CPF: Learning a Contact Potential Field to Model the Hand-object Interaction

Lixin Yang, Xinyu Zhan, Kailin Li, Wenqiang Xu, Jiefeng Li, Cewu Lu
Shanghai Jiao Tong University, Shanghai, China
{siriusyang, kelvin34501, kailinli, vinjohn, ljf_likit, lucewu}@sjtu.edu.cn

Abstract

Estimating hand-object (HO) pose during interaction has been brought remarkable growth in virtue of deep learning methods. Modeling the contact between the hand and object properly is the key to construct a plausible grasp. Yet, previous works usually focus on jointly estimating HO pose but not fully explore the physical contact preserved in grasping. In this paper, we present an explicit contact representation, Contact Potential Field (CPF) that models each hand-object contact as a spring-mass system. Then we can refine a natural grasp by minimizing the elastic energy w.r.t those systems. To recover CPF, we also propose a learning-fitting hybrid framework named MIHO. Extensive experiments on two public benchmarks have shown that our method can achieve state-of-the-art in several reconstruction metrics, and allow us to produce more physically plausible HO pose even when the ground-truth exhibits severe interpenetration or disjointedness. Our code is available at https://github.com/lixiny/CPF.

1. Introduction

Being able to model hand-object interaction from a single image is essential for understanding the behavior of humans. And simulating a natural plausible grasp based on the estimated pose is also crucial for VR/AR, teleoperation, and grasping applications. Given an image as input, the problem aims not only to estimate proper hand-object pose inside camera view but also to recover a natural grasp configuration. While estimating hand [31, 26, 49, 2, 16, 48] or object [17, 20, 12, 46, 47] alone has made a remarkable success in the past decades, jointly estimating hand-object pose [21, 43, 20, 24, 11] during interaction has only emerged in the past few years. Admittedly, jointly pose estimation is harder since the inevitable mutual occlusion. But in the meantime, hand-object interaction also imposes strong constraints on their relative configuration.

Previous works on joint hand-object estimation usually treat the contact as a consequence of the correct pose estimation [20, 25, 36]. None of them have fully explored the *contact* that dwells in manipulations. Since contact can provide rich cues to guide a natural plausible grasp, recently, more attention has been drawn on the contact modeling [3, 5] and contact representation [23, 4], and several contact datasets [3, 5, 42] have been publicly released. However, how to properly integrate the contact into the current hand-object modeling pipeline remains in the air.

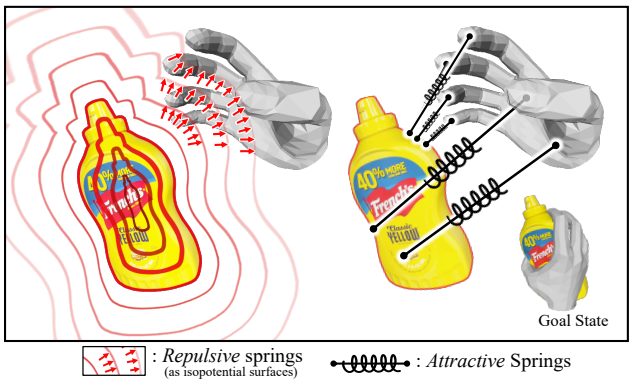

Figure 1. **Illustration of the proposed Contact Potential Filed.** The contacts between hand and object vertices are modeled as the attractive (right) and repulsive (left) springs that connected paired vertex on them.

Apparently, if the hand and object can be perfectly reconstructed, the contact between them will thereby be perfect. Yet, such perfection cannot be achieved in practice. Previous arts either propose a repulsive loss during training [21] to mitigate the penetration effect, which is an incomplete treatment, or refine the predicted grasp in a physics simulator [24, 25], which lacks flexibility. To systematically model the contact, we propose an explicit representation named **C**ontact **P**otential **F**ield (**CPF**). It is built upon the idea that contact between a hand and an object is a multi-point contact, which involves multiple hand-object vertex pair connections. Each connection can be modeled as a stretched or compressed spring to form a spring-mass system, and thus the whole contact is a field spanned by elastic potential energy, as shown in Fig.1. Therefore, estimating

a grasping pose is equivalent to minimize the elastic potential energy in that field. Such representation can by nature avoid interpenetration and manage to control the disjointedness. Based on CPF, we also propose a novel learning-fitting hybrid framework for **M**odeling the **I**nteraction scenario of **H**and and **O**bject, as we call **MIHO**. MIHO learns to firstly recover the contact potential field and then adopts it to model the hand-object pose under interaction.

Another problem with current methods is the representation of the hand model. Some of them adopt the parametric skinning models, such as MANO [39] to represent the hand. However recovering an articulated and deformable hand requires high DoF regression, which is prone to anatomical abnormality. Several other researchers in the robotics community adopt a rigid dexterous hand model and refine the joint pose in virtue of the off-the-shelf grasping software [30]. But the shape and appearance are not paid attention to when modeling the rigid hand, and the stiffness of those rod-like hands is less suitable for applications in CV/CG. To make the best of both worlds, we propose a novel representation named K-MANO. It inherits the formulation of MANO and constrains the hand's DoF within the local frame of *twist-splay-bend* (Fig.2) axis along the hand **K**inematic chains. With K-MANO, we can impose the rigidity on each joint within the proposed local frame, while still preserving the flexibility of the skinning model, Detailed discussion can be referred to §3.

For evaluation, we report our score on FHB [15] and HO3D [19, 18] dataset with reconstruction and physical quality metric. To note, the ground truth of FHB is noisy and suffers from severe interpenetration [23]. Since our method can avoid the penetration issue in the first place, our results are much more visually pleasant and physically plausible. Therefore, we argue that, in this dataset, a higher reconstruction score does not necessarily benchmark the performance of the method. While on HO3D, which is a cleaner dataset in terms of penetration and disjointedness, we achieve state-of-the-art performance on both reconstruction and physical quality.

Our contribution can be summarized as follows:

- We highlight *contact* in the hand-object interaction modeling task by proposing an explicit representation named CPF. With the inherited attraction and repulsion, it can by nature avoid interpenetration and disjointedness issues.
- We introduce K-MANO, a novel hand model that considers kinematic constraints along the proposed *twist-splay-bend* axis in the local frame, helping to fit a more natural hand pose inside CPF.
- We present a novel framework, MIHO, for hand-object interaction modeling. It can achieve state-of-the-art performance on HO3D, and also produce more plausible results on FHB.

2. Related work

Our method closely relates to the 3D hand reconstruction and 3D hand-object interaction modeling.

3D Hand Reconstruction. Recently, an emerging trend in hand mesh recovery has arisen. Most of current methods [2, 49, 1] adopted a deformable skinning hand model, MANO [39] as the template. To derive an articulated hand, obtaining joint rotation along hand kinematic chains is indispensable. Boukhayma *et al*. [2] proposed to directly regress the PCA components of rotations. However, regression on the lossy compressed PCA rotations is not preferable for hand recovery. Indeed, Zhou *et al*. [49] and Yang *et al*. [48] who adopted directly regressing the full joint rotation proved superior to those PCA-based methods. However, high DoF regression is prone to abnormality. Spurr *et al*. [41] proposed biomechanical constraints over hands in an end-to-end training scheme. Different from [41], we apply kinematic constraints over the proposed coordinates of *twist-splay-bend*, and integrate it into our optimization.

Hand-object Interaction Modeling. In the wild range of modeling hand-object interaction, the most commonly referred topics fell into two categories: 1) joint hand-object pose estimation [21, 20, 11] and 2) grasping taxonomy classifications [36]. For the first category, early methods under hand-object interaction scenario focused on either hand [35, 37, 44] or object [45] pose alone. With one step further, by assuming object shape could influence the hand pose, several methods [13, 6, 7, 8] estimated hand in grasping pose with an object. Jointly estimating hand and object was first presented by Romero *et al*. [38] via searching in a large database for nearest neighbors. Most recently, Hasson *et al*. proposed a bisected pipeline [21] that recover hand-object meshes in manipulation scenario, and its featured version [20] to exploit photometric consistency over video sequence. In the meantime, Doosti *et al*. [11] exploited the graph neural networks [14] to lift the 2D keypoints of hand and object into their pose in 3D space. And Tekin *et al*. [43] adopted 3D YOLO [34] to predict HO pose in one stage. These methods demonstrate the practicability of integrating the current elaborated neural networks into the hand-object pose estimation pipeline. Yet, the contact of the hand-object is still not fully addressed. Hasson *et al*. [21] treated contact as a consequence of pose prediction, and applied penalty on the intersection or penetration. Different solutions were proposed by Kokic *et al*. [24, 25], in which the proper contact was guaranteed by GraspIt [30] software. Rogez *et al*. [36] directly classified the contact-force pattern based on grasping taxonomy [28]. While our method falls in the category of hand-object pose estimation, we explicitly model the contact in form of a spring-mass system and a novel solution upon it.

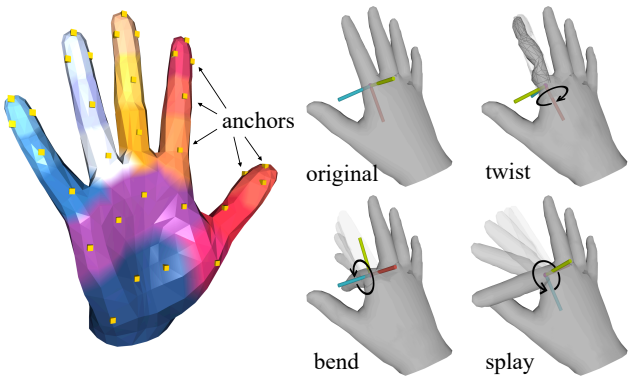

Figure 2. **Illustration of the proposed K-MANO.** Left: the subdivision of hand regions and anchors attached to it. Right: the local frame of *twist-splay-bend* axis along with hand kinematic chains.

3. Kinematic-chained MANO, K-MANO

Our kinematic hand model K-MANO inherits from a parametric skinning model, MANO [39]. Similarly, K-MANO drives an articulated hand mesh with pose parameters θ and shape parameters β. θ is 16 joint rotations (15 local, 1 global frame, axis-angle form) along with the hand kinematic chains. And β represents the 10 shape PCA components of shape learned from scans. The key differences between K-MANO and MANO are: 1) the restriction on the *twist-splay-bend* local frame with respect to human anatomy; 2) the anchor point representation on phalanges region.

Local Frame of *Twist-Splay-Bend*. Apparently, fitting on 15 local rotations (45 DoFs) of K-MANO inserts high nonlinearity therefore may cause monstrous hand posture (Fig. 7). Since the human hand is by nature a kinematic chain, and the majority of the articulations only have DoF along the bending axis, we can impose constraints over the abnormal pose. More specifically, for each articulation along kinematic chain, we assign a Cartesian coordinates system whose x, y, z axis is along the 3 revolute directions: *twist*, *splay*, and *bend* that mimic Euler angle representation, as shown in Fig.2. Then we can impose kinematic constraints along the *twist* and *splay* axis in local frame. Details of assigning *twist-splay-bend* axis are elaborated in Appx. A.1.

Anchor Points. Since the hand mesh of different subjects are almost identical in the local region (*e.g.* metacarpals and phalanges), we can down-sample the vertices on hand mesh largely relieving the computation burden. Instead of attaching springs from objects surface to all the vertices on hand mesh, we attach them on several sub-region centers, called *anchor* point (Fig.2). Based on the statistics of frequently contact hand part reported in [21, 5], as well as the human anatomy, we first divide the full hand palm into 17 regions: 3 for each phalange of 5 fingers, 1 for metacarpals, and another for carpals. Then, we assign up-to 4 anchors for each region. Details of hand region and anchors are described in Appx. A.2, A.3.

4. Contact Potential Field

We regard the contact between hand and object as a multi-point contact [4], which can be directly described by the vertex representation. Natural grasping configuration [9, 23] implies that the hand and object should have more regions that are in stable contact with each other, and have less volume that interpenetrates each other. These can be modeled as the adversarial interaction of attraction and repulsion on the surface of hand and object. In light of this, [21] designed a repulsive loss based on ray-casting to supervise CNN on alleviating collision, [4] adopted a predefined object contact-map to fit hand in proper configuration. In this paper, we formulate each vertex contact pair as a spring-mass system, and the whole system as an elastic potential field (as we call CPF). Therefore, attraction and repulsion has a simple yet effective representation: stretched and compressed springs.

Contact as Spring-Mass System. A single contact is a spring-mass system consists of a spring and two mass points on each side (hand and object). We set the mass for each point as unit mass. When the spring is at its rest position, it exhibits no force, but when it is stretched or compressed, according to Hooke's Law[1], it will exhibit a force of $-k \cdot \boldsymbol{\Delta l}$, where k is the spring elasticity, $\boldsymbol{\Delta l}$ is the change in length with respect to its direction.

Attractive Spring. We attach each vertex $\mathcal{V}_j^o$ on object surface with a spring to the closest vertex $\mathcal{V}_i^h$ on hand surface to construct the spring-mass system. Thus, $\boldsymbol{\Delta l}_{ij} = \mathcal{V}_i^h - \mathcal{V}_j^o$. Supposing the rest position equals 0, then all those stretched springs will attract the disjoint hand-object and connect the vertex pair to touch. Its force and potential energy can be formulated as: (*atr* as "attraction")

$$F_{ij}^{\text{atr}} = -k_{ij}^{\text{atr}} * \boldsymbol{\Delta l}_{ij}^{\text{atr}}; \quad E_{ij}^{\text{atr}} = \frac{1}{2} k_{ij}^{\text{atr}} * |\boldsymbol{\Delta l}_{ij}^{\text{atr}}|^2 \quad (1)$$

Repulsive Spring. Solely depending on the attractive springs cannot effectively avoid interpenetration. *e.g.* The hand may penetrate the object surface but still with the contact vertices in touch. To overcome this, we also define a compressed spring. Supposing the compressed spring has rest length at $+\infty$, its force and potential energy are: (*rpl* as "repulsion")

$$F_{ij}^{\text{rpl}} = -k_{ij}^{\text{rpl}} * e^{-\boldsymbol{\Delta l}_{ij}^{\text{rpl}}}; \; E_{ij}^{\text{rpl}} = \frac{1}{2} k_{ij}^{\text{rpl}} * |e^{-\boldsymbol{\Delta l}_{ij}^{\text{rpl}}}|^2 \quad (2)$$

where $\boldsymbol{\Delta l}_{ij}^{\text{rpl}} = (\mathcal{V}_i^h - \mathcal{V}_j^o) \cdot \mathbf{n}_j^o$ is the projection of the distance vector on object normal $\mathbf{n}_j^o$. As illustrated in Fig.1

[1] https://en.wikipedia.org/wiki/Hookes_law

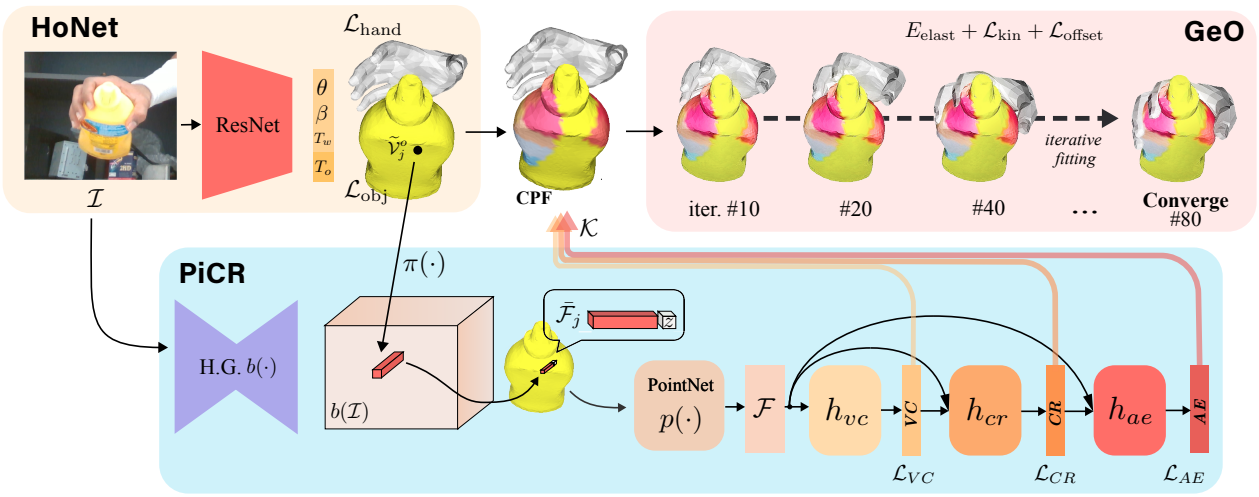

Figure 3. **The architecture of the hybrid model MIHO.** The MIHO consists of three submodules: the first HONet estimates a coarse hand-object pose, the second PiCR learns to recover the CPF and the last GeO retrieves the refined pose based on the CPF.

(left), the energy stored in compressed springs forms exponentially decaying isopotential surfaces. When the hand is penetrating or in the vicinity of the object, a large repulsive force will be applied on each side. Detailed analysis of the attraction-repulsion equilibrium can be found in Appx. B.1.

Grasping inside Contact Potential Field. By collecting all the attractive and repulsive energy, to form a natural grasping is equivalent to minimize the elastic energy:

$$E_{\text{elast}} = \sum_{i}^{N_H} \sum_{j}^{N_O} (E_{ij}^{\text{atr}} + E_{ij}^{\text{rpl}}) \tag{3}$$

As discussed in §3, the hand vertices can be simplified to region-based *anchors*, which will largely relax the difficulty of learning and fitting inside elastic potential field. We replace the $\boldsymbol{\Delta l}_{ij}$ to $\boldsymbol{\Delta l}'_{ij} = \mathcal{A}_i^h - \mathcal{V}_j^o$ in Eq.1, where $\mathcal{A}_i^h$ is the closest anchor on the hand surface to $\mathcal{V}_j^o$.

While the attractive force is related to specific hand-object pairs, the repulsive force is rather ambient. Thus to integrate the CPF into a learning-based framework, we define the elasticity of attractive spring $k_{ij}^{\text{atr}} \in \mathcal{K}$ as learnable predictions of neural network. Since we hope the attractive force asserted to each spring is rather balanced, a k_{ij}^{atr} that is inverse-proportional to the $|\boldsymbol{\Delta l}_{ij}|$ is adopted to describe the vertex pairs. Later we treat this guidance k as the pseudo-ground-truth k_{ij}^{GT} and supervise our learning framework based on it. As for the elasticity of the repulsive springs, we empirically set all the k_{ij}^{rpl} to 1×10^{-3}. Detailed analysis of generating k^{GT} is provided in Appx. B.2.

5. Our Approach – MIHO

In virtue of the proposed CPF (§4), our approach MIHO reconstructs the hand-object interaction in three stages, namely HoNet (§5.1), PiCR (§5.2), and GeO (§5.3). The output of MIHO not only guarantees a natural grasping configuration but is also close to the ground-truth.

As shown in Fig.3, given an RGB image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$, HoNet predicts a coarse hand mesh $\widetilde{\mathcal{V}}^h \in \mathbb{R}^{N_H \times 3}$ and object mesh $\widetilde{\mathcal{V}}^o \in \mathbb{R}^{N_O \times 3}$, where N_H and N_O is the number of the vertex of the hand and object respectively. Then, PiCR learns to construct the CPF which consists of $N_H \times N_O$ spring-mass systems $\mathcal{K}(\widetilde{\mathcal{V}}^h, \widetilde{\mathcal{V}}^o) \in \mathbb{R}^{N_H \times N_O}$ that connects paired vertices on hand and object. For each index (i, j) in $\mathcal{K}$ represents the elasticity k_{ij}^{atr} of the spring that connect $\widetilde{\mathcal{V}}_i^h$ to $\widetilde{\mathcal{V}}_j^o$. Finally, GeO minimizes the elastic potential energy E_{elast} in that field to yield refined hand and object mesh $\hat{\mathcal{V}}^o$, $\hat{\mathcal{V}}^h$.

5.1. Hand-object Pose Estimation Network, HoNet

By exploiting the CPF, we should first have a coarse hand and object mesh model. It is achieved by the adopted HoNet. Following the implementation of [20], for the object part, we assume the object's 3D mesh model is given as a prior. Thus estimating the vertices of the object is equal to regressing its 6D pose $\in \mathbb{R}^6$. For the hand part, we obtain full hand mesh by regressing the MANO $\theta \in \mathbb{R}^{15}$, $\beta \in \mathbb{R}^{10}$, and a global 6D pose $T \in \mathbb{R}^6$ of the wrist joint. In all, the HoNet regresses total of 37 coefficients to recover absolute hand and object mesh in the camera system. Details of HoNet can be referred to [20].

5.2. Pixel-wise Contact Recovery Module, PiCR

Constructing the CPF needs to firstly construct the per-vertex hand and object connection and then recover the spring elasticity that describes its connection. This is achieved by the proposed Pixel-wise Contact Recovery

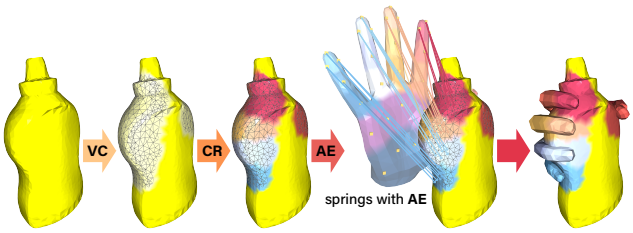

Figure 4. The process of assigning vertex contact, contact region and anchor elasticity onto object surface.

(PiCR) module. PiCR adopts three cascaded predictions: 1) *Vertex contact* (VC) implies which vertices on object are in contact with hand; 2) *Contact region* (CR) implies which hand region that the vertices in VC are connected to; 3) *Anchor elasticity* (AE) represents the elasticity of the connecting spring. With VC, CR, and AE obtained, we can then recover the CPF as illustrated in Fig.4.

The proposed PiCR consists of a backbone b that extracts features from the image and three heads h_{vc}, h_{cr} and h_{ae} which convert the per-vertex features into VC, CR, and AE, respectively. For the backbone part, we adopt the Stacked Hourglass Networks [32], and for the heads part we use the PointNet [33] architecture. As illustrated in Fig.3, the process of PiCR encoder can be expressed as:

$$\bar{\mathcal{F}} = [f(\pi(\widetilde{\mathcal{V}}^o), b(\mathcal{I})), z(\widetilde{\mathcal{V}}^o)]; \quad \mathcal{F} = p(\bar{\mathcal{F}}) \tag{4}$$

where $\pi(\cdot)$ is perspective camera projection, $f(\cdot)$ is the PIFu [40] function of aligning 3D vertices $\widetilde{\mathcal{V}}^o$ to its pixel-wise features $\bar{\mathcal{F}}$, and $p(\cdot)$ is the PointNet encoder that converts $\bar{\mathcal{F}}$ into its point-wise features $\mathcal{F}$. Since $\pi(\cdot)$ will cause the geometry lost along the z axis, we also concatenate the object's root relative z value $z(\widetilde{\mathcal{V}}^o)$ at the end of $\bar{\mathcal{F}}$. The process of PiCR heads are:

$$VC = h_{vc}(\mathcal{F});\ CR = h_{cr}(VC, \mathcal{F});\ AE = h_{ae}(CR, \mathcal{F}) \tag{5}$$

Vertex Contact. The first prediction in PiCR is VC $\in \mathbb{R}^{N_O}$. Each index j in VC is a probability of the object vertex is in contact with the hand. Therefore, the loss of VC prediction is naturally set as binary focal loss [27]:

$$\mathcal{L}_{VC} = -\sum_{j}^{N_O} \mathbb{1}_j^{img} * \alpha_j (1 - f_j)^{\gamma} \log(f_j) \tag{6}$$

where $f_j = p_j$ if ground-truth is in contact, otherwise $f_j = (1 - p_j)$, and the p_j is the predicted contact likelihood. $\mathbb{1}_j^{img}$ denotes if the vertex is projected inside the image plane. α_j is inverse class frequency and γ is empirically set to be 2.

Contact Region. The second prediction CR $\in \mathbb{R}^{N_O \times 17}$ is a multi-class classification that indicates the probabilities of total of 17 hand regions that an object vertex is connected to. The loss function $\mathcal{L}_{CR}$ is defined as a multi-class focal loss.

Algorithm 1: Procedure of recovering CPF

Input: $\widetilde{\mathcal{V}}^o, \widetilde{\mathcal{V}}^h, VC, CR, AE$
Output: E: elastic potential energy

recovery anchors: $\widetilde{\mathcal{A}} \leftarrow linear\_interp.(\widetilde{\mathcal{V}}^h)$;
for *j in range of N_O* **do**
  if $VC[j] > t_{vc}$ **then**
    recover region: $\mathcal{R} \leftarrow argmax(CR[j])$;
    for *i in range of (N_{anchor} in $\mathcal{R}$)* **do**
      recover elasticity: $k_{ij}^{\text{atr}} \leftarrow AE[j]$;
      $E+ \leftarrow \frac{1}{2} k_{ij}^{\text{atr}} * |\widetilde{\mathcal{A}}_n - \widetilde{\mathcal{V}}_j^o|^2$;
    for *i in range of N_H* **do**
      if $|\widetilde{\mathcal{V}}_i^h - \widetilde{\mathcal{V}}_j^o)|^2 < t_{dist}$ **then**
        assign elasticity: $k_{ij}^{\text{rpl}} \leftarrow 1 \times 10^{-3}$;
        $E+ \leftarrow \frac{1}{2} k_{ij}^{\text{rpl}} |\exp(-(\widetilde{\mathcal{V}}_i^h - \widetilde{\mathcal{V}}_j^o) \cdot \mathbf{n}_j^o)|^2$;

$$\mathcal{L}_{CR} = -\sum_{j}^{N_O} \mathbb{1}_j^{VC} * \mathbb{1}_j^{img} * (1 - m_j)^{\gamma} * \log(m_j) \tag{7}$$

where the $m_j = \sum(p_j * t_j)$ in which $p_j \in \mathbb{R}^{17}$ is the predicted per-region probabilities with *softmax*, and $t_j \in \mathbb{R}^{17}$ is the ground-truth one-hot label. $\mathbb{1}_j^{VC}$ denotes that the ground-truth VC of vertex j is positive.

Anchor Elasticity. The last prediction in PiCR is AE $\in \mathbb{R}^{N_O}$. each index j in AE stands for the elasticity k_{ij} for the given i-th anchor $\mathcal{A}_i^h$. Since the value of k_{ij} falls into range $0 \sim 1$, the magnitude of it can be viewed as the confidence of how close $\mathcal{V}_j^o$ is in the vicinity of $\mathcal{A}_i^h$. Thus we treat AE as a classification problem. The loss function $\mathcal{L}_{AE}$ is a binary cross-entropy (BCE):

$$\mathcal{L}_{AE} = \sum_{j}^{N_O} \mathbb{1}_j^{VC} * \mathbb{1}_j^{img} * \text{BCE}(\, k_{ij},\, k_{ij}^{\text{GT}}) \tag{8}$$

where k_{ij} is the predicted anchor elasticity and $k_{i,j}^{\text{GT}}$ is the guidance.

Given the VC, CR and AE that predicted in PiCR, as well as the $\widetilde{\mathcal{V}}^o, \widetilde{\mathcal{V}}^h$ in HONet, PiCR finally recovers the CPF with the elastic energy E_{elast} as described in Algm.1. We empirically set the probability threshold of VC: t_{vc} and the distance threshold: t_{dist} to 0.8 and $20mm$, respectively.

5.3. Grasping Energy Optimizer, GeO

Due to the noisy regression from HoNet, we cannot guarantee a natural grasping configuration. To remedy this, we propose a Grasping Energy Optimizer (GeO) to refine the hand-object pose in virtue of the predicted CPF.

For the object part, since we use a given object model as prior, adjusting the object's vertices is equal to adjust the 4×4 transformation $T_o \in \mathbb{SE}(3)$ from object canonical

system to the camera system. For the hand part, we jointly adjust the K-MANO' s 15 local rotations R_j and one wrist transformations T_w.

In order to abbreviate the monstrous hand posture during optimization, we define a kinematic loss $\mathcal{L}_{\text{kin}}$ that penalize the abnormality on 15 local *axis-angle* based on the *twist-splay-bend* coordinates of K-MANO. First, for joints along hand kinematic chains, we penalize the components of *axis* $\mathbf{a}^{rot}$ on *twist* direction $\mathbf{n}^{twist}$, since any rotation that cause the finger twisting along its pointing direction is prohibited. Second, for joints that not belongs to 5 knuckles, we also penalize the component of $\mathbf{a}^{rot}$ on *splay* direction $\mathbf{n}^{splay}$. Last, we penalize the *angle* ϕ that revolutes along the *bend* axis if it is greater than $\pi/2$. The total kinematic loss on hand can be written as:

$$\begin{aligned}\mathcal{L}_{\text{kin}} = &\sum_{j\in\text{all}} \mathbf{a}_j^{rot}\cdot\mathbf{n}_j^{twist} + \sum_{j\notin knuck.} \mathbf{a}_j^{rot}\cdot\mathbf{n}_j^{splay} \\ &+ \sum_{j\in all}(\max(\phi_j^{bend} - \frac{\pi}{2}), 0)\end{aligned} \tag{9}$$

We also penalize the offset of the refined hand-object vertices ($\hat{\mathcal{V}}^o$, $\hat{\mathcal{V}}^h$) from its initial estimations ($\widetilde{\mathcal{V}}^o$, $\widetilde{\mathcal{V}}^h$) in form of L2 distance: $\mathcal{L}_{\text{offset}}$. The whole minimization can be expressed as:

$$\hat{\mathcal{V}}^o, \hat{\mathcal{V}}^h \longleftarrow \underset{T_o,\,(T_w,R_j)}{\arg\min}\ (E_{\text{elast}} + \mathcal{L}_{\text{kin}} + \mathcal{L}_{\text{offset}}) \tag{10}$$

6. Experiments and Results

In this section, we first describe datasets (§6.1) and evaluation metrics (§6.2) in our experiments. Then, we provide our implementation details (§6.3). Finally, we compare our results with the previous arts and present an ablation study to show the efficacy of MIHO (§6.5).

6.1. Datasets

There are mainly four public available datasets, namely ObMan [21], FHB [15] and HO3D [18, 19] and ContactPose [5] that contain images and ground-truth 3D meshes of hand-object interaction. However, since ObMan is purely synthetic, while ContactPose only involves textureless objects, these two datasets are less suitable for evaluating the performance of MIHO. Thus, we only evaluate our framework on FHB and HO3D.

First-person hand action benchmark, FHB. FHB collects first-person RGBD video of hand in manipulation with objects. The ground-truth of hand poses captured via magnetic sensors. In our experiments, we use a subset of FHB that contains 4 objects with a scanned model and pose annotation. We only use the *action* split following the protocol of [20, 43], and filter out the samples with a minimum hand-object distance greater than 5 mm, which yields us 7223 samples for training and 7373 for testing.

HO3D. HO3D is another dataset that contains precise hand-object pose during the interaction. Due to historical reasons, there is two version of HO3D, namely v1 [18] and v2 [19]. In our experiments, we mainly compare our methods with the baseline [20] on HO3Dv1, but also conduct several comparisons with the recently released pre-trained model of [20] on HO3Dv2. Similar to FHB, we filter out samples with distance threshold $5mm$. It's also worth mentioning that, since our method requires a known object model, as well as a stable grasping configuration, nearly 5448 samples in HO3Dv2 test set are not suitable for our methods to report. Therefore, we manually select 6076 samples in HO3dv2 test set to compare MIHO with [20]. We call this split by HO3Dv2$^-$. Besides, training HO3Dv1 in previous methods [18, 20] requires an extra synthetic dataset that is not publicly available. Following the augmentation procedure described in [18], we manually extend the HO3Dv1 train set (referred as HO3Dv1$^+$) and reproduce the results (referred as [20]$^+$) comparable with those in [20]. Details on the HO3Dv1/v2 analysis and selection, as well as augmentation details are provided in Appx. C.

6.2. Metrics

Since hand-object interaction estimation requires not only the proper pose of both hand and object but also the natural grasping configuration, we totally report 5 metrics that cover both reconstruction and physical quality.

Mean 3D errors. We compute the mean vertex distance for hand and object to assess the quality of pose estimation.

Penetration depth. To measure how deep the hand is penetrating the object's surface, we calculate the penetration depth which is the maximum distance of all the penetrated hand vertices to their closet object surface.

Solid intersection volume.To measure how much space intersection that occurs during estimation, we voxelize the object mesh into 80^3 voxels, and calculate the sum of the voxel volume inside the hand surface.

Disjointedness errors. We also encourage stable hand-object contact, which can be depicted as attracting fingertips (5 most frequently contact region) onto or in the vicinity of the object surface. Therefore, we define the disjointedness metrics as the average over the average distance of hand vertices in 5 fingertips region to their closet object surface.

Abnormality scores. As referred in §5.3, it is abnormal to observe finger's rotation along *twist* and *splay* axis. Thus we report the average of rotation axis's component on *twist* and *splay* direction as the hand's abnormality score.

6.3. Implementation Details

We implement our framework on PyTorch with Adam solver. As for HoNet, we use the pre-trained model from [20] on FHB and HO3Dv2, and reproduce a comparable

Datasets	FHB					HO3Dv1$^+$				HO3Dv2$^-$	
Method	Ours†	Ours‡	GT	[20]	ObMan*	Ours†	Ours‡	GT	[20]$^+$	Ours‡	[20]
Hand vertex error $(mm)\downarrow$	21.16	19.54	-	17.51	18.42	24.61	24.05	-	24.80	-	-
Object vertex error $(mm)\downarrow$	-	21.57	-	21.06	21.17	-	19.60	-	18.10	73.28 $^\diamond$	75.77 $^\diamond$
Penetration depth $(mm)\downarrow$	**16.13**	**16.92**	19.55	20.63	19.76	10.58	8.25	7.55	18.57	**16.47**	20.02
Solid intersec. volume $(cm^3)\downarrow$	**12.56**	**11.76**	20.41	21.10	16.16	**2.72**	**1.47**	3.57	9.62	**7.44**	9.25
Disjointedness error $(mm)\downarrow$	**24.54**	**22.41**	37.28	37.40	27.95	**14.46**	22.38	14.53	18.62	**37.04**	41.41
Abnormality scores $\downarrow$	0.2348	0.2546	0.2924	0.2381	-	0.1679	0.1699	0.2001	0.1751	0.2277	0.2213

Table 1. **Quantitative results and detailed comparison with previous state-of-the-art [20, 21] on FHB and HO3D datasets.** "†" denotes ours *hand-alone* optimization setting and "‡" denotes the jointly *hand-object* setting. "$*$" denotes the reproduced ObMan [21]. "$\diamondsuit$" denotes the *wrist-relative* object vertex error. "-" indicates the results that are not available.

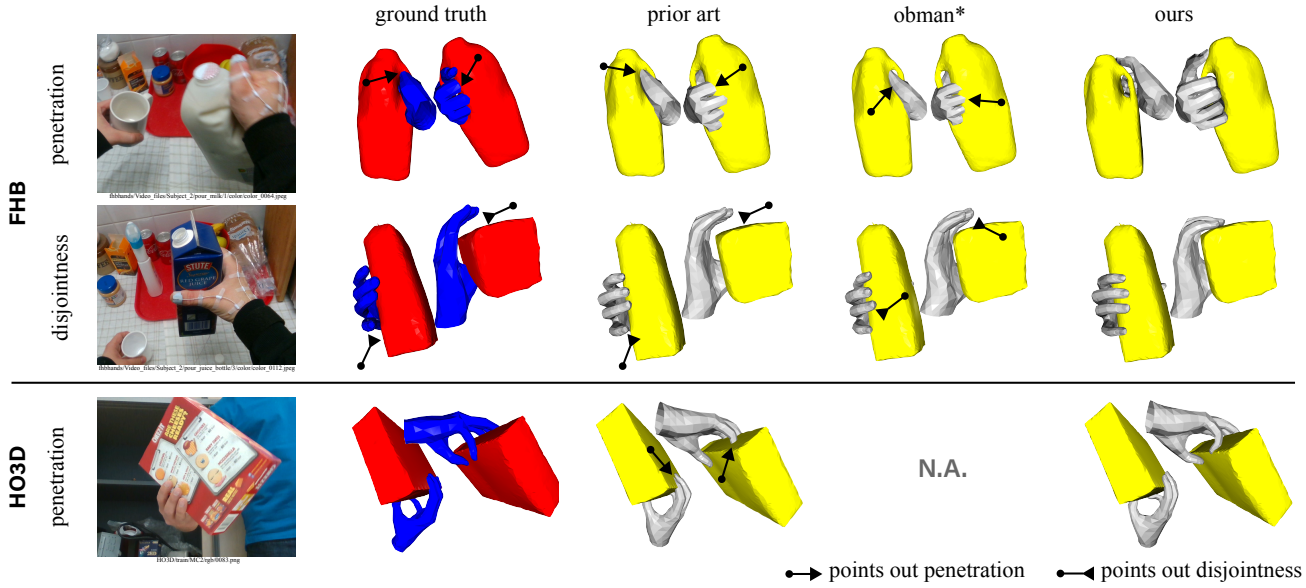

Figure 5. **Qualitative Comparison with ground-truth and previous arts on FHB and HO3D dataset.**

model on HO3Dv1. As for PiCR, we adopt hourglass networks with 2 stacks as the backbone. PiCR's heads are attached to the end of each stack. While training losses are computed over the sum of all stacks, prediction values are only extracted from the last one. We also implement our fitting module GeO in PyTorch thanks for its auto derivative, but GeO can also support other optimization toolboxes with minor effort. Training details are presented in Appx. D.1.

6.4. Comparison with Previous Arts

For the FHB dataset, we compare our methods with the previous art [20, 21] of hand-object reconstruction. [20] reported results on two experiments setting: 1) fully supervised with single-frame, and 2) sparsely supervised with photometric consistency. We only selected those results from setting 1). Since [20] didn't exploit repulsive and attractive loss during training, direct comparison on intersection and disjointedness may not be convincing enough. While the contact losses were considered in another work named ObMan [21], it only represented the genus 0 object mesh as a deformed icosphere, which is also not directly comparable with ours (known object model). To ensure rational comparison, we migrate the *repulsion loss* and *attraction loss* of ObMan to the MeshRegNet in [20], and reproduce the results on par with it. We call this adaptation: ObMan*. For the HO3Dv1 dataset, we compare our results with the reproduced [20]$^+$ that is trained on the aforementioned HO3Dv1$^+$.

We report our results under two experimental settings: 1) ***hand-alone*** that fixes the object at the predicted pose in HoNet, and only optimizes the hand pose in GeO; 2) ***hand-object*** that jointly optimizes the hand and object poses in GeO. In Tab.1 we show our results as well as the comparison with the previous arts in all 5 metrics. For FHB dataset, as analyzed in [5], its ground-truth suffers from frequent interpenetration. We find that lower vertex error does not necessarily benchmark a higher reconstruction quality. Indeed, as shown in Tab.1 (col. 4, 5), either ground-truth or [20] reveals substantial solid intersection volume, penetration depth or disjointedness. We find that with the proposed

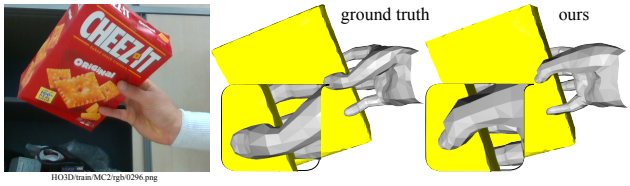

Figure 6. Example to show that our method can by nature mitigate the unexpected twist (see thumb) exhibited in ground-truth.

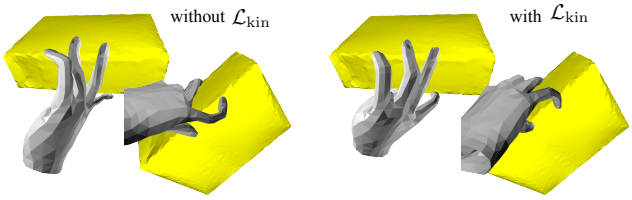

Figure 7. Example to illustrates that our proposed K-MANO with kinematic constraints ($\mathcal{L}_{\text{kin}}$) can effectively avoid monstrous pose during fitting.

CPF, MIHO can outperform [20] by a margin of 3.71 mm in penetration depth, 9.34 cm^3 in solid intersection volume, and 14.99 mm in disjointedness errors, while only suffers from minor performance cost in mean hand vertex errors 2.03 mm and mean object vertex errors of 0.51 mm. These are consistent with our expectation that the CPF can by nature repulse intersection away and attract disjointedness to touch. As for HO3Dv1, our method also outperformed previous art over the reconstruction metric and physical quality. Visual comparison are shown in Fig.5. Besides, we also observe an unexpected twist of thumb in HO3Dv1 test set. We show in Fig.6 that, since K-MANO inserts restriction on the $twist$ components of the rotation axis, our method can achieve a more visually pleasing result in such case.

As for HO3Dv2, since we only test MIHO on a subset of HO3Dv2, as we call HO3Dv2$^-$, our results are not directly suitable for submitting on HO3Dv2 CodaLab Challenge [2] for hand. Thus, we only report the object 3D vertex errors on HO3Dv2$^-$. We firstly align the predicted object vertex to the predicted hand wrist joint, then compute the *wrist-relative* object vertex error with those in ground-truth. Detailed comparison in Tab.1 (col.11, 12) shows that MIHO achieves better performance than the previous art.

6.5. Ablation Study

In this experiment, we further evaluate the effectiveness of the proposed CPF and K-MANO in MIHO. The ablation studies are mainly conducted on the FHB test set with *action* split. The main text includes 4 representative experiments (3 quantitative evaluations and 1 qualitative demonstration) on the repulsive springs and kinematic constraints. For more studies on 1) the impact of the magnitude of k^{rpl}; 2) another version of K-MANO with PCA pose; 3) the comparison between fitting w.r.t E_{elast} and fitting w.r.t ObMan contact loss; please visit Appx. D.2.

[2] https://competitions.codalab.org/competitions/22485

	settings		scores			
	E^{rpl}	$\mathcal{L}_{\text{kin}}$	PD↓	SIV↓	D↓	AS↓
(a)	✓	✓	16.92	11.76	22.41	0.2546
(b)	✗	✓	17.79	13.76	20.27	0.2546
(c)	✓	✗	15.38	12.07	22.55	0.2719

Table 2. **Ablative study on repulsive springs and K-MANO kinematic constraints.** PD stands for penetration depth (*mm*); SIV stands for solid intersection volume (cm^3); D stands for disjointness error (*mm*); AS stands for abnormality score.

Effectiveness of Repulsive Springs. In order to measure the efficacy of the repulsive springs in CPF, we remove all the elastic energy E^{rpl} induced by them, leaving the attractions as the unique type of forces applied on hand and object. As we expected, the result in Tab.2(b) witnesses the accumulation of penetration depth and solid intersection volume. By removing the repulsive springs, we still witness a slight improvement of solid intersection volume over FHB ground-truth. These are attributed to the repulsive behavior of the attractive springs: when the hand vertex is inside the object surface, the attraction force will act as repulsion that pushing the hand out of penetration.

Effectiveness of the K-MANO kinematic constraints. We further highlight the efficacy of adopting the rotation constraints along *twist* and *splay* axis, as well as the angle constraints of *bend* direction. By removing the $\mathcal{L}_{\text{kin}}$ in Eq.9, we have witness a noticeable increase on abnormality scores in Tab.2(c), which indicates the monstrous pose along the prohibited direction. In order to further demonstrate the abnormal behavior, we also conduct a contrastive experiment whose only difference is the absence of $\mathcal{L}_{\text{kin}}$. Both experiments start from a zero (flat) hand and minimize the E_{elast} based on the same predicted CPF. We show in Fig.7 that the kinematic constraints are able to effectively prevent abnormality during the fitting process.

7. Conclusion

In this work, we propose a novel contact representation named CPF and a learning-fitting hybrid framework MIHO to help to model the interaction scenario of hand and object. We show in our experiments that our methods, while being able to recover precise hand-object pose, can also effectively 1) avoid interpenetration and control disjointedness, and 2) prevent abnormality in hand pose. Our formulation can also enable future works including grasping generation and grasping affordance analysis, which we intend to translate human manipulations into humanoid robotic actions.

References

[1] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In *CVPR*, 2019. 2

[2] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3d hand shape and pose from images in the wild. In *CVPR*, 2019. 1, 2

[3] Samarth Brahmbhatt, Cusuh Ham, Charles C Kemp, and James Hays. Contactdb: Analyzing and predicting grasp contact via thermal imaging. In *CVPR*, 2019. 1

[4] Samarth Brahmbhatt, Ankur Handa, James Hays, and Dieter Fox. Contactgrasp: Functional multi-finger grasp synthesis from contact. In *IROS*, 2019. 1, 3

[5] Samarth Brahmbhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. ContactPose: A dataset of grasps with object contact and hand pose. In *ECCV*, 2020. 1, 3, 6, 7

[6] Minjie Cai, Kris M Kitani, and Yoichi Sato. Understanding hand-object manipulation with grasp types and object attributes. In *RSS*, 2016. 2

[7] Minjie Cai, Kris M Kitani, and Yoichi Sato. An ego-vision system for hand grasp analysis. *IEEE Transactions on Human-Machine Systems*, 2017. 2

[8] Chiho Choi, Sang Ho Yoon, Chin-Ning Chen, and Karthik Ramani. Robust hand pose estimation during the interaction with an unknown object. In *ICCV*, 2017. 2

[9] Enric Corona, Albert Pumarola, Guillem Alenya, Francesc Moreno-Noguer, and Grégory Rogez. Ganhand: Predicting human grasp affordances in multi-object scenes. In *CVPR*, 2020. 3

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 13

[11] Bardia Doosti, Shujon Naha, Majid Mirbagheri, and David J Crandall. Hope-net: A graph-based model for hand-object pose estimation. In *CVPR*, 2020. 1, 2

[12] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, 2017. 1

[13] Thomas Feix, Ian M Bullock, and Aaron M Dollar. Analysis of human grasping behavior: Object characteristics and grasp type. *IEEE transactions on haptics*, 2014. 2

[14] Hongyang Gao and Shuiwang Ji. Graph u-nets. *ICLR*, 2019. 2

[15] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *CVPR*, 2018. 2, 6, 15, 16

[16] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *CVPR*, 2019. 1

[17] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *CVPR*, 2018. 1

[18] Shreyas Hampali, Markus Oberweger, Mahdi Rad, and Vincent Lepetit. Ho-3d: A multi-user, multi-object dataset for joint 3d hand-object pose estimation. *arXiv preprint arXiv:1907.01481*, 2019. 2, 6, 14, 16

[19] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *CVPR*, 2020. 2, 6, 15, 16

[20] 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. 1, 2, 4, 6, 7, 8, 13

[21] 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. 1, 2, 3, 6, 7, 13, 15

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 13

[23] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. *3DV*, 2020. 1, 2, 3

[24] Mia Kokic, Danica Kragic, and Jeannette Bohg. Learning to estimate pose and shape of hand-held objects from rgb images. *arXiv preprint arXiv:1903.03340*, 2019. 1, 2

[25] Mia Kokic, Danica Kragic, and Jeannette Bohg. Learning task-oriented grasping from human activity datasets. *RA-L*, 2020. 1, 2

[26] Dominik Kulon, Riza Alp Guler, Iasonas Kokkinos, Michael M Bronstein, and Stefanos Zafeiriou. Weakly-supervised mesh-convolutional hand reconstruction in the wild. In *CVPR*, 2020. 1

[27] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *ICCV*, 2017. 5

[28] Jia Liu, Fangxiaoyu Feng, Yuzuko C Nakamura, and Nancy S Pollard. A taxonomy of everyday grasps in action. In *IEEE-RAS HUMANOID*, 2014. 2

[29] Kevin M Lynch and Frank C Park. *Modern Robotics*. Cambridge University Press, 2017. 11

[30] Andrew T Miller and Peter K Allen. Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, 2004. 2

[31] Gyeongsik Moon, Takaaki Shiratori, and Kyoung Mu Lee. Deephandmesh: A weakly-supervised deep encoder-decoder framework for high-fidelity hand mesh modeling. In *ECCV*, 2020. 1

[32] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 5

[33] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 5

[34] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *CVPR*, 2017. 2

[35] Grégory Rogez, Maryam Khademi, JS Supančič III, Jose Maria Martinez Montiel, and Deva Ramanan. 3d hand pose detection in egocentric rgb-d images. In *ECCV*, 2014. 2

[36] Grégory Rogez, James S Supancic, and Deva Ramanan. Understanding everyday hands in action from rgb-d images. In *ICCV*, 2015. 1, 2

[37] Javier Romero, Hedvig Kjellström, and Danica Kragic. Monocular real-time 3d articulated hand pose estimation. In *IEEE-RAS HUMANOID*, 2009. 2

[38] Javier Romero, Hedvig Kjellström, and Danica Kragic. Hands in action: real-time 3d reconstruction of hands in interaction with objects. In *ICRA*, 2010. 2

[39] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *TOG*, 2017. 2, 3, 11

[40] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, 2019. 5

[41] Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Otmar Hilliges, and Jan Kautz. Weakly supervised 3d hand pose estimation via biomechanical constraints. In *ECCV*, 2020. 2

[42] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *ECCV*, 2020. 1

[43] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+o: Unified egocentric recognition of 3d hand-object poses and interactions. In *CVPR*, 2019. 1, 2, 6

[44] 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. 2

[45] Dimitrios Tzionas and Juergen Gall. 3d object reconstruction from hand-object interactions. In *ICCV*, 2015. 2

[46] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, 2018. 1

[47] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. Marrnet: 3d shape reconstruction via 2.5 d sketches. In *NeurIPS*, 2017. 1

[48] Lixin Yang, Jiasen Li, Wenqiang Xu, Yiqun Diao, and Cewu Lu. Bihand: Recovering hand mesh with multi-stage bisected hourglass networks. In *BMVC*, 2020. 1, 2

[49] Yuxiao Zhou, Marc Habermann, Weipeng Xu, Ikhsanul Habibie, Christian Theobalt, and Feng Xu. Monocular real-time hand shape and motion capture using multi-modal data. In *CVPR*, 2020. 1, 2

Appendix

A. K-MANO

A.1. Local Frame Derivation

In this section, we introduce the proposed *twist-splay-bend* frame of K-MANO. Since both the original MANO [39] and our K-MANO hand model are driven by the relative rotation at each articulation, we can intuitively apply the constraints on the rotation *axis* and *angle* [3]. We intend to decompose the rotation *axis* into 3 components along 3 orthogonal axes that each component depict the proportion of rotation along that axis. However, there have infinity choices of the 3 orthogonal axes. MANO adopt all identical frames whose 3 orthogonal axes are not strictly coaxial to the direction of the kinematic chain (Fig.8 left). Different from them, we borrow the idea in Universal Robot Description Format (URDF) [29] (Chapter.4.2) to describe the articulations along with hand kinematic as the revolute joint[4]. Nevertheless, the common revolute joint in URDF only has a fixed DoF which is not enough to mimic the motion of the real hand. Thus, we extend the URDF model to have 3 revolute joints, named as *twist*, *splay* and *bend*, at each articulation (Fig.8 right),

Next, we elaborate the conversion from the coordinate system of MANO's all identicals to our frame with *twist-splay-bend* axis. For each articulation (or joint in URDF), we firstly retrieve a rough *splay* axis that is directly equal to MANO's y (up) axis. Then we compute the *twist* axis as the vector from the child of the current joint to itself. With the *twist* and *splay* axis, we can derive the *bend* axis that is perpendicular to the *twist-splay* plane by cross-product. Finally, we obtain the refined *splay* axis by applying cross product again on the *bend-twist* axis. The whole process is illustrated in Fig.9.

A.2. Hand Region Assignment

As explained in §3 (Anchor Points.) in the main text, we divide the hand palm into 17 regions. Since MANO drives the hand by the kinematic-chained links that are very similar to the skeleton structure of hand anatomy, we can derive those hand regions based on the MANO's link.

According to hand anatomy, it consists of carpal bones, metacarpal bones, and phalanges, where phalanges can be further divided into three kinds: proximal phalanges, intermediate phalanges, and distal phalanges. The region's name and its abbreviation are defined in Fig.10. For clarity, we number the MANO's links from 1 to 20 as illustrated in Fig.11 (left). To assign the MANO vertices to its corresponding region, we need firstly assign the vertices to the link that lies inside the region. This is achieved by the *control points*. For link 0-3, 5-7, 9-11, 13-15, 17-20, we set one control point at the midpoint of the link's ends, while for link 4, 8, 12, and 16, we set two control points at the upper and lower third of the link's ends. For clarity, we also number the control points from 0 to 23 as shown in Fig.11 (middle). After a list of control, points are obtained, we label each hand vertex to one of these control points by querying which control point it has the least distance from. Finally, we merge the vertices that belong to control points 0, 5, 10, 15, and 20 to derive region Palm Metacarpal, and merge those vertices that belong to control points 4, 9, 14, 19 to derive region Carpal.

[3] Rotation cay be represented as rotating along an *axis* by an *angle*.

[4] https://en.wikipedia.org/wiki/Revolute_joint

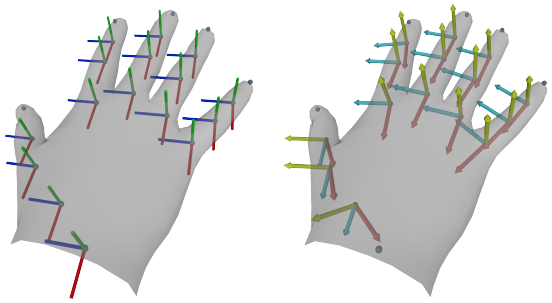

Figure 8. Visual comparison of MANO' s coordinate system to the proposed *twist-splay-bend* system.

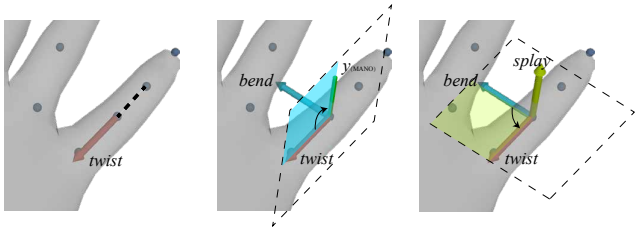

Figure 9. Illustration of converting MANO' s coordinates system to the proposed *twist-splay-bend* system.

A.3. Hand Anchor Selection

Here we elaborate on how we generate the *anchors* based on the regions and its control points. To ensure these anchors can be used in a common optimization framework and keep their representative power during the process of optimization, we propose the following protocols: **a)** hand anchor points should be located on the surface of the hand mesh. **b)** hand anchor points should distribute uniformly on the surface of the region it represents. **c)** hand anchor points can be derived from hand vertices in a differentiable way.

Hand anchor points are located on the surface of hand mesh (protocal **a**), so they must be located on some certain faces of the hand mesh. We can use the vertices of the face on which hand anchor points reside to represent hand anchor points. Suppose the hand mesh is $M = (V, F)$, where V is a list of all vertices and F is a list of all faces. Con-

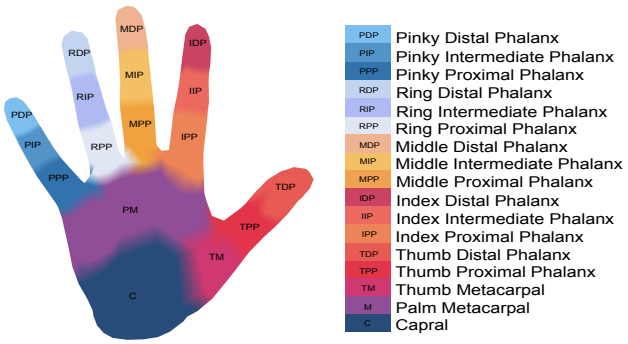

Figure 10. Hand region with name and abbreviation.

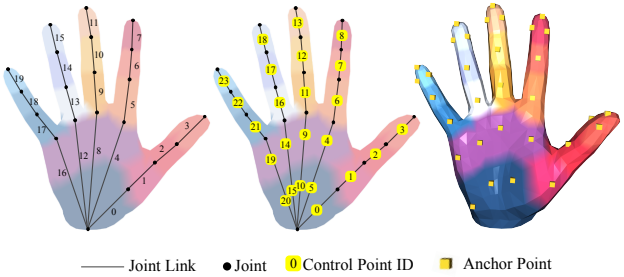

Figure 11. Left: joint links with ID; Middle: control points with ID. Right: anchor points

sidering one face $f \in F$ of mesh whose vertices are stored in order: $f = \{i_k\}, v_k = V[i_k], k \in \{1, 2, 3\}$. We can get two edges of that face: $e_1 = v_2 - v_1, e_2 = v_3 - v_1$. Then the local position of anchor a inside the face can be represented by linear combination of e_1 and e_2: $a = x_1 e_1 + x_2 e_2$, where the x_1, x_2 are weights. Finally, the global position of hand anchor points will be $v_1 + a = v_1 + x_1 e_1 + x_2 e_2 = (1 - x_1 - x_2)v_1 + x_1 v_2 + x_2 v_3$. During the optimization process, we can use the precomputed face f and weights x_1, x_2, along with the predicted hand vertices V to calculate the position of anchor. As the anchor is a linear combination of hand vertices, any loss that is applied to the anchors' position can be backpropagated to the vertices on the MANO surface, making the K-MANO differentiable.

We utilize control points introduced in §A.1 to derive anchors. Since the anchor selection is independent of hand configuration, we adopt a flat hand in the canonical coordinate system. As illustrated in Fig.11 (middle, right), the control points are roughly uniformly distributed in each region. Thus each control point will correspond to an anchor of that region. The Carpal region is an only exception: We select only 3 over 5 (ID: 5, 10, 20) of the control points in the Carpal region for anchor derivation. To derive an anchor from the control point, we need to get one face (consist of 3 integers) and two weights. For non-tip regions, we cast a ray in palm direction from each control point that the region contains and retrieve the first intersection with hand mesh. The intersection point will be the anchor correspond to that control point. For tip regions, we would select three anchors of each control point to increase the density of anchors in that region, as tip involves more contact information during manipulation. For the control point in tip regions, we first cast a ray originated from the control point and get the intersection point on the hand mesh. Then a cone is created with the control point as apex, the intersection point as the base center and the base radius is estimated by the maximum distance from points in the region to the control point. Three generatrices equally distributed on cone surface are selected as new ray casting directions. We cast three rays from the control point in the direction of the three generatrices and retrieve the intersection points with hand mesh. These intersection points will be selected as anchors to that control point in the fingertip region.

B. Elaborations of Spring's Elasticity

B.1. Elastic Energy Analysis

Here we illustrate elastic energy between a pair of points $\mathcal{V}_i^h$ and $\mathcal{V}_j^o$, denoting one vertex on hand surface and another vertex on object surface respectively. The vertex on object surface binds with a directional vector $\boldsymbol{n}_j^o$ representing the direction of repulsive force. Then we compute the offset vector $\boldsymbol{\Delta l}_{ij}^{\text{atr}} = \mathcal{V}_i^h - \mathcal{V}_j^o$, and the projection of the offset vector on object normal $\boldsymbol{n}_j^o$: $\boldsymbol{\Delta l}_{ij}^{\text{rpl}} = (\mathcal{V}_i^h - \mathcal{V}_j^o) \cdot \boldsymbol{n}_j^o$. $\boldsymbol{\Delta l}_{ij}^{\text{rpl}}$ is positive if $\mathcal{V}_i^h$ falls outside the object, and negative if $\mathcal{V}_i^h$ falls inside the object. The attractive force F_{ij}^{atr} and energy E_{ij}^{atr} and the repulsive force F_{ij}^{rpl} and energy E_{ij}^{rpl} is consistent with §4 in the main text. We use an exponential function here to provide magnitude and gradient heuristic for optimizer: **a)** the less $\boldsymbol{\Delta l}_{ij}^{\text{rpl}}$ is, the more $\mathcal{V}_i^h$ penetrate into the object, thus the greater the repulsive force will be. The repulsive force (gradient of repulsive energy) will be an increasing function of $\boldsymbol{\Delta l}_{ij}^{\text{rpl}}$, so that the join force of attraction and repulsion will be asymmetric with respect to the object surface; **b)** when $\mathcal{V}_i^h$ intersects into the object, both the repulsive force and the attractive force will push $\mathcal{V}_i^h$ towards the surface; when $\mathcal{V}_i^h$ is outside the object, the attractive force and repulsive force will point to opposite directions, leading to a balance point outside but in the vicinity to the object's surface.

For more intuitive illustration, we restrict the hand vertex v_h onto the ray which originates from v_o and whose direction is $\boldsymbol{n}_{ij}^o$. Then $\boldsymbol{\Delta l}_{ij}^{\text{rpl}} = \boldsymbol{\Delta l}_{ij}^{\text{atr}} = \mathcal{V}_{ij}^h - \mathcal{V}_{ij}^o$. Under such condition the magnitude of elastic energy E_{ij} with respect to $\boldsymbol{\Delta l}_{ij}^{\text{atr}}$ and $\boldsymbol{\Delta l}_{ij}^{\text{rpl}}$ is illustrated in Fig. 12.

B.2. Anchor Elasticity Assignment

As discussed in §4 (Grasping inside Contact Potential Field.) from the main text. We treat the elasticity of the attractive spring as the network prediction. Therefore, given

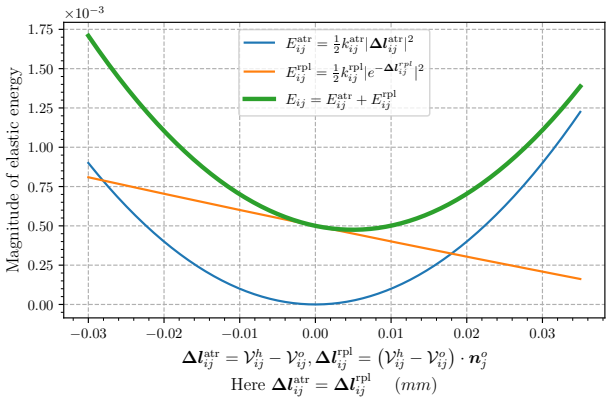

Figure 12. **Illustration on attractive and repulsive elastic energy field.** Here $\left(\mathcal{V}_{ij}^h - \mathcal{V}_{ij}^o\right)$ is in the same direction of $\boldsymbol{n}_j^o$, leading to $\Delta \boldsymbol{l}_{ij}^{\text{atr}} = \Delta \boldsymbol{l}_{ij}^{\text{rpl}}$. The blue and orange curve stands for attractive and repulsive energy respectively, and the thick green curve stands for the combined energy.

the ground-truth hand and object pose, we can manually generate the guidance elasticity as the pseudo ground-truth k^{GT} to supervise the network. First, we set the anchor $\mathcal{A}_i^h$ vertex $\mathcal{V}_j^o$ pair with distance $|\Delta \boldsymbol{l}_{ij}| > 20mm$ as invalid contact and has $k_{ij}^{\text{GT}} = 0$. Second, for those pairs with valid contact, we hope the force asserted to each spring is rather balanced, thus an inverse-proportional k_{ij}^{GT} is assigned according to the $|\Delta \boldsymbol{l}_{ij}|$, where $|\cdot|$ denotes the magnitude:

$$k_{ij}^{\text{GT}} = 0.5 * \cos(\frac{\pi}{20} * |\Delta \boldsymbol{l}_{ij}|) + 0.5 \tag{11}$$

To note, we do not have a strict preference on the function of k_{ij}^{GT}. Any other functions should also work when satisfy: **a)** $k = 1$ when $|\Delta \boldsymbol{l}| = 0$; **b)** k is inverse proportional to $|\Delta \boldsymbol{l}|$ in the range of 0 to 20; **c)** k is bounded by 0 and 1. The choice of cosine function is simply due to its smoothness.

Since we set zero the k_{ij}^{GT} with distance threshold of $20mm$, and assign only one closest region (up-to 4 anchors) for each vertex, the elasticity matrix of the attractive springs $\mathcal{K} \in \mathbb{R}^{N_{\text{anchor}} \times N_O}$ forms a large sparse matrix. Directly supervise neural networks with the guidance $\mathcal{K}^{\text{GT}}$ is inefficient and vulnerable to over-smoothing. Thus as described in §5.2 in the main text, we adopt three cascaded predictions in the PiCR module.

C. HO3D Dataset

C.1. Analysis And Selection

As we mentioned in the main text §6.1, many cases in the test dataset do not suit for evaluating MIHO. Firstly, since GeO requires the predicted 6D pose of the known objects, all the grasps of the *pitcher* have to be removed. Secondly, many interactions of hand and objects in the test dataset are not stable. For example, sliding the palm over the surface of a *bleach cleanser bottle*, may cause a strange CPF and mislead GeO. Therefore, we only select the grasps that can pick up the objects firmly. Table.3 shows our final selection on HO3Dv2 test set, as we called HO3Dv2$^-$.

Sequences	Frame ID
SM1	All
MPM10-14	30-450, 585-685
SB11	340-1355, 1415-1686
SB13	340-1355, 1415-1686

Table 3. **HO3Dv2$^-$ selection.** We select 6076 samples in the HO3Dv2 test set to evaluate MIHO.

C.2. Data Augmentation

We augment the training sample in HO3Dv1 in view of the following protocols: **a)** To generate more poses, we firstly randomize a disturbing transformation to the annotated hand and object pose in the object canonical coordinate system. Then, we render these disturbed meshes to a picture with respect to the camera intrinsic. **b)** To generate more grasps, we fit more stable grasps around the objects. Specifically, as we show in Fig. 13, the generation procedure is achieved by 2 steps: 1) Manually move the hand around the tightest bounding cuboid of the object. 2) Refine the hand pose in the proposed GeO. Since the attractive springs in CPF are unavailable here, we replace the attractive energy in Eq.3 of the main text with the $\mathcal{L}_A$ in Eq.4 of [21], while remaining the repulsive energy. The minimization process during grasping generation can be expressed as:

$$\hat{\mathcal{V}}^h \longleftarrow \underset{(T_w, R_j)}{\arg\min}(\mathcal{L}_A + E^{\text{rpl.}} + \mathcal{L}_{\text{kin}}) \tag{12}$$

D. Experiments and Results

D.1. Implementation Details

In this section we provide more training details about the HoNet and PiCR module.

HoNet. We use pre-trained checkpoints from [20] on FHB and HO3Dv2 and reproduce a comparable model on HO3Dv1. The reproduced HO3Dv1 uses ResNet-18 [22] backbone initialized with ImageNet [10] pretrained weights.

PiCR. We train our PiCR model into 2 stages. PiCR is first pretrained by feeding as input the color-jittered image and the ground-truth object vertices with random rotation and translation disturbance. At pretraining stage, the model is optimized with Adam solver with an initial learning rate 1×10^{-3}, and the learning rate is decayed to 50% every 100

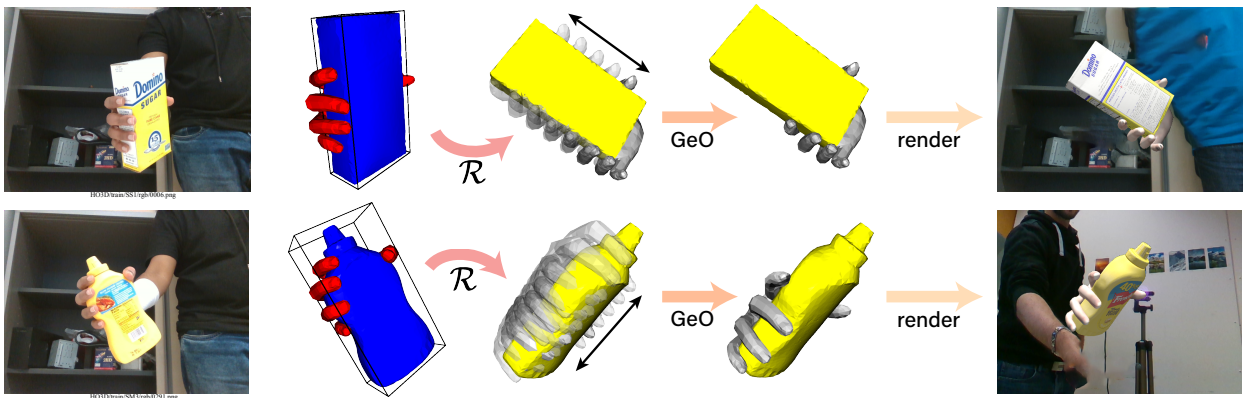

Figure 13. **HO3D [18] Dataset augmentation.** We demonstrate the process of generating synthetic training images.

epochs. The total epochs during pretraining stage are set to be 300. When the pretraining stage is done, its weights are used to initialize the model in fine-tuning stage. At fine-tuning stage, the model is fed with object vertices predictions from a HoNet whose weights have been frozen. The model is optimized with Adam solver with an initial learning rate 5×10^{-4}, decayed to 50% every 100 epochs, and finished at 200 epochs. In both settings, the training batch size is 8 per GPU, and a total of 4 GPUs are used.

D.2. Ablation Study

As referred in §6.5 (Ablation Study) in main text, this section contains another three ablation studies. During all the following experiments, the hand and object are jointly optimized.

The Impact of the k^{rpl}. While the elasticity k^{atr} of the attractive springs are predicted by the PiCR, the elasticity k^{rpl} of those repulsive strings are empirically set to 1×10^{-3}. In order to measure the impact of the magnitude of k^{rpl} on repulsion, we test our GeO with seven experiment settings in which the k^{rpl} is set to $\{0.2,\ 0.6,\ 1.0,\ 1.4,\ 2.0,\ 4.0,\ 8.0\} \times 10^{-3}$, respectively. The experiment with $k^{\text{rpl}} = 1 \times 10^{-3}$ is in accord with the default experiment in main text. As shown in Tab.4, while the large k^{rpl} can reduce the solid interpenetration volume, it may also push the attraction apart thus not preferable in the reconstruction metric of vertex errors.

K-MANO with PCA rotations. Since the MANO can also be driven by the PCA components of joint rotation, we further conduct an experiment to demonstrate the superiority of our full K-MANO (K-MANO with 15 relative joint rotations) over the PCA K-MANO (K-MANO with PCA components of rotations). Tab.5 shows that our full K-MANO can achieve a notable decrease in the reconstruction errors of hand vertices. We attribute this to the fact

k^{rpl}	scores				
	HE $(mm)\downarrow$	OE $(mm)\downarrow$	PD $(mm)\downarrow$	SIV $(cm^3)\downarrow$	D $(mm)\downarrow$
2.0×10^{-4}	19.49	21.57	17.77	13.22	20.85
6.0×10^{-4}	19.51	21.57	17.22	12.40	21.63
$\mathbf{1.0 \times 10^{-3}}$	**19.54**	**21.57**	**16.92**	**11.76**	**22.41**
1.4×10^{-3}	19.59	21.58	16.75	11.00	23.24
2.0×10^{-3}	19.69	21.59	16.41	10.09	24.55
4.0×10^{-3}	20.03	21.63	15.09	7.65	29.33
8.0×10^{-3}	20.95	21.92	12.86	4.27	40.79

Table 4. **Ablation results:** the impact of the magnitude of k^{atr}. HE stands for hand vertex error (mm); OE stands for object vertex error (mm); PD stands for penetration depth (mm); SIV stands for solid intersection volume (cm^3); D stands for disjointedness error (mm).

that the PCA K-MANO tends to recovery a hand that is inclined to a slightly curved mean pose, while our full version imposes higher flexibility on the hand pose. It also can be shown in Tab.5 (col.7) that the PCA K-MANO can improve the abnormality scores. This is due to the fact that MANO's PCA components were directly derived from real human hand scans, which by nature avoid the monstrous poses. Since the kinematic constraints in full K-MANO can still effectively prevent abnormality during fitting (See Fig.7 and §6.5 in the main text), our results are still sound enough to represent valid hand configurations.

settings	scores					
	HE $\downarrow$	OE $\downarrow$	PD $\downarrow$	SIV $\downarrow$	D $\downarrow$	AS $\downarrow$
full K-MANO	**19.54**	**21.57**	**16.92**	**11.76**	**22.41**	**0.2546**
PCA K-MANO	23.32	24.41	22.47	11.90	26.72	0.1781

Table 5. **Ablation results:** the K-MANO with PCA rotations. AS stands for abnormality score.

Fitting w.r.t ObMan Contact Loss. In this experiment, we examine the effectiveness of our elastic potential energy in comparison with the contact loss in ObMan. To ensure a

fair comparison, we replace all the elastic energy from our attractive and repulsive springs to the contact (*attraction* + *repulsion*) losses in ObMan, and examine GeO under those conditions. As shown in Tab. 6, our method surpasses them on both the hand and object reconstruction metrics. To note, since the ObMan *attraction* loss directly optimizes the disjointedness term in our metric, their method shows better resistance on the disjointness over ours. The last column in Tab.6 shows that by applying the CPF with elastic potential energy in GeO, we can save the average time per iteration t_{iter} by 46% compared with the ObMan contact losses.

settings	scores					t_{iter} (ms)
	HE ↓	OE ↓	PD ↓	SIV ↓	D ↓	
with our E_{elast}	**19.54**	**21.57**	**16.92**	**11.76**	**22.41**	**55.77**
with [21] loss	22.15	22.54	15.13	16.20	11.97	103.41

Table 6. **Ablation result:** fitting w.r.t the ObMan Contact Loss.

E. More Qualitative Results

We demonstrate the qualitative results of MIHO in Fig. 14 on both the FHB [15] and HO3D dataset [19]. Note that the ground truth of the test set in HO3Dv2^{-} [19] is not available.

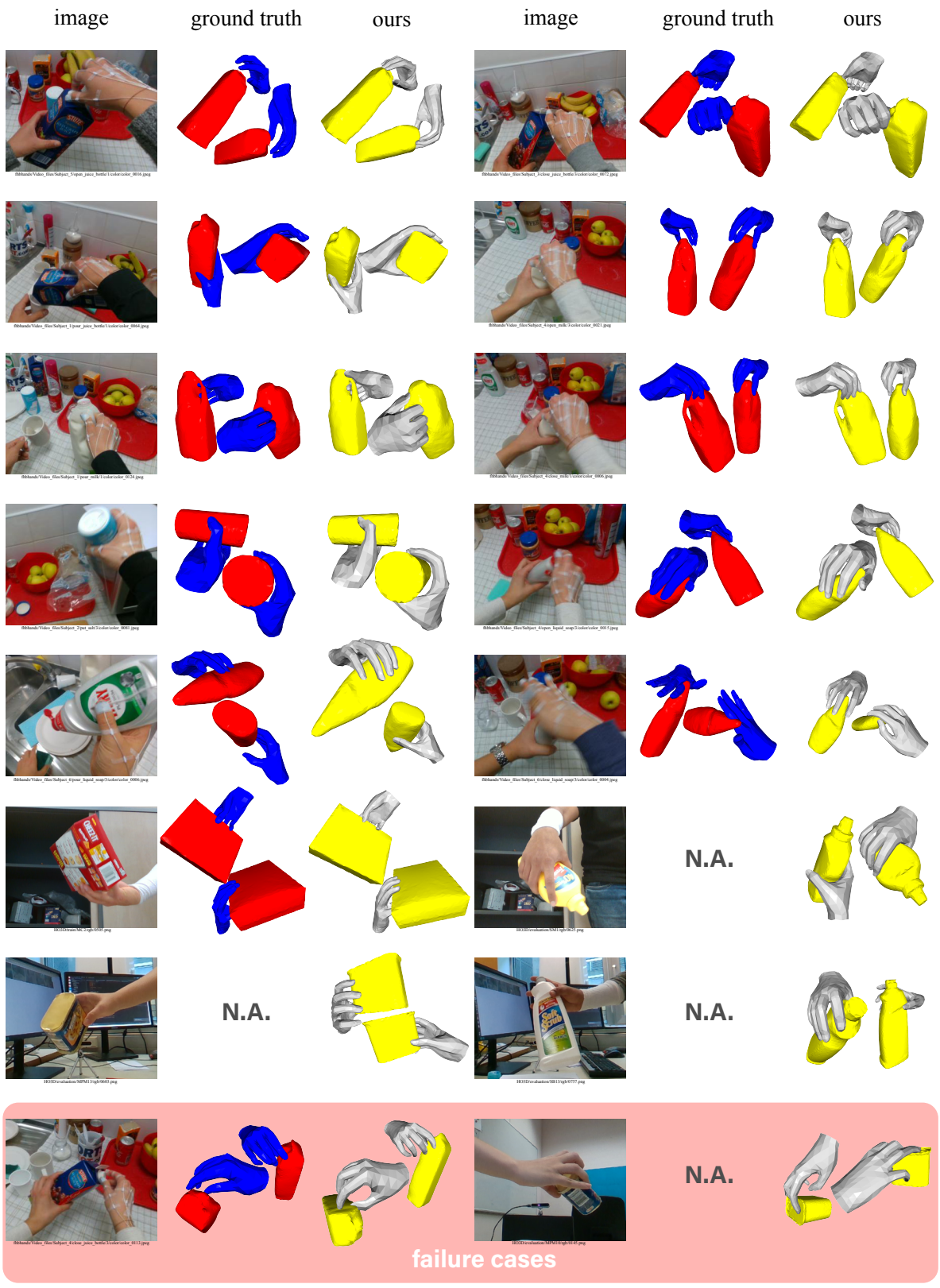

Figure 14. **Qualitative results on FHB [15], HO3Dv1[18] and HO3Dv2$^-$ [19] datasets.** The last row shows the failure cases.