}_o^i - \hat{v}_{last}^i||. \tag{13}$$

where $\tilde{v}_o^i$ is the deformed position of $\hat{v}_o^i$, and $\hat{v}_{last}^i$ is the corresponding vertex on the object mesh of the previous frame after the global transformation in Sec. 4.2.

Rigid term. The term E_{rigid} used to restrict the affine transformation to be as rigid as possible, which is the same [33] and is formulated as:

$$\begin{aligned} E_{\text{rigid}} = & \sum_s \left(\left(\mathbf{a}_{s,1}^T \mathbf{a}_{s,2} \right)^2 + \left(\mathbf{a}_{s,1}^T \mathbf{a}_{s,3} \right)^2 + \left(\mathbf{a}_{s,2}^T \mathbf{a}_{s,3} \right)^2 \right) \\ & + \sum_s \sum_i \left(\left(1 - \mathbf{a}_{s,i}^T \mathbf{a}_{s,i} \right)^2 \right), \end{aligned} \tag{14}$$

where $\mathbf{a}_{s,1}$, $\mathbf{a}_{s,2}$ and $\mathbf{a}_{s,3}$ are the column vectors of A_s.

Regularization term. The term E_{reg} is served as a regularizer for the deformation by indicating that the affine transformations of adjacent graph nodes should agree with one another. Specifically, it means that the affine transformation of node g_m is applied to node g_n, which should be consistent with the affine transformation of node g_n being applied to itself.

$$\begin{aligned} E_{\text{reg}} = & \sum_m \sum_{g_n \in \mathcal{S}(g_m)} \omega(g_n, g_m) \, || g_m + t_m \\ & + A_m (g_n - g_m) - (g_n + t_n) \, ||. \end{aligned} \tag{15}$$

We optimize the function Eq. 8 and update the object mesh at the end of each iteration. The deformation schematic diagram is shown in Fig. 6.

5. Experiments

5.1. Implementation Details

All of our experiments are performed on a computer configured with an Intel i7- 12700 CPU, NVIDIA GeForce RTX 3090 GPU. Our hand surface model has 6829 vertices and 13654 faces. The number of vertices of our object templates is between 6000 and 13000, and the number of faces is between 12000 and 25000.

Hand pose estimation. The architecture of our 2D hand pose estimation network is based on SRNet [37]. We use the existing hand and rigid object interaction datasets HO-3D [16], DexYCB [8], CBF [46] and ContactPose [6] as our training dataset. During network training, we use the SGD optimizer with the learning rate set to 10^{-5}. After training, we collect data in our synchronized multi-view capture system, estimate 2D hand pose from these data using the trained model, and manually adjust for the incorrect poses. We then fine-tune the hand 2D pose estimation model on these adjusted data. The 3D pose is estimated from the multi-view 2D keypoints, and we use the LDLT method to solve it. The non-linear optimization of hand surface model is solved by Ceres [2].

Object mask segmentation. During network training, we use the SGD optimizer, the network learning rate is set to 10^{-5}, and the MSE loss is used for supervision. Regarding the training data, for each object, we first use a depth camera to collect coarse data through distance threshold segmentation, and we use color threshold segmentation and manual processing to obtain accurate masks. In addition, We perform data augmentation on these data, including random combinations with data from existing hand datasets, geometric transformations, and color transformations of the images.

Object global pose optimization. In the first frame, we use the uniform distribution to initialize the samples, the population size of samples is set to 500, and the number of iterations is set to 20. In subsequent frames, initial population distribution and population size remain the same, but only 1 iteration is performed.

Object deformation. The weight $\lambda_c = 0.2$, $\lambda_1 = 5$, $\lambda_2 = 5$,$\lambda_3 = 1$, $\lambda_4 = 1$, $\lambda_5 = 2$. The non-linear optimization of object deformation is solved by Ceres [2].

5.2. Qualitative Results

We show the reconstruction results from the captured motion data in Fig. 1, Fig. 8 and Fig. 7. In Fig. 1, we show the interaction between the hand and the tiger. In Fig. 8, we show the interaction between the hand and the rabbit. We capture at a high frame rate and there is less variation from frame to frame, so to reflect the difference, the displayed frames are shown at specific time intervals. Fig. 7 shows randomly selected frames in the interactive motion

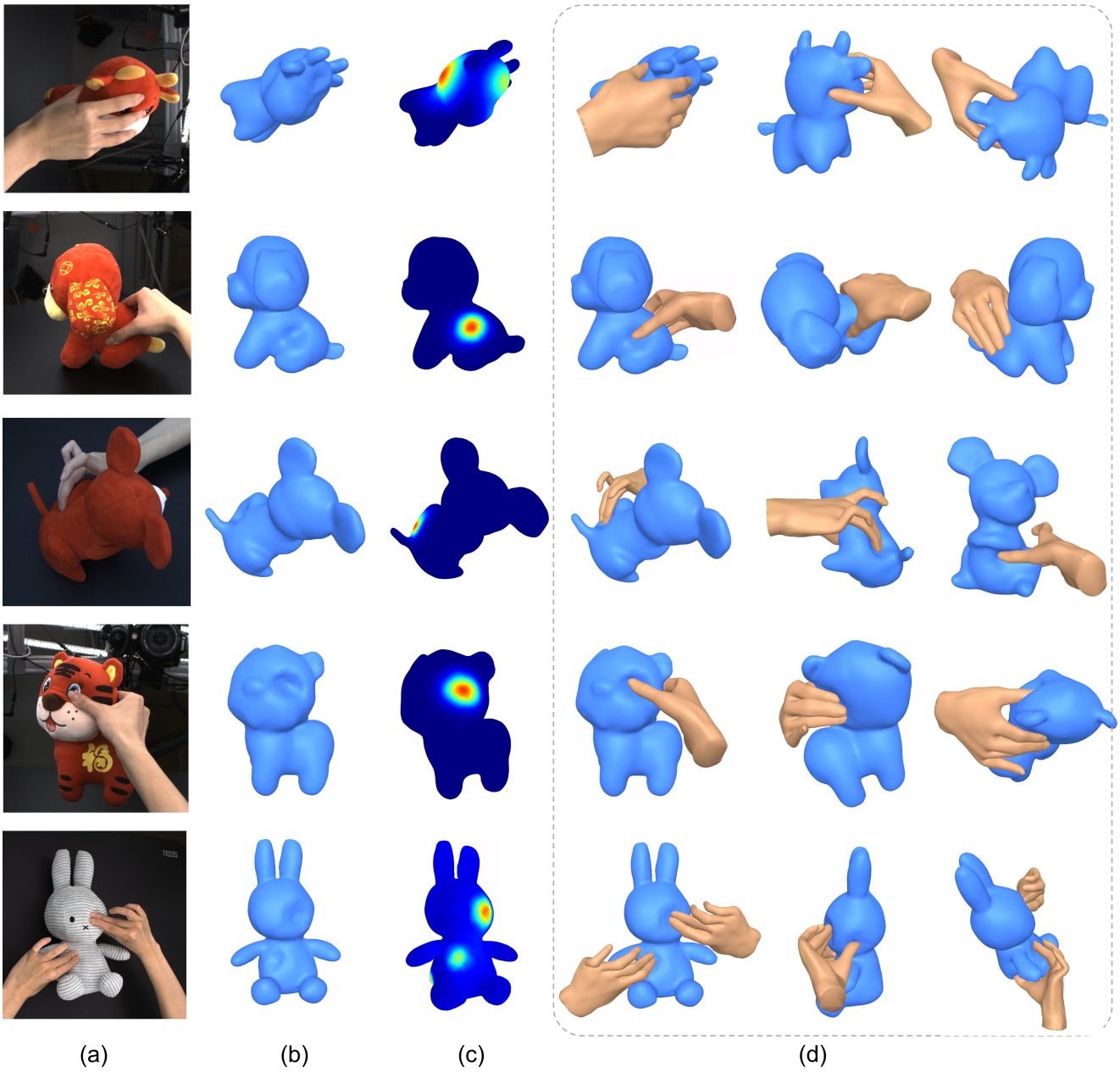


Figure 7. **Qualitative results of our method on different objects.** Randomly selected frames from hands and objects interaction sequences. (a) Input frames; (b) Object meshes; (c) Contact deformation maps; (d) Hand and object meshes (3 views).

of the hand and different objects. From these reconstruction results, we can see that our solution is able to track and reconstruct the interaction of hands and deformable objects in various poses with high quality. Manipulated objects can vary greatly in visual and shape. It is worth mentioning that our method can handle multiple hands interacting with objects, as our framework adopts a top-down strategy. In Fig. 7 we show the interaction result between hands and rabbits. In addition, in Fig. 9, we show the reconstruction of the hand and plastic water bottle interactions using the proposed method.

5.3. Evaluation of Hand 3D Pose Estimation

Estimating accurate hand pose is important for hand motion tracking, while also correctly guiding rigid transformations and non-rigid deformations of objects. To quantitatively evaluate the accuracy of the hand pose estimated by our annotation method, we manually annotated the 3D lo-

Methods	4 views	6 views	8 views	10 views
Mean.(mm)	14.75	10.66	7.31	5.58
Std.(mm)	6.81	4.53	3.64	2.29

Table 1. **Evaluations the accuracy of hand pose estimation.** We report the average hand joint errors for different camera number settings.

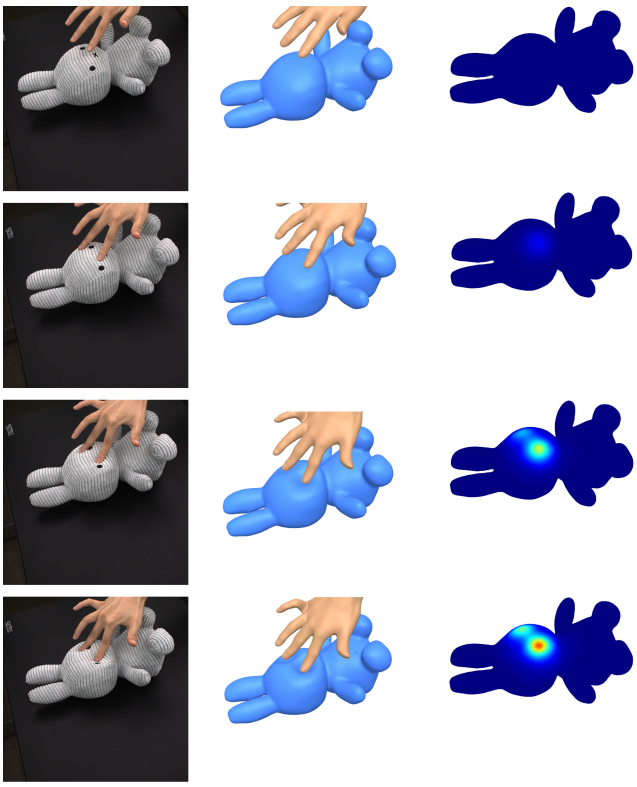


Figure 8. **Reconstruction results on the sequence "Rabbit".** (a) Selected frames; (b) Reconstruction results; (c) Contact deformation distribution maps.

cations of the 3D joints in randomly selected frames. Tab. 1 shows the estimated hand pose accuracy for different numbers of views. From the results in Tab. 1, as the number of camera views increases, the estimation error gradually decreases. We can achieve an average joint error accuracy of lower than 6mm on average with all camera view settings.

5.4. Evaluation of Object Pose Estimation

The population size, initial population distribution, and optimization iterations number can affect the performance of our object global pose estimation algorithm. To evaluate the impact of these parameters on object pose estimation, we conduct qualitative and quantitative experiments. For experiments, we manually annotated object masks in 2 sequences. In these evaluation samples, the objects have no non-rigid deformations. We use time-consuming and the mean intersection over union(mIoU) to evaluate the performance of object pose estimation under different parameter settings. The mIoU is the mean of ratio of the intersection and union of the predicted value and the true value. We measure the difference between the rendered mask of the object mesh and ground-truth.

We first treat these data as independent initial frames and optimize them to obtain object pose. The effect of different population size is compared in the first 4 rows of Tab. 2. When the population size is set to 500, similar results can be obtained with the population size set to 700, and the time is shorter. The middle four rows of Tab. 2 evaluate the effect of the number of iterations on the results. Satisfactory results can be achieved with 20 iterations. The experiments on the effect of the initial population distribution on the results are shown in the last two rows of Tab. 2. It can be seen from the table that the initial population set to uniform distribution has higher accuracy of pose estimation than a

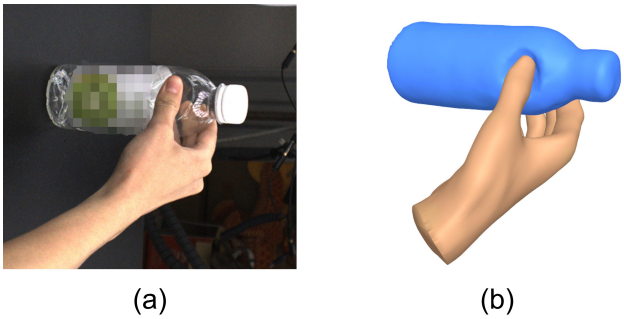


Figure 9. **Reconstruction result of plastic water bottle**. (a) Test frame; (b) Reconstruction result.

Parameters	mIoU.(%) ↑	Time. (s) ↓
size = 100, iter = 20, $\mathcal{U}$	83.33	17.72
size = 300, iter = 20, $\mathcal{U}$	84.72	49.76
size = 500, iter = 20, $\mathcal{U}$	86.91	82.70
size = 700, iter = 20, $\mathcal{U}$	86.90	123.59
size = 500, iter = 5, $\mathcal{U}$	58.36	20.72
size = 500, iter = 10, $\mathcal{U}$	82.98	43.27
size = 500, iter = 20, $\mathcal{U}$	86.91	82.70
size = 500, iter = 30, $\mathcal{U}$	86.79	120.18
size = 500, iter = 20, $\mathcal{O}$	84.74	84.47
size = 500, iter = 20, $\mathcal{U}$	86.91	82.70

Table 2. **Evaluation of object pose estimation accuracy in initial frames with different parameter settings.** $\mathcal{U}$ represents uniform distribution, and $\mathcal{O}$ represents normal distribution. "size" represents population size, and "iter" represents optimization iterations number.

Terms	Initialization	E_{cont}	$E_{\text{cont}} + E_{\text{reg}}$	$E_{\text{cont}} + E_{\text{reg}} + E_{\text{rigid}}$	$E_{\text{cont}} + E_{\text{reg}} + E_{\text{rigid}} + E_{\text{temp}}$	$E_{\text{cont}} + E_{\text{reg}} + E_{\text{rigid}} + E_{\text{temp}} + E_{\text{silh}}$
mIoU.(%) ↑	78.54	83.26	84.92	85.44	85.61	87.53
Inter.(cm^3) ↓	15.10	6.13	4.05	3.52	3.64	3.27

Table 3. **Evaluation of terms for object deformation optimization.** "Initialization" denotes the object mesh before deformation.

Parameters	mIoU.(%) ↑	Time. (s) ↓
size = 500, iter = 1, $\mathcal{U}$	84.63	4.17
size = 500, iter = 3, $\mathcal{U}$	84.82	12.53

Table 4. **Evaluation of object pose estimation accuracy in non-initial frames with different parameter settings.** $\mathcal{U}$ represents uniform distribution. "size" represents population size, and "iter" represents optimization iterations number.

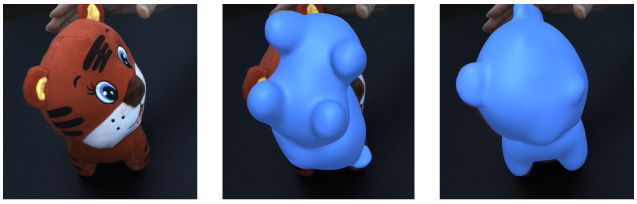


Figure 10. **Incorrect object pose estimation**. (a) Test frame; (b) Result with incorrect pose; (c) Result with correct pose. Inappropriate parameter settings are easy to fall into local optimal solution, resulting in incorrect object pose estimation.

normal distribution. This is because the initial frame has no prior information. In other words, there is no guidance for the initial pose, so using uniform distribution is less likely to fall into local optimal solutions than a normal distribution. As shown in Fig. 10, inappropriate parameter settings are easy to fall into the local optimal solution, resulting in incorrect object pose estimation.

For non-initial frames, we use the result of the previous frame as the initial value to optimize the object pose. As we collect data at a high frame rate, there is less variation from frame to frame. We evaluate the effects of iteration number and initial population distribution on non-initial frame pose estimation with two sequences that are manually annotated. As shown in Tab. 4, setting the number of iterations to 1 can quickly converge to a suitable solution.

5.5. Evaluation of Object Deformation Optimization

We perform ablation experiments on the terms in Eq. 8. Regarding the evaluation metrics, We use mIoU to evaluate the quality of deformed object reconstruction results. In addition, *intersection volume* (denoted as Inter. in tables) proposed in [19] is adopted to evaluate the contact quality between hand and deformed object. The experimental results are shown in Tab. 3. Although the term E_{temp} does not help improve the contact quality, it results in smoother

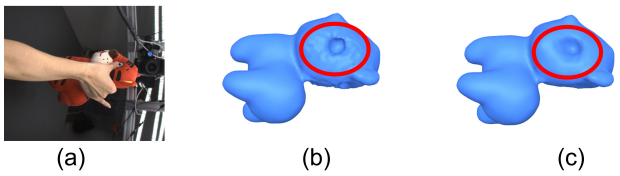


Figure 11. **Evaluation of the regularization term for object deformation.** (a) Reference frame; (b) Result without the regularization term; (c) Result with the regularization term.

deformation from frame to frame. Satisfactory reconstruction quality can be achieved with all terms introduced. In addition, we show the qualitative evaluation of the regularization term for object deformation in Fig. 11. As shown in Fig. 11 (b), without the regularization term, the transformations of adjacent graph nodes are inconsistent, which leads to unnatural deformation. After introducing the regularization term, a reasonable result is obtained.

6. Conclusion

We construct the first dataset to record the interactive motion of hands and deformable objects to fill the gap in the hand and deformable object datasets. It captures deformable interactions in multiple interaction forms from 10 perspectives with our multi-view capture system. We propose a method to annotate our captured motion data. The method makes full use of information from various perspectives to reconstruct the accurate hand, and the collaboration between the hand and the object is considered to guide the object pose estimation and the object deformation. Through comprehensive evaluation, we demonstrate that our method can reconstruct interactive motions of hands and different deformable objects with high quality. In the future, this dataset can be used for research on hand and deformable object reconstruction.

Limitations and Future Work. We constructed a dataset containing different forms of deformable interactions, where the main focus is the non-rigid contact deformation of interacting objects. The interacting objects in our dataset do not have large deformations, such as 180-degree twisting or bending. The proposed hand and deformable object reconstruction method requires the material of deformable objects to be uniform, otherwise our deformation diffusion strategy may not work properly. In the future, we will add large deformations of objects, and consider introducing depth and color information.

References

[1] Recfusion: Create 3d models in real-time with rgb-d sensors. https://www.recfusion.net/. 4

[2] S. Agarwal and K. Mierle. Others,"ceres solver,". *Avaliable: http://ceres-solver. org*, 2015. 7

[3] P. Alliez, S. Tayeb, and C. Wormser. 3d fast intersection and distance computation (aabb tree). *CGAL User and Reference Manual*, 4, 2012. 6

[4] A. Bozic, P. Palafox, M. Zollhöfer, A. Dai, J. Thies, and M. Nießner. Neural non-rigid tracking. *Advances in Neural Information Processing Systems*, 33:18727–18737, 2020. 3

[5] A. Bozic, M. Zollhofer, C. Theobalt, and M. Nießner. Deepdeform: Learning non-rigid rgb-d reconstruction with semi-supervised data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7002–7012, 2020. 3

[6] S. Brahmbhatt, C. Tang, C. D. Twigg, C. C. Kemp, and J. Hays. Contactpose: A dataset of grasps with object contact and hand pose. In *European Conference on Computer Vision*, pages 361–378. Springer, 2020. 2, 3, 7

[7] Z. Cao, I. Radosavovic, A. Kanazawa, and J. Malik. Reconstructing hand-object interactions in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12417–12426, 2021. 3

[8] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield, et al. Dexycb: A benchmark for capturing hand grasping of objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9044–9053, 2021. 2, 3, 7

[9] S. Christen, M. Kocabas, E. Aksan, J. Hwangbo, J. Song, and O. Hilliges. D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions. *arXiv preprint arXiv:2112.03028*, 2021. 3

[10] K. Crane, C. Weischedel, and M. Wardetzky. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Transactions on Graphics (TOG)*, 32(5):1–11, 2013. 7

[11] B. Frank, R. Schmedding, C. Stachniss, M. Teschner, and W. Burgard. Learning the elasticity parameters of deformable objects with a manipulation robot. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1877–1883. IEEE, 2010. 2, 3

[12] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 409–419, 2018. 2, 3

[13] F. Gomez-Donoso, S. Orts-Escolano, and M. Cazorla. Large-scale multiview 3d hand pose dataset. *Image and Vision Computing*, 81:25–33, 2019. 3

[14] P. Grady, C. Tang, C. D. Twigg, M. Vo, S. Brahmbhatt, and C. C. Kemp. Contactopt: Optimizing contact to improve grasps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1471–1481, 2021. 3

[15] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu. Real-time geometry, albedo, and motion reconstruction using a single rgb-d camera. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017. 2, 3

[16] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3196–3206, 2020. 2, 3, 7

[17] S. Hampali, S. D. Sarkar, M. Rad, and V. Lepetit. Handsformer: Keypoint transformer for monocular 3d pose estimation ofhands and object in interaction. *arXiv preprint arXiv:2104.14639*, 2021. 2

[18] Y. Hasson, B. Tekin, F. Bogo, I. Laptev, M. Pollefeys, and C. Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 571–580, 2020. 3

[19] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11807–11816, 2019. 2, 3, 10

[20] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision*, pages 548–562. Springer, 2012. 3

[21] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger. Volumedeform: Real-time volumetric non-rigid reconstruction. In *European Conference on Computer Vision*, pages 362–379. Springer, 2016. 3

[22] H. Jiang, S. Liu, J. Wang, and X. Wang. Hand-object contact consistency reasoning for human grasps generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11107–11116, 2021. 3

[23] K. Karunratanakul, J. Yang, Y. Zhang, M. J. Black, K. Muandet, and S. Tang. Grasping field: Learning implicit representations for human grasps. In *2020 International Conference on 3D Vision (3DV)*, pages 333–344. IEEE, 2020. 3

[24] T. Kwon, B. Tekin, J. Stühmer, F. Bogo, and M. Pollefeys. H2o: Two hands manipulating objects for first person interaction recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10138–10148, 2021. 2, 3

[25] W. Lin, C. Zheng, J.-H. Yong, and F. Xu. Occlusionfusion: Occlusion-aware motion estimation for real-time dynamic 3d reconstruction. *arXiv preprint arXiv:2203.07977*, 2022. 2, 3

[26] X. Liu, S. Iwase, and K. M. Kitani. Stereobj-1m: Large-scale stereo image dataset for 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10870–10879, 2021. 3

[27] R. A. Newcombe, D. Fox, and S. M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 343–352, 2015. 2, 3

[28] A. Petit, S. Cotin, V. Lippiello, and B. Siciliano. Capturing deformations of interacting non-rigid objects using rgb-d data. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 491–497. IEEE, 2018. 2, 3

[29] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (ToG)*, 36(6):1–17, 2017. 2

[30] A. Sengupta, R. Lagneau, A. Krupa, E. Marchand, and M. Marchal. Simultaneous tracking and elasticity parameter estimation of deformable objects. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10038–10044. IEEE, 2020. 2, 3

[31] Q. She, R. Hu, J. Xu, M. Liu, K. Xu, and H. Huang. Learning high-dof reaching-and-grasping via dynamic representation of gripper-object interaction. *arXiv preprint arXiv:2204.13998*, 2022. 3

[32] M. Slavcheva, M. Baust, D. Cremers, and S. Ilic. Killingfusion: Non-rigid 3d reconstruction without correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1395, 2017. 2, 3

[33] R. W. Sumner, J. Schmid, and M. Pauly. Embedded deformation for shape manipulation. In *ACM siggraph 2007 papers*, pages 80–es. 2007. 6, 7

[34] A. Tsoli and A. A. Argyros. Joint 3d tracking of a deformable object in interaction with a hand. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 484–500, 2018. 2, 3

[35] D. Tzionas and J. Gall. 3d object reconstruction from hand-object interactions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 729–737, 2015. 3

[36] Y. Wang, R. Rao, and C. Zou. Personalized hand modeling from multiple postures with multi-view color images. In *Computer Graphics Forum*, volume 39, pages 339–350. Wiley Online Library, 2020. 7

[37] Y. Wang, B. Zhang, and C. Peng. Srhandnet: Real-time 2d hand pose estimation with simultaneous region localization. *IEEE transactions on image processing*, 29:2977–2986, 2019. 4, 7

[38] S. Weiss, R. Maier, D. Cremers, R. Westermann, and N. Thuerey. Correspondence-free material reconstruction using sparse surface constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4686–4695, 2020. 2, 3

[39] S. Wuhrer, J. Lang, M. Tekieh, and C. Shu. Finite element based tracking of deforming surfaces. *Graphical Models*, 77:1–17, 2015. 2, 3

[40] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 3

[41] R. Yu, C. Russell, N. D. Campbell, and L. Agapito. Direct, dense, and deformable: Template-based non-rigid 3d reconstruction from rgb video. In *Proceedings of the IEEE international conference on computer vision*, pages 918–926, 2015. 3

[42] H. Zhang, Z.-H. Bo, J.-H. Yong, and F. Xu. Interactionfusion: real-time reconstruction of hand poses and deformable objects in hand-object interactions. *ACM Transactions on Graphics (TOG)*, 38(4):1–11, 2019. 2, 3

[43] H. Zhang, Y. Ye, T. Shiratori, and T. Komura. Manipnet: neural manipulation synthesis with a hand-object spatial representation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 3

[44] H. Zhang, Y. Zhou, Y. Tian, J.-H. Yong, and F. Xu. Single depth view based real-time reconstruction of hand-object interactions. *ACM Transactions on Graphics (TOG)*, 40(3):1–12, 2021. 2, 3

[45] Z. Zhao, T. Wang, S. Xia, and Y. Wang. Hand-3d-studio: A new multi-view system for 3d hand reconstruction. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2478–2482. IEEE, 2020. 3

[46] Z. Zhao, B. Zuo, W. Xie, and Y. Wang. Stability-driven contact reconstruction from monocular color images. *arXiv preprint arXiv:2205.00848*, 2022. 2, 3, 7

[47] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 813–822, 2019. 3

[48] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (ToG)*, 33(4):1–12, 2014. 2, 3