InterHand2.6M: A Dataset and Baseline for 3D Interacting Hand Pose Estimation from a Single RGB Image

Gyeongsik Moon[1], Shoou-I Yu[2], He Wen[2], Takaaki Shiratori[2], and Kyoung Mu Lee[1]

[1] ECE & ASRI, Seoul National University, Korea
[2] Facebook Reality Labs
{mks0601,kyoungmu}@snu.ac.kr, {shoou-i.yu,hewen,tshiratori}@fb.com

Abstract. Analysis of hand-hand interactions is a crucial step towards better understanding human behavior. However, most researches in 3D hand pose estimation have focused on the isolated single hand case. Therefore, we firstly propose (1) a large-scale dataset, InterHand2.6M, and (2) a baseline network, InterNet, for 3D interacting hand pose estimation from a single RGB image. The proposed InterHand2.6M consists of **2.6M labeled single and interacting hand frames** under various poses from multiple subjects. Our InterNet simultaneously performs 3D single and interacting hand pose estimation. In our experiments, we demonstrate big gains in 3D interacting hand pose estimation accuracy when leveraging the interacting hand data in InterHand2.6M. We also report the accuracy of InterNet on InterHand2.6M, which serves as a strong baseline for this new dataset. Finally, we show 3D interacting hand pose estimation results from general images. Our code and dataset are available[1].

1 Introduction

The goal of 3D hand pose estimation is to localize semantic keypoints (*i.e.*, joints) of a human hand in 3D space. It is an essential technique for human behavior understanding and human-computer interaction. Recently, many methods [6,11,15,38,46] utilize deep convolutional neural networks (CNNs) and have achieved noticeable performance improvement on public datasets [29,33,36,43,46].

Most of the previous 3D hand pose estimation methods [6, 11, 15, 38, 46] are designed for single hand cases. Given a cropped single hand image, models estimate the 3D locations of each hand keypoint. However, single hand scenarios have limitations in covering all realistic human hand postures because human hands often interact with each other to interact with other people and objects. To address this issue, we firstly propose a large-scale dataset, *InterHand2.6M*, and a baseline, *InterNet*, for 3D interacting hand pose estimation.

[1] https://mks0601.github.io/InterHand2.6M/

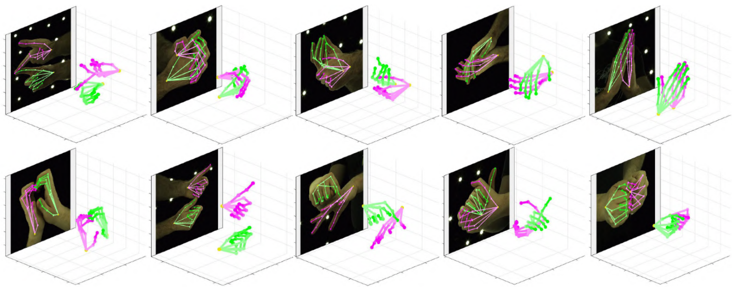

Fig. 1: Qualitative 3D interacting hand pose estimation results from our InterNet on the proposed InterHand2.6M.

Our newly constructed InterHand2.6M is the first large-scale real (*i.e.*, non-synthetic) RGB-based 3D hand pose dataset that includes both single and interacting hand sequences under various poses from multiple subjects. Each hand sequence contains a single hand or interacting right and left hands of a single person. InterHand2.6M is captured in a precisely calibrated multi-view studio with 80 to 140 high-resolution cameras. For 3D keypoint coordinate annotation, we use a semi-automatic approach, which is a combination of manual human annotation and automatic machine annotation. This approach makes annotation procedure much more efficient compared with full manual annotation while achieving similar annotation accuracy as the fully manual one.

The proposed InterNet simultaneously estimates 3D single and interacting hand pose from a single RGB image. For this, we design InterNet to predict handedness, 2.5D right and left hand pose, and right hand-relative left hand depth. The handedness can tell whether right or left hands are included in the input image; therefore InterNet can exclude the pose of a hand that does not exist in the testing stage. The 2.5D hand pose consists of 2D pose in x- and y-axis and root joint (*i.e.*, wrist)-relative depth in z-axis, widely used in state-of-the-art 3D human body [16] and hand [11] pose estimation from a single RGB image. It provides high accuracy because of its image-aligned property and ability to model the uncertainty of the prediction. To lift 2.5D right and left hand pose to 3D space, we obtain an absolute depth of the root joint from RootNet [16]. However, as obtaining absolute depth from a single RGB image is highly ambiguous, RootNet outputs unreliable depth in some cases. To resolve this, we design InterNet to predict right hand-relative left hand depth by leveraging the appearance of the interacting hand from the input image. This relative depth can be used instead of the output of the RootNet when both right and left hands are visible in the input image.

To demonstrate the benefit of the newly captured interacting hand data, we compare the performance of models trained on only single hand data, on only interacting hand data, and on both. We observed that models trained on interacting hand data achieve significantly lower interacting hand pose error than a model trained on single hand data. This comparison shows that interacting hand data is essential for accurate 3D interacting hand pose estimation. We also demonstrate the effectiveness of our dataset for practical purposes by training InterNet on InterHand2.6M and showing its 3D interacting hand pose results from general images. Figure 1 shows 3D interacting hand pose estimation results from our InterNet on the proposed InterHand2.6M.

Our contributions can be summarized as follows.

- Our InterHand2.6M firstly contains large-scale high-resolution multi-view single and interacting hand sequences. By using a semi-automatic approach, we obtained accurate 3D keypoint coordinate annotations efficiently.
- We propose InterNet for 3D single and interacting hand pose estimation. Our InterNet estimates handedness, 2.5D hand pose, and right hand-relative left hand depth from a single RGB image.
- We show that single hand data is not enough, and interacting hand data is essential for accurate 3D interacting hand pose estimation.

2 Related works

Depth-based 3D single hand pose estimation. Early depth-based 3D hand pose estimation methods are mainly based on a generative approach. They fit a pre-defined hand model to the input depth image by minimizing hand-crafted cost functions [25,34]. Particle swarm optimization [25], iterative closest point [31], and their combination [22] are the common algorithms used to obtain optimal hand poses.

Recent deep neural network-based methods are mainly based on a discriminative approach, which directly localizes hand joints from an input depth map. Tompson et al. [36] firstly utilized the deep neural network to localize hand keypoints by estimating 2D heatmaps for each hand joint. Ge et al. [5] extended this method by estimating multi-view 2D heatmaps. Guo et al. [7] proposed a region ensemble network to estimate the 3D coordinates of hand keypoints accurately. Moon et al. [15] designed a 3D CNN model that takes voxel input and outputs a 3D heatmap for each keypoint. Wan et al. [38] proposed a self-supervised system, which can be trained only from an input depth map.

RGB-based 3D single hand pose estimation. Pioneering works [13,39] estimate hand pose from RGB image sequences. Gorce et al. [13] proposed a model that estimates 3D hand pose, texture, and illuminant dynamically. Recently, deep learning-based methods show noticeable improvement. Zimmermann et al. [46] proposed a deep neural network that learns a network-implicit 3D articulation prior. Mueller et al. [17] used an image-to-image translation model to generate a realistic hand pose dataset from a synthetic dataset. Cai et al. [2] and Iqbal et al. [11] implicitly reconstruct depth map and estimate 3D hand keypoint coordinates from it. Spurr et al. [27] and Yang et al. [41] proposed deep generative models to learn latent space for hand.

3D interacting hand pose estimation. There are few works that tried to solve the 3D interacting hand pose estimation. Oikonomidis et al. [20] firstly

dataset	source	resolution	annotation	sub.	fr.	int.hand
ICVL [33]	real depth	320×240	track	10	18K	✗
NYU [36]	real depth	640×480	track	2	243K	✗
MSRA [29]	real depth	320×240	track	9	76K	✗
BigHand2.2M [44]	real depth	640×480	marker	10	2.2M	✗
FPHA [4][2]	real RGBD	1920×1080	marker	6	105K	✗
Dexter+Object [28]	real RGBD	640×480	manual	1	3K	✗
EgoDexter [19]	real RGBD	640×480	manual	4	3K	✗
STB [45]	real RGBD	640×480	manual	1	36K	✗
FreiHAND [47]	real RGB	224×224	semi-auto.	32	134K	✗
RHP [46]	synth. RGBD	320×320	synth.	20	44K	✗
Tzionas et al. [37]	real RGBD	640×480	manual	n/a	36K	✓
Mueller et al. [18]	synth. depth	n/a	synth.	5	80K	✓
Simon et al. [26]	real RGB	1920×1080	semi-auto.	n/a	15K	✓
InterHand2.6M **(ours)**	real RGB	**512×334** **(4096×2668)**	semi-auto.	**27**	**2.6M**	✓

Table 1: Comparison of existing 3D hand pose estimation datasets and the proposed InterHand2.6M. For the RGBD-based datasets, we report their RGB resolution. For the multi-view captured datasets, we consider each image from different views as different images when reporting the number of frames. InterHand2.6M was initially captured at 4096×2668 resolution, but to protect fingerprint privacy, the released set has resolution 512×334.

attempted to address this problem using particle swarm optimization from an RGBD sequence. Ballan et al. [1] presented a framework that outputs 3D hand pose and mesh from multi-view RGB sequences. They combined a generative model with discriminatively trained salient points to achieve a low tracking error. Tzionas et al. [37] extended Ballan et al. [1] by incorporating a physical model. Taylor et al. [35] proposed to perform joint optimization over both the hand model pose and the correspondences between observed data points and the hand model surface. Simon et al. [26] performed 2D hand pose estimation from multi-view images and triangulated them into the 3D space. Mueller et al. [18] proposed a model that estimates a correspondence map and hand segmentation map from a single depth map. The correspondence map provides a correlation between mesh vertices and image pixels, and the segmentation map separates right and left hand. They fit a hand model [23] to the estimated maps.

However, all of the above methods have limitations to be used for 3D single and interacting hand pose estimation from a single RGB image. Tzionas et al. [37] and Simon et al. [26] require additional depth map or multi-view images. The model of Mueller et al. [18] takes a single depth map and not a single RGB image. In contrast, our proposed InterNet can perform 3D single and interacting hand pose estimation simultaneously from a single RGB image.

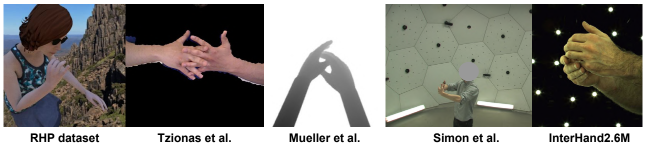

Fig. 2: Comparisons of interacting hand images from RHP [46], Tzionas et al. [37], Mueller et al. [18], Simon et al. [26], and the proposed InterHand2.6M.

3D hand pose estimation datasets. Table 1 shows specification of existing 3D hand pose datasets and the proposed InterHand2.6M. Compared with depth-based 3D hand pose estimation datasets [4, 29, 33, 36, 44], existing RGB-based datasets [19, 28, 45, 46] have very limited number of frames and subjects because obtaining accurate 3D annotation from RGB images is difficult. Recently, Zimmermann et al. [47] captured a large-scale single hand pose and mesh dataset.

Several datasets contain two or interacting hand sequences, and Figure 2 shows example images of the datasets. RHP [46] contains two isolated hand data. However, their images are far from real because they are synthesized by animating 3D human models using commercial software. In addition, in most of their two hand images, right and left hands perform separate actions and are not interacting with each other. The dataset of Tzionas et al. [37] is the most similar dataset with ours in that it is constructed to analyze RGB interacting hand-focused sequences. It contains RGBD interacting hand sequences, however only 2D joint coordinates annotations are available instead of the 3D coordinates. In addition, the scale of the dataset is much smaller compared with that of our dataset. The dataset of Mueller et al. [18] mainly consists of synthesized depth maps, which are not very realistic. Although some depth maps of their dataset are real-captured ones, 3D keypoint coordinate annotations of them are not available. The dataset of Simon et al. [26] is not large-scale, and their annotations from a machine annotator are unstable because the resolution of the hand area of their dataset is low.

Compared with them, our InterHand2.6M consists of large-scale real-captured RGB images and includes more variety of sequences. In addition, our strong machine annotator provides accurate and less jittering 3D hand joints coordinates annotations because of our strong semi-automatic annotation and high-resolution hand area. Our dataset can be used when the hand is a central subject in the input image, for example, capturing the hand by a head-mounted device for virtual/augmented reality.

3 InterHand2.6M

3.1 Data Capture

InterHand2.6M is captured in a multi-camera studio consisting of 80-140 cameras capturing at 30-90 frames-per-second (fps), and 350-450 directional LED point lights directed at the hand to promote uniform illumination[3]. The cameras captured at image resolution 4096 × 2668. The multi-view system was calibrated with a 3D calibration target [8] and achieved pixel root mean square error ranging from 0.42 to 0.48.

We captured a total of 36 recordings consisting of 26 unique subjects, where 19 of them are males, and other 7 are females. There are two types of hand sequences[4]. First, peak pose (PP) is a short transition from neutral pose to pre-defined hand poses (*e.g.*, fist) and then transition back to neutral pose. The pre-defined hand poses include various sign languages that are frequently used in daily life and extreme poses where each finger is maximally bent or extended. There are 40 pre-defined hand poses for each right and left hand, and 13 for the interacting hand. In the neutral pose, hands are in front of the person's chest, fingers do not touch, and palms face the side. The second type is a range of motion (ROM), which represents conversational gestures with minimal instructions. For example, subjects are instructed to wave their hands as if telling someone to come over. There are 15 conversational gestures for each right and left hand, and 17 for the interacting hand. The hand poses from PP and ROM in our dataset are chosen to sample a variety of poses and conversational gestures while being easy to follow by capture participants. The proposed InterHand2.6M is meant to cover a reasonable and general range of hand poses instead of choosing an optimal hand pose set for specific applications.

3.2 Annotation

To annotate keypoints of hands, we directly extend the commonly used 21 keypoints per hand annotation scheme [46] to both hands, thus leading to a total of 42 unique points. For each finger, we annotate the fingertip and the rotation centers of three joints. In addition to the 20 keypoints per hand, the wrist rotation center is also annotated.

Annotating rotation centers is challenging because the rotation center of a joint is occluded by the skin. The annotations become more challenging when the fingers are occluded by other fingers, or viewed from an oblique angle. Therefore, we developed a 3D rotation center annotation tool which allows the annotator to view and annotate 6 images simultaneously[5]. These 6 images are captured at the same time, but viewing the hand from different angles. When the annotator annotates a joint in two views, the tool will automatically perform triangulation and re-project the point to all other views, thus enabling the annotator to verify that the annotations are consistent in 3D space.

Despite having the annotation tool, manually annotating large amounts of images is still very labor-intensive. Thus, we adopted a two-stage procedure to

[3] There were two settings. Setting 1: on average 34 RGB and 46 monochrome cameras (80 cameras total), 350 lights, and 90fps. Setting 2: on average 139 color cameras, 450 lights, and 30fps. Due to camera failures, not all cameras were operational; thus, each capture would have slightly different number of cameras.

[4] The examples of hand sequences are described in supplementary material.

[5] The human annotation procedure is described in supplementary material.

annotate the images following Simon et al. [26]. In the first stage, we rely on human annotators. The annotators leveraged our annotation tool and manually annotated 94,914 2D images from 9,036 unique time instants where 1,880 of them had two hand annotations. These 2D annotations are triangulated to get 3D positions of joints, which are subsequently projected to all roughly 80 views to get 2D annotations for each view. The unique time steps are sampled to cover many hand poses of our recording scripts. At the end of this stage, a total of 698,922 images are labeled with 2D keypoints.

In the second stage, we utilize an automatic machine annotator. For this, we trained a state-of-the-art 2D keypoint detector [14] from the images annotated in the previous stage. EfficientNet [32] is used as a backbone of the keypoint detector for computational efficiency. The detector was then run through unlabeled images, and the 3D keypoints were obtained by triangulation with RANSAC. As our InterHand2.6M is captured from a large number of high-resolution cameras, this machine-based annotation gives highly accurate estimations. We tested this method on the held-out evaluation set, and the error is *2.78 mm*. The final dataset is an integration of human annotations from the first stage and machine annotations from the second stage. Simon et al. [26] performed iterative bootstrap because their initial machine annotator does not provide accurate annotations, and the hand area of their dataset has low resolution. In contrast, our strong machine annotator on high-resolution hand images achieves significantly low error (2.78 mm); therefore, we did not perform iterative bootstrap.

3.3 Dataset release

The captured hand sequences will be released under two configurations: downsized 512×334 image resolution at 5 fps, and downsized 512×334 resolution at 30 fps. Downsizing is to protect fingerprint privacy. The annotation file includes camera type, subject index, camera index, bounding box, handedness, camera parameters, and 3D joint coordinates. All reported frame numbers and experimental results in the paper are from the 5 fps configuration.

4 InterNet

Our InterNet takes a single RGB image $\mathbf{I}$ as an input and extracts the image feature $\mathbf{F}$ using ResNet [9] whose fully-connected layers are trimmed. We prepare $\mathbf{I}$ by cropping the hand region from an image and resizing it to uniform resolution. From $\mathbf{F}$, InterNet simultaneously predicts handedness, 2.5D right and left hand pose, and right hand-relative left hand depth, which will be described in the following subsections. We do not normalize the hand scale for the 2.5D hand pose estimation. Figure 3 shows overall pipeline of InterNet.

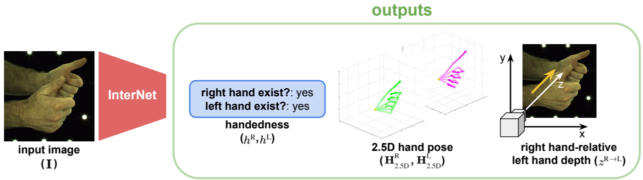
outputs
input image
(**I**)
InterNet
right hand exist?: yes
left hand exist?: yes
handedness
(h^{R}, h^{L})
2.5D hand pose
($\mathbf{H}^{\mathrm{R}}_{\mathrm{2.5D}}$, $\mathbf{H}^{\mathrm{L}}_{\mathrm{2.5D}}$)
y
x
z
right hand-relative
left hand depth ($z^{\mathrm{R}\rightarrow\mathrm{L}}$)

Fig. 3: Three outputs of the proposed InterNet.

4.1 Handedness estimation

To decide which hand is included in the input image, we design our InterNet to estimate the probability of the existence of the right and left hand $\mathbf{h} = (h^{\mathrm{R}}, h^{\mathrm{L}}) \in \mathbb{R}^2$ in the input image. For this, we build two fully-connected layers, which take the image feature $\mathbf{F}$ and estimates the probabilities $\mathbf{h}$. The hidden activation size of the fully-connected layers is 512. Each fully-connected layer is followed by the ReLU activation function except for the last one. We apply a sigmoid activation function at the last layer to get the probabilities.

4.2 2.5D right and left hand pose estimation

To estimate 2.5D right and left hand pose, denoted as $\mathbf{P}^{\mathrm{R}}_{\mathrm{2.5D}} \in \mathbb{R}^{J\times 3}$ and $\mathbf{P}^{\mathrm{L}}_{\mathrm{2.5D}} \in \mathbb{R}^{J\times 3}$, respectively, we construct two upsamplers for each right and left hand. Each upsampler consists of three deconvolutional and one convolutional layers, and each deconvolutional layer is followed by batch normalization layers [10] and ReLU activation functions, therefore it upsamples the input feature map 2^3 times. The upsamplers take $\mathbf{F}$ and output 3D Gaussian heatmaps of the right and left hand joints, denoted as $\mathbf{H}^{\mathrm{R}}_{\mathrm{2.5D}}$ and $\mathbf{H}^{\mathrm{L}}_{\mathrm{2.5D}}$ following Moon et al. [16], where they have the same dimension $\mathbb{R}^{J\times D\times H\times W}$. D, H, and W denote depth discretization size, height, and width of the heatmaps, respectively. x- and y-axis of $\mathbf{H}^{\mathrm{R}}_{\mathrm{2.5D}}$ and $\mathbf{H}^{\mathrm{L}}_{\mathrm{2.5D}}$ are in image space, and z-axis of them are in root joint (*i.e.*, wrist)-relative depth space. To obtain a 3D Gaussian heatmap from the 2D feature map, we reshape the output of the upsampler by a reshaping function $\psi\colon \mathbb{R}^{JD\times H\times W} \rightarrow \mathbb{R}^{J\times D\times H\times W}$. Each voxel of the 3D Gaussian heatmap of the joint j represents the likelihood of the existence of a hand joint j in that position.

4.3 Right hand-relative left hand depth estimation

The depth of each hand is defined as that of the hand root joint. We construct two fully-connected layers and the ReLU activation function after each fully connected layer except for the last layer. The hidden activation size of the fully-connected layers is 512. It takes $\mathbf{F}$ and outputs 1D heatmap $\mathbf{d}^{\mathrm{R}\rightarrow\mathrm{L}} \in \mathbb{R}^{64}$. Then, soft-argmax [30] is applied to $\mathbf{d}^{\mathrm{R}\rightarrow\mathrm{L}}$ and output the relative depth value $z^{\mathrm{R}\rightarrow\mathrm{L}}$. We observed that estimating the 1D heatmap followed by soft-argmax operation provides a more accurate relative depth value compared with directly regressing it, which is a similar spirit to Moon et al. [15].

4.4 Final 3D interacting hand pose

The final 3D hand pose $\mathbf{P}^{\mathrm{R}}_{\mathrm{3D}}$ and $\mathbf{P}^{\mathrm{L}}_{\mathrm{3D}}$ are obtained as follows:

$$\mathbf{P}^{\mathrm{R}}_{\mathrm{3D}} = \Pi(\mathbf{T}^{-1}\mathbf{P}^{\mathrm{R}}_{\mathrm{2.5D}} + \mathbf{Z}^{\mathrm{R}}), \quad \text{and} \quad \mathbf{P}^{\mathrm{L}}_{\mathrm{3D}} = \Pi(\mathbf{T}^{-1}\mathbf{P}^{\mathrm{L}}_{\mathrm{2.5D}} + \mathbf{Z}^{\mathrm{L}}),$$

where Π, $\mathbf{T}^{-1}$, and $\emptyset$ denote camera back-projection, inverse affine transformation (*i.e.*, 2D crop and resize), and empty pose set, respectively. We use normalized camera intrinsic parameters if not available following Moon et al. [16]. $\mathbf{Z}^{\mathrm{R}} \in \mathbb{R}^{1\times 3}$ and $\mathbf{Z}^{\mathrm{L}} \in \mathbb{R}^{1\times 3}$ are defined as follows:

$$\mathbf{Z}^{\mathrm{R}} = [(0),(0),(z^{\mathrm{R}})], \qquad \mathbf{Z}^{\mathrm{L}} = \begin{cases} [(0),(0),(z^{\mathrm{L}})], & \text{if } h^{\mathrm{R}} < 0.5 \\ [(0),(0),(z^{\mathrm{R}} + z^{\mathrm{R}\to\mathrm{L}})], & \text{otherwise,} \end{cases}$$

where z^{R} and z^{L} denote the absolute depth of the root joint of right and left hand, respectively. We use RootNet [16] to obtain them.

4.5 Loss functions

To train our InterNet, we use three loss functions.
Handedness loss. For the handedness estimation, we use binary cross-entropy loss function as defined as follows: $L_{\mathrm{h}} = -\frac{1}{2}\sum_{\mathcal{Q}\in(\mathrm{R},\mathrm{L})}(\delta^{\mathcal{Q}} \log h^{\mathcal{Q}} + (1-\delta^{\mathcal{Q}})\log(1-h^{\mathcal{Q}}))$, where $\delta^{\mathcal{Q}}$ is a binary value which represents existence of the $\mathcal{Q}$ hand in an input image.
2.5D hand pose loss. For the 2.5D hand pose estimation, we use $L2$ loss as defined as follows: $L_{\mathrm{pose}} = \sum_{\mathcal{Q}\in(\mathrm{R},\mathrm{L})} ||\mathbf{H}^{\mathcal{Q}}_{\mathrm{2.5D}} - \mathbf{H}^{\mathcal{Q}*}_{\mathrm{2.5D}}||_2$, where $*$ denotes groundtruth. If one of the right or left hand is not included in the input image, we set the loss from it zero. The groundtruth 3D Gaussian heatmap is computed using a Gaussian blob [15] as follows:
$\mathbf{H}^{\mathcal{Q}*}_{\mathrm{2.5D}}(j,z,x,y) = \exp\left(-\frac{(x-x^{\mathcal{Q}}_j)^2+(y-y^{\mathcal{Q}}_j)^2+(z-z^{\mathcal{Q}}_j)^2}{2\sigma^2}\right)$, where $x^{\mathcal{Q}}_j$, $y^{\mathcal{Q}}_j$, and $z^{\mathcal{Q}}_j$ are jth joint coordinates of $\mathcal{Q}$ hand from $\mathbf{P}^{\mathcal{Q}}_{\mathrm{2.5D}}$.
Right hand-relative left hand depth loss. For the right hand-relative left hand localization, we use $L1$ loss as defined as follows: $L_{\mathrm{rel}} = |z^{\mathrm{R}\to\mathrm{L}} - z^{\mathrm{R}\to\mathrm{L}*}|$, where $*$ denotes groundtruth. The loss becomes zero when only a single hand is included in the input image.

We train our model in an end-to-end manner using all the three loss functions as follows: $L = L_{\mathrm{h}} + L_{\mathrm{pose}} + L_{\mathrm{rel}}$.

5 Implementation details

PyTorch [21] is used for implementation. The backbone part is initialized with the publicly released ResNet-50 [9] pre-trained on the ImageNet dataset [24], and the weights of the remaining part are initialized by Gaussian distribution with $\sigma = 0.001$. The weights are updated by the Adam optimizer [12] with a mini-batch size of 64. To crop the hand region from the input image, we use

split	sequence	subjects	frames (SH)	frames (IH)	frames (All)
Train (H)	PP+ROM	16	142K	386K	528K
Train (M)	PP+ROM	9	594K	315K	909K
Train (H+M)	PP+ROM	21	688K	674K	**1361K**
Val (M)	ROM	1	234K	146K	**380K**
Test (H)	PP+ROM	6	34K	88K	122K
Test (M)	ROM	2	455K	272K	728K
Test (H+M)	PP+ROM	8	489K	360K	**849K**

Table 2: Training, validation, and test set split of the proposed InterHand2.6M. H and M denote human annotation and machine annotation, respectively. SH and IH denote single and interacting hand, respectively.

groundtruth bounding box in both of training and testing stages. The cropped hand image is resized to 256×256; thus the spatial size of the heatmap is $H \times W = 64 \times 64$. We set $D = 64$. Data augmentations including translation (±15%), scaling (±25%), rotation (±90°), horizontal flip, and color jittering (±20%) is performed in training. The initial learning rate is set to 10^{-4} and reduced by a factor of 10 at the 15th and 17th epoch. We train our model for 20 epochs with four NVIDIA TitanV GPUs, which takes 48 hours when training on our InterHand2.6M. Our InterNet runs at a speed of 53 fps.

6 Experiment

6.1 Dataset and evaluation metric

STB. STB [45] includes 6 pairs of stereo sequences of diverse poses with different backgrounds from a single person. For evaluation, end point error (EPE) is widely used, which is defined as a mean Euclidean distance (mm) between the predicted and ground-truth 3D hand pose after root joint alignment.

RHP. RHP [46] has a large number of synthesized images. They used 3D human models of 20 different subjects to synthesize 39 actions. For the evaluation metric, EPE is used.

InterHand2.6M. InterHand2.6M is our newly captured 3D hand pose dataset. We split our InterHand2.6M into training, validation, and test set, as shown in Table 2. Val (M) and Test (M) contain many unseen hand poses and only subjects not seen in Train (H+M). Also, Val (M) and Test (M) only consists of ROM, which includes longer and more diverse sequences than that of Train (H+M). This can make Val (M) and Test (M) more similar to real-world scenarios. Test (H) contains many seen hand poses, and half of the subjects are seen in Train (H). There are duplicated frames and annotations in Train (H) and Train (M), and we overwrite them with Train (H).

For the evaluation, we introduce three metrics. First, we use the average precision of handedness estimation ($\mathrm{AP_h}$) to measure the accuracy of handedness

training set	SH MPJPE	IH MPJPE
SH only	13.08	51.19
IH only	13.70	16.86
SH+IH (ours)	**12.16**	**16.02**

Table 3: Single and interacting hand MPJPE comparison from models trained on the different training sets. SH and IH denote single and interacting hand, respectively.

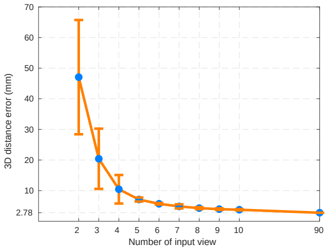

Fig. 4: The 3D distance error of our machine annotation on the Test (H).

estimation. Second, the mean per joint position error (MPJPE) is defined as a Euclidean distance (mm) between predicted and groundtruth 3D joint locations after root joint alignment. The root joint alignment is performed for each left and right hand separately. This metric measures how accurately the root-relative 3D hand pose estimation is performed. Last, mean relative-root position error (MRRPE) is defined as a Euclidean distance (mm) between predicted and groundtruth right hand root-relative left hand root position. It measures how right hand-relative left hand localization is accurately performed.

6.2 Ablation study

Benefit of the interacting hand data. To investigate the benefit of the interacting hand data for 3D interacting hand pose estimation, we compare single and interacting hand MPJPE of our InterNet trained with and without interacting hand data in Table 3. For all settings, we used the same RootNet trained on both single and interacting hand data. As the table shows, a model trained only on interacting hand provides significantly lower interacting hand pose error than a model trained only on single hand data. This shows existing 3D single hand pose estimation datasets are not enough for accurate 3D interacting hand pose estimation. In addition, we trained a model on combined single and interacting hand data, which is our InterHand2.6M. We observed that additional interacting hand data improves not only interacting hand pose estimation performance, but also single hand pose estimation. These comparisons clearly show the benefit of our newly introduced interacting hand data for 3D single and interacting hand pose estimation.

Accuracy of the machine annotation. To show the accuracy of our machine annotation model, we train our annotation model on Train (H) and test on Test (H). Figure 4 shows 3D distance error (mm) on Test (H) according to the number of input views. For each number of input views, the vertical line represents a standard deviation, which shows performance variation due to view selection. In the testing time, the model takes randomly selected v views and performs 2D hand pose estimation, followed by triangulation. To cover various combinations

training set	Val (M)	Test (H)	Test (M)	Test (H+M)
Train (H)	15.02/19.70	10.42/13.05	12.74/18.10	12.58/17.16
Train (M)	15.36/20.13	10.64/14.26	12.56/18.59	12.43/17.79
Train (H+M)	**14.65/18.58**	**9.85/12.29**	**12.32/16.88**	**12.16/16.02**

Table 4: Single and interacting hand MPJPE comparison from models trained on the different training sets. The numbers on the left of the slash are single hand, and the ones on the right are interacting hand MPJPE.

of selecting v views from all V views, we repeat the same testing procedure 100 times for each v views. The figure shows that as the number of input views increases, both the error and standard deviation becomes smaller, and finally, the error becomes *2.78 mm* when all 90 views are used. This shows our annotation method is highly accurate by utilizing state-of-the-art 2D keypoint detection network and a large number of views for triangulation.

Benefit of machine-generated annotation. To show the benefit of the automatically obtained machine annotations, we compare the accuracy of models trained without and with Train (M) in Table 4. As the table shows, a model trained on Train (H) achieves better performance than a model trained on Train (M). We hypothesize that although our machine annotator has very low 3D distance error, human annotation is still more accurate, which makes a model trained on Train (H) performs better. However, as the machine provides annotation more efficiently than a human, it can annotate many frames easily that may not be included in Train (H). Therefore, this machine-generated annotations can have better coverage of hand pose space, which can be a benefit in the training stage. To utilize this better coverage, we add machine annotation to the human annotation. The last row of the table shows that a model trained on the combined dataset achieves the best accuracy. This comparison clearly shows the benefit of adding the machine-generated annotation to the human annotation. We provide more analysis of human and machine annotation in the supplementary material.

Benefit of using $z^{\mathbf{R}\rightarrow\mathbf{L}}$. To show the benefit of using $z^{\mathrm{R}\rightarrow\mathrm{L}}$ when right hand is visible (*i.e.*, $h^{\mathrm{R}} \geq 0.5$) instead of always using z^{L}, we compare MRRPE between the two cases. We checked that MRRPE of always using z^{L} is 92.14 mm, while that of using $z^{\mathrm{R}\rightarrow\mathrm{L}}$ when $h^{\mathrm{R}} \geq 0.5$ is 32.57 mm. This is because estimating z^{L} from a cropped single image inherently involves high depth ambiguity because the camera position is not provided in the cropped input image. In contrast, estimating $z^{\mathrm{R}\rightarrow\mathrm{L}}$ from a cropped image involves less depth ambiguity because both hands are visible in the cropped input image.

6.3 Comparison with state-of-the-art methods

We compare the performance of our InterNet with previous state-of-the-art 3D hand pose estimation methods on the STB and RHP in Table 5. The table shows

methods	GT S	GT H	EPE (STB)	EPE (RHP)
Zimm. et al. [46]	✓	✓	8.68	30.42
Chen et al. [3]	✓	✓	10.95	24.20
Yang et al. [41]	✓	✓	8.66	19.95
Spurr et al. [27]	✓	✓	8.56	19.73
Spurr et al. [27]	✗	✗	9.49	22.53
InterNet (ours)	✗	✗	**7.95**	**20.89**

Table 5: EPE comparison with previous state-of-the-art methods on STB and RHP. The checkmark denotes a method use groundtruth information during inference time. S and H denote scale and handness, respectively.

the proposed InterNet outperforms previous methods without relying on groundtruth information during inference time. Our InterNet estimates 3D heatmap of each joint, while other methods directly estimate 3D joint coordinates. As shown in Moon et al. [15], directly regressing 3D joint coordinates from an input image is a highly non-linear mapping. In contrast, our InterNet estimates per-voxel likelihoods, which makes learning easier and provides state-of-the-art performance.

6.4 Evaluation on InterHand2.6M

Table 4 shows 3D errors of InterNet on InterHand2.6M. Table 3 shows that InterNet trained on both single and interacting hand data yields the 32% larger error on interacting hand sequences than single hand sequences. This comparison tells us that interacting hand sequences are harder to analyze than single hand cases. To analyze the difficulty of InterHand2.6M, we compare our error with 3D hand pose error of current state-of-the-art depth map-based 3D hand pose estimation methods [15, 40] on the large-scale depth map 3D hand pose datasets [42, 44]. They achieved $8 \sim 9$ mm error on large scale depth map dataset [42,44], which is far less than 3D interacting hand pose estimation error of our InterNet (*i.e.*, 16.02 mm). Considering our InterNet achieves state-of-the-art performance on publicly available datasets [45,46], we can conclude that 3D interacting hand pose estimation from a single RGB image is far from solved. Our InterNet achieves 99.09 $\mathrm{AP_h}$ and 32.57 MRRPE on Test (H+M).

6.5 3D interacting hand pose estimation from general images

We show 3D interacting hand pose estimation results from general images in Figure 5. For this, we additionally utilize the dataset of Tzionas et al. [37], which is captured from the general environment but only provides the 2D groundtruth joints coordinates. We randomly split the dataset of Tzionas et al. [37] at a 9:1 ratio as a training and testing set, respectively. During the training, a mini-batch consists of half-InterHand2.6M and half-dataset of Tzionas et al. [37]. For the simultaneous 3D and 2D supervision from our dataset and that of Tzionas et

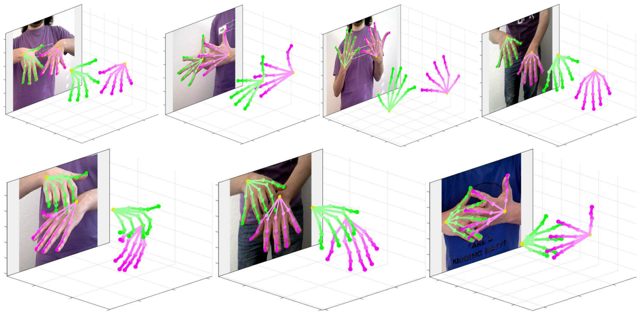

Fig. 5: Qualitative results on the dataset of Tzionas et al. [37], which is captured from a general environment.

al. [37], respectively, we perform soft-argmax [30] on the estimated heatmaps $\mathbf{H}^{\mathrm{R}}$ and $\mathbf{H}^{\mathrm{L}}$ to extract the 3D coordinates in a differentiable way. Then, we modify L_{pose} to a $L1$ distance between the extracted 3D coordinates and the groundtruth. We set a loss of z-axis coordinate to zero when the input image is from the dataset of Tzionas et al. [37]. The figure shows our InterNet successfully produces 3D interacting hand pose results from general images from the dataset of Tzionas et al. [37], although the 3D supervision is only applied to the data from our InterHand2.6M.

7 Conclusion

We propose a baseline, InterNet, and dataset, InterHand2.6M, for 3D interacting hand pose estimation from a single RGB image. The proposed InterHand2.6M is the first large-scale 3D hand pose dataset that includes various single and interacting hand sequences from multiple subjects. As InterHand2.6M only provides 3D hand joint coordinates, fitting 3D hand model [23] to our dataset for the 3D rotation and mesh data of interacting hand can be interesting future work.

Acknowledgments. We would like to thank Alexander Hypes, David Whitewolf, Eric Brockmeyer, Kevyn McPhail, Mark Pitts, Matt Stewart, Michal Perdoch, Scott Ardisson, Steven Krenn, and Timothy Godisart for building the capture system, Autumn Trimble, Danielle Belko, Junko Saragih, Laura Millerschoen, Lucas Evans, Rohan Bali, Taylor Koska, and Xiaomin Luo for the data capture and annotation efforts, and Chenglei Wu, Jason Saragih, Tomas Simon, and Yaser Sheikh for constructive feedback on data collection and the paper. This work was partially supported by the Next-Generation Information Computing Development Program (NRF-2017M3C4A7069369) and the Visual Turing Test project (IITP-2017-0-01780) funded by the Ministry of Science and ICT of Korea.

References

1. Ballan, L., Taneja, A., Gall, J., Van Gool, L., Pollefeys, M.: Motion capture of hands in action using discriminative salient points. In: ECCV (2012)
2. Cai, Y., Ge, L., Cai, J., Yuan, J.: Weakly-supervised 3D hand pose estimation from monocular RGB images. In: ECCV (2018)
3. Chen, L., Lin, S.Y., Xie, Y., Tang, H., Xue, Y., Xie, X., Lin, Y.Y., Fan, W.: Generating realistic training images based on tonality-alignment generative adversarial networks for hand pose estimation. arXiv preprint arXiv:1811.09916 (2018)
4. Garcia-Hernando, G., Yuan, S., Baek, S., Kim, T.K.: First-person hand action benchmark with RGB-D videos and 3D hand pose annotations. In: CVPR (2018)
5. Ge, L., Liang, H., Yuan, J., Thalmann, D.: Robust 3D hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In: CVPR (2016)
6. Ge, L., Liang, H., Yuan, J., Thalmann, D.: 3D convolutional neural networks for efficient and robust hand pose estimation from single depth images. In: CVPR (2017)
7. Guo, H., Wang, G., Chen, X., Zhang, C., Qiao, F., Yand, H.: Region ensemble network: Improving convolutional network for hand pose estimation. ICIP (2017)
8. Ha, H., Perdoch, M., Alismail, H., So Kweon, I., Sheikh, Y.: Deltille grids for geometric camera calibration. In: CVPR (2017)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
10. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. ICML (2015)
11. Iqbal, U., Molchanov, P., Breuel Juergen Gall, T., Kautz, J.: Hand pose estimation via latent 2.5 d heatmap regression. In: ECCV (2018)
12. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ICLR (2014)
13. de La Gorce, M., Fleet, D.J., Paragios, N.: Model-based 3D hand pose estimation from monocular video. IEEE TPAMI (2011)
14. Li, W., Wang, Z., Yin, B., Peng, Q., Du, Y., Xiao, T., Yu, G., Lu, H., Wei, Y., Sun, J.: Rethinking on multi-stage networks for human pose estimation. arXiv preprint arXiv:1901.00148 (2019)
15. Moon, G., Ju, Y.C., Lee, K.M.: V2V-PoseNet: Voxel-to-voxel prediction network for accurate 3D hand and human pose estimation from a single depth map. In: CVPR (2018)
16. Moon, G., Ju, Y.C., Lee, K.M.: Camera distance-aware top-down approach for 3D multi-person pose estimation from a single RGB image. In: ICCV (2019)
17. Mueller, F., Bernard, F., Sotnychenko, O., Mehta, D., Sridhar, S., Casas, D., Theobalt, C.: Ganerated hands for real-time 3D hand tracking from monocular RGB. In: CVPR (2018)
18. Mueller, F., Davis, M., Bernard, F., Sotnychenko, O., Verschoor, M., Otaduy, M.A., Casas, D., Theobalt, C.: Real-time pose and shape reconstruction of two interacting hands with a single depth camera. ACM TOG (2019)
19. Mueller, F., Mehta, D., Sotnychenko, O., Sridhar, S., Casas, D., Theobalt, C.: Real-time hand tracking under occlusion from an egocentric RGB-D sensor. In: ICCV (2017)
20. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Tracking the articulated motion of two strongly interacting hands. In: CVPR (2012)
21. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)

22. Qian, C., Sun, X., Wei, Y., Tang, X., Sun, J.: Realtime and robust hand tracking from depth. In: CVPR (2014)
23. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM TOG (2017)
24. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. IJCV (2015)
25. Sharp, T., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., Rhemann, C., Leichter, I., Vinnikov, A., Wei, Y., et al.: Accurate, robust, and flexible realtime hand tracking. In: ACM Conference on Human Factors in Computing Systems (2015)
26. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: CVPR (2017)
27. Spurr, A., Song, J., Park, S., Hilliges, O.: Cross-modal deep variational hand pose estimation. In: CVPR (2018)
28. Sridhar, S., Mueller, F., Zollhöfer, M., Casas, D., Oulasvirta, A., Theobalt, C.: Real-time joint tracking of a hand manipulating an object from RGB-D input. In: ECCV (2016)
29. Sun, X., Wei, Y., Liang, S., Tang, X., Sun, J.: Cascaded hand pose regression. In: CVPR (2015)
30. Sun, X., Xiao, B., Wei, F., Liang, S., Wei, Y.: Integral human pose regression. In: ECCV (2018)
31. Tagliasacchi, A., Schröder, M., Tkach, A., Bouaziz, S., Botsch, M., Pauly, M.: Robust articulated-icp for real-time hand tracking. In: Computer Graphics Forum (2015)
32. Tan, M., Le, Q.V.: EfficientNet: Rethinking model scaling for convolutional neural networks. In: ICML (2019)
33. Tang, D., Jin Chang, H., Tejani, A., Kim, T.K.: Latent regression forest: Structured estimation of 3D articulated hand posture. In: CVPR (2014)
34. Tang, D., Taylor, J., Kohli, P., Keskin, C., Kim, T.K., Shotton, J.: Opening the black box: Hierarchical sampling optimization for estimating human hand pose. In: ICCV (2015)
35. Taylor, J., Bordeaux, L., Cashman, T., Corish, B., Keskin, C., Sharp, T., Soto, E., Sweeney, D., Valentin, J., Luff, B., et al.: Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. ACM Transactions on Graphics (TOG) (2016)
36. Tompson, J., Stein, M., Lecun, Y., Perlin, K.: Real-time continuous pose recovery of human hands using convolutional networks. ACM TOG (2014)
37. Tzionas, D., Ballan, L., Srikantha, A., Aponte, P., Pollefeys, M., Gall, J.: Capturing hands in action using discriminative salient points and physics simulation. IJCV (2016)
38. Wan, C., Probst, T., Gool, L.V., Yao, A.: Self-supervised 3D hand pose estimation through training by fitting. In: CVPR (2019)
39. Wu, Y., Lin, J., Huang, T.S.: Analyzing and capturing articulated hand motion in image sequences. IEEE TPAMI (2005)
40. Xiong, F., Zhang, B., Xiao, Y., Cao, Z., Yu, T., Zhou, J.T., Yuan, J.: A2J: Anchor-to-joint regression network for 3D articulated pose estimation from a single depth image. In: ICCV (2019)
41. Yang, L., Yao, A.: Disentangling latent hands for image synthesis and pose estimation. In: CVPR (2019)

42. Yuan, S., Garcia-Hernando, G., Stenger, B., Moon, G., Yong Chang, J., Mu Lee, K., Molchanov, P., Kautz, J., Honari, S., Ge, L., Yuan, J., Chen, X., Wang, G., Yang, F., Akiyama, K., Wu, Y., Wan, Q., Madadi, M., Escalera, S., Li, S., Lee, D., Oikonomidis, I., Argyros, A., Kim, T.K.: Depth-based 3D hand pose estimation: From current achievements to future goals. In: CVPR (2018)
43. Yuan, S., Ye, Q., Garcia-Hernando, G., Kim, T.K.: The 2017 hands in the million challenge on 3D hand pose estimation. arXiv preprint arXiv:1707.02237 (2017)
44. Yuan, S., Ye, Q., Stenger, B., Jain, S., Kim, T.K.: Bighand2.2M benchmark: Hand pose dataset and state of the art analysis. In: CVPR (2017)
45. Zhang, J., Jiao, J., Chen, M., Qu, L., Xu, X., Yang, Q.: 3D hand pose tracking and estimation using stereo matching. arXiv preprint arXiv:1610.07214 (2016)
46. Zimmermann, C., Brox, T.: Learning to estimate 3D hand pose from single RGB images. In: ICCV (2017)
47. Zimmermann, C., Ceylan, D., Yang, J., Russell, B., Argus, M., Brox, T.: FreiHand: A dataset for markerless capture of hand pose and shape from single RGB images. In: ICCV (2019)