Supplementary Material for Hand Pose Estimation in Egocentric Images in the Wild

Aditya Prakash Matthew Chang Matthew Jin Saurabh Gupta
University of Illinois Urbana-Champaign
bit.ly/WildHands

In this document, we first provide additional implementation details about the architecture (Sec. 1.1) & training protocol (Sec. 1.2). We then present additional ablations (Sec. 2) & describe our EPIC-HandKps dataset (Sec. 3). Next, we provide qualitative comparisons (Sec. 4) of our WildHands model with FrankMocap [19] on EPIC-HandKps (Fig. 2) & with ArcticNet [6] on ARCTIC (Fig. 4). We also show failure cases of WildHands on both EPIC-HandKps (Fig. 3) & ARCTIC (Fig. 5) and more visualizations of perspective distortion induced ambiguity in hands (Fig. 6). Moreover, the video contains 3D hand mesh visualizations and a summary of our key contributions.

1. Implementation Details

1.1. Architecture

We build on top of the ArcticNet-SF [6] and FrankMocap [19] models. Here, we provide additional details about the encoder, intrinsics-aware positional encoding (KPE [16]), decoder, bounding box predictor used in the architecture.

Encoder: Our models uses hand crops as input (resized to 224×224 resolution), which are processed by a ResNet50 [8] backbone to get $7 \times 7 \times 2048$ feature maps. The left and right hand crops as processed separately but the parameters are shared. We also use global image features in our model, computed by average pooling the $7 \times 7 \times 2048$ feature map to get a 2048-dimensional vector.

Incorporating KPE encoding: We add the intrinsics-aware positional encoding (KPE [16]) to the $7 \times 7 \times 2048$ feature map. KPE comprises sinusoidal encoding of the angles θ_x and θ_y (Sec. 4.1 in the main paper), resulting in $5 * 4 * K$ dimensional sparse encoding (4 for corners and 1 for center pixel) and $H \times W \times 4 * K$ resolution dense encoding, where K is the number of frequency components (set to 4). For the sparse KPE variant, we broadcast it to 7×7 resolution whereas for the dense KPE variant, we interpolate it to 7×7 resolution and concatenate to the feature map. This concatenated feature is passed to a 3 convolutional layers (with 1024, 512, 256 channels respectively, each with kernel size of 3×3 and ReLU [1, 14] non-linearity) to get a $3 \times 3 \times 256$ feature map. This is flattened to 2304-dimensional vector and passed through a 1-layer MLP to get a 2048-dimensional feature vector. We do not use batchnorm [10] here since we want to preserve the spatial information in the KPE encoding whereas normalization would deteriorate it.

Decoder: It consists of an iterative architecture, similar to decoder in HMR [11]. The inputs are the 2048-dimensional feature vector and initial MANO [18] (shape β, articulation θ_{local} and global pose θ_{global}, all initialized as 0-vectors) & weak perspective camera parameters (initialized from the 2048-dimensional feature vector). Each of these parameters are predicted using a separate decoder head. The rotation parameters θ_{local}, θ_{global} are predicted in matrix form and converted to axis-angle representation to feed to MANO model. Each decoder is a 3-layer MLP with the 2 intermediate layers having 1024 channels and the output layer having the same number of channels as the predicted parameter. The output of each decoder is added to the initial parameters to get the updated parameters. This process is repeated for 3 iterations. The output of the last iteration is used for the final prediction.

For the auxiliary supervision used to train our model, we also predict hand segmentation masks and grasp labels. The hand parameters β, θ_{local}, and θ_{global} are passed to a differentiable MANO layer [7, 18] to get the hand vertices. These vertices are used to differentiably render a soft segmentation mask using SoftRasterizer [13, 17]. The grasp classifier head on θ_{local}, θ_{global} & β (predicted by WildHands) as input and is implemented as a 4-layer MLP (with 1024, 1024, 512, 128 channels and ReLU non-linearity after each). This MLP predicts logits for the 8 different grasp classes defined in [3].

Bounding box predictor: Since our model takes hand crops as input, we need to predict the bounding box of the hand in the image. On ARCTIC, we train a bounding box predictor on the ARCTIC training set by finetuning a MaskRCNN [9] model. For EPIC-HandKps, we use the recently released hand detector from [2]. During training, we use the ground truth bounding box for the hand crop (with small perturbation), estimated using the 2D keypoints and scaled by a fixed value of 1.5 to provide additional context around the hand. All the ablations use ground truth bounding box for the hand crop.

1.2. Training

We use different sources of supervision to train our model, depending on the dataset. We train WildHands jointly on multiple datasets: (1) Arctic [6]: using 3D supervision on β, θ_{local}, θ_{global}, 3D hand keypoints, 2D projections of 3D keypoints in the image and translation of root joint w.r.t. camera, (2) AssemblyHands [15]: using supervision on 3D hand keypoints and 2D projections of 3D keypoints in the image (it does not represent hands using MANO), and (3) EPIC-Kitchens [4]: using segmentation masks and grasp labels. Following [6], we use L2 loss for β, θ_{local}, θ_{global}, 3D keypoints and 2D keypoints. L1 loss is used for segmentation masks and cross-entropy for grasp lables. The loss weights used for each terms are: 5.0 for 2D keypoints, 5.0 for 3D keypoints, 10.0 for θ_{global}, 10.0 for θ_{local}, 0.001 for β, 1.0 for translation, 10.0 for segmentation and 0.1 for grasp loss.

We use the training sets of ARCTIC (187K images), AssemblyHands (360K) and VISOR split (30K) of EPIC-Kitchens to train our model. All models are trained for 100 epochs with a learning rate of $1e-5$. The multi-dataset training is done on 2 A40 GPUs with a batch size of 144 and Adam optimizer [12].

2. Ablations

Using predicted hand bounding box: As mentioned in Sec. 1.1, we finetune a MaskRCNN model to predict bounding box on ARCTIC and use hand predictor from [3] on EPIC-HandKps. When using predicted bounding box instead, we see a drop in performance on both ARCTIC and EPIC-HandKps (Tab. 1). This is expected since the predicted bounding box is not always accurate. We see similar trend for FrankMocap [19] as well.

Ignoring camera intrinsics: The KPE [16] encoding captures the location of the hand crop in the camera's field of view. However, camera intrinsics may not always be available in the wild. So, we also explore a variant of our model which ignores camera intrinsics and uses only the crop location w.r.t. to the image center (Tab. 1). We expect the intrinsics to matter more in multiple datasets setting involving images captured from different cameras, so we remove intrinsics from the model trained jointly on ARCTIC, AssemblyHands & EPIC-Kitchens. While we notice a slight dip in performance, it is significantly better than ArcticNet-SF [6] or FrankMocap [19].

Removing global features from grasp classifier: Our model uses a classifier to predict the grasp type from the estimated hand pose parameters. While hand pose is indicative of grasp type, hand poses might also occur in thin air, without actually grasping the object. So, we verify if using global image information could be useful in this case. From the results in Tab. 1, we see that removing global image features improves L2 error on EPIC-HandKps. This could be because the pseudo ground truth is generated using the predictions of a recent model [3] which might already be trained to distinguish between hand poses in thin air and grasping poses. However, we see a drop in performance on ARCTIC when removing global features. This is likely due to the model exploiting information from surrounding objects since all the 11 objects are scene during training.

Transformer variant: We modify the architecture of WildHands to use transformers [5, 20] instead of convolutions. This increases the capacity our the model by 4 times. We notice a significant improvement on ARCTIC (Tab. 1) but not on EPIC-HandKps. In fact, the performance decreases. This could be due to large capacity leading to overfitting on ARCTIC since varition in ARCTIC is limited. Another could be that strong supervision is not available on in-the-wild data used for training which could be causing the transformer to overfit to weird signals in the data. This needs more analysis to better understand the effect of transformer in our model.

Object loss in ArcticNet-SF: The default ArcticNet-SF [6] model predicts both the hand pose and the object pose from a single image. Since we focus only on hand pose, we also examine the effect of removing the object loss. We notice that MPJPE worsens but MRRPE improves. It is hard to identify the reason but could be due to the model overfitting to weird signals in objects since all 11 objects are seen during training. The performance on EPIC-HandKps decreases, which also indicates some extent of overfitting to ARCTIC objects.

Method	ARCTIC val		EPIC-HandKps
	MPJPE ↓	MRRPE ↓	L2 error ↓
WildHands	20.57	28.81	6.68
− GT hand box	21.02	30.31	7.28
− camera intrx	20.65	31.09	7.47
− segmentation	20.99	29.76	13.60
− segmentation − grasp-glb	21.90	29.87	12.41
WildHands (no aux.)	21.50	28.70	17.07
+ Transformer	19.50	27.11	20.36
ArcticNet-SF [6]	22.60	32.71	35.07
− no object loss	23.25	31.50	33.61
FrankMocap [19]	53.99	N/A	13.33
− GT hand box	57.00	N/A	14.57

Table 1. **Ablations**. We explore variants of WildHands which use predicted bounding box instead of ground truth bounding box (GT hand box), ignoring camera intrinsics & using only crop location w.r.t. to image center, removing global image features from the grasp classifier (grasp-glb) and using transformer [5, 20] instead of convolutional architecture. We also consider ArcticNet-SF [6] trained without object loss and FrankMocap [19] evaluation with predicted hand bounding box instead of GT hand box.

3. EPIC-HandKps

Since 3D hand annotations are difficult to collect for in-the-wild images, we instead collect 2D annotations for the 21 hand joints and use it to evaluate the 2D projections of the predicted 3D keypoints. We refer to this dataset as EPIC-HandKps. We sample 5K images from the validation set of VISOR split of EPIC-Kitchens and get the 21 joints annotated via Scale AI. We use the same joint convention as ARCTIC [6]. We crop the images around the hand using the segmentation masks in VISOR and provide the crops to annotators for labeling. Note that most of these images do not have all the 21 keypoints visible. Following ARCTIC, we only consider images with atleast 3 visible joints for evaluation. Moreover, since the models in our experiments required hand crops as input, we only evaluate on those images for which hand bounding box is predicted by the recently released hand detector model from [3]. This leaves us with 4724 hand annotations, with 2697 right hands and 2027 left hands. We show sample annotations in Fig. 1.

4. Visualizations

Qualitative comparisons: We provide visualizations of the predicted hand pose from WildHands and FrankMocap in Fig. 2. We show projection of the predicted hand in the image and rendering of the hand mesh from 2 more views. Our WildHands model is able to predict better hand poses from a single image than FrankMocap [19] in challenging occlusion scenarios involving dexterous interactions, *e.g.* stirring a ladle, grasping objects, rolling dough, pick & place, sprinkling toppings.

We provide similar visualizations on ARCTIC in Fig. 4 and compare with ArcticNet-SF [6]. Our WildHands model predicts better hand poses in scenes involving interaction with articulated objects.

Failure Cases: On EPIC-HandKps (Fig. 3), we observe that images in which the fingers are barely visible, *e.g.* when kneading a dough in top row, are quite challenging. Moreover, our model is sometimes unable to predict wide palm poses, *e.g.* grasps in bottom row. On ARCTIC (Fig. 5), we observe similar failure cases, *i.e.* when fingers are barely visible (top row), wide palm poses (bottom row).

5. Perspective Distortion Induced Ambiguity in Hands (Qualitative Visualizations)

Figure 6 reproduces Figures 3 and 4 from the main paper, but Figure 6 (b.1 and b.2) additionally provide qualitative visualizations for hands that are dissimilar in 3D shape and pose but similar in centered 2D hand keypoints, as found via a PnP alignment of 3D hands from the ARCTIC dataset to the reference 2D keypoints with appropriate shifts. Figure 7 and Figure 8 provide another 2 examples.

Figure 1. **EPIC-HandKps annotations**. We collect 2D joint annotations (shown in blue) for 5K in-the-wild egocentric images from EPIC-Kitchens [4] dataset. We show few annotations here. We also have the label for the joint corresponding to each keypoint. Note the large variation in dexterous poses of hands interactiong with objects.

Figure 2. **Visualizations on EPIC-HandKps**. Our WildHands model is able to predict better hand poses from a single image than FrankMocap [19] in challenging occlusion scenarios involving dexterous interactions, *e.g*. images of hands grasping the objects from EPIC-HandKps. We show projection of the predicted hand in the image and rendering of the hand mesh from 2 more views.

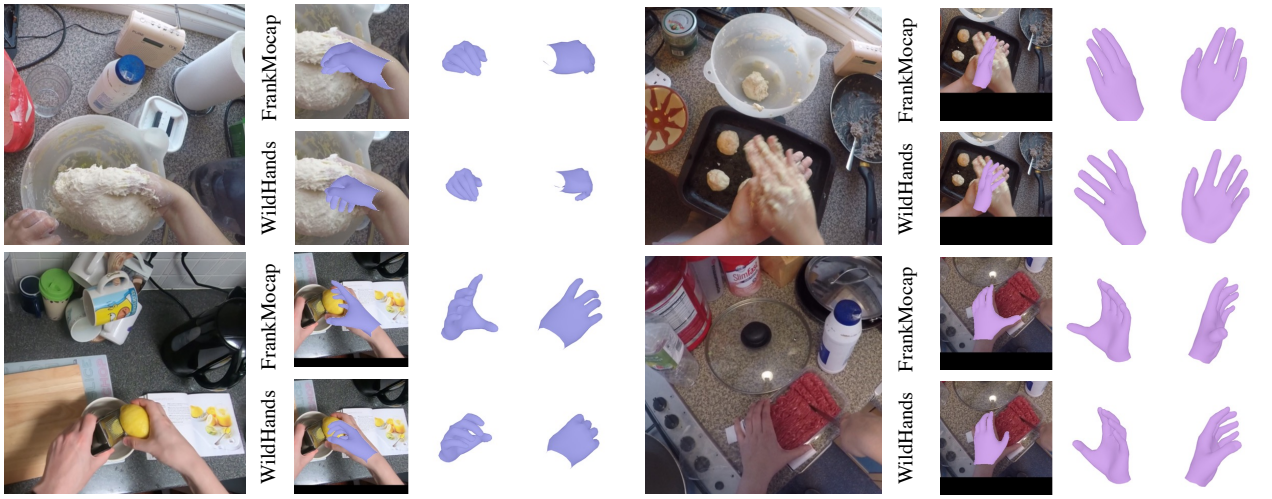

Figure 3. **Failure cases on EPIC-HandKps**. We observe that images in which the fingers are barely visible, *e.g*. when kneading a dough in top row, are quite challenging. Moreover, our model is sometimes unable to predict wide palm poses, *e.g*. grasps in bottom row.

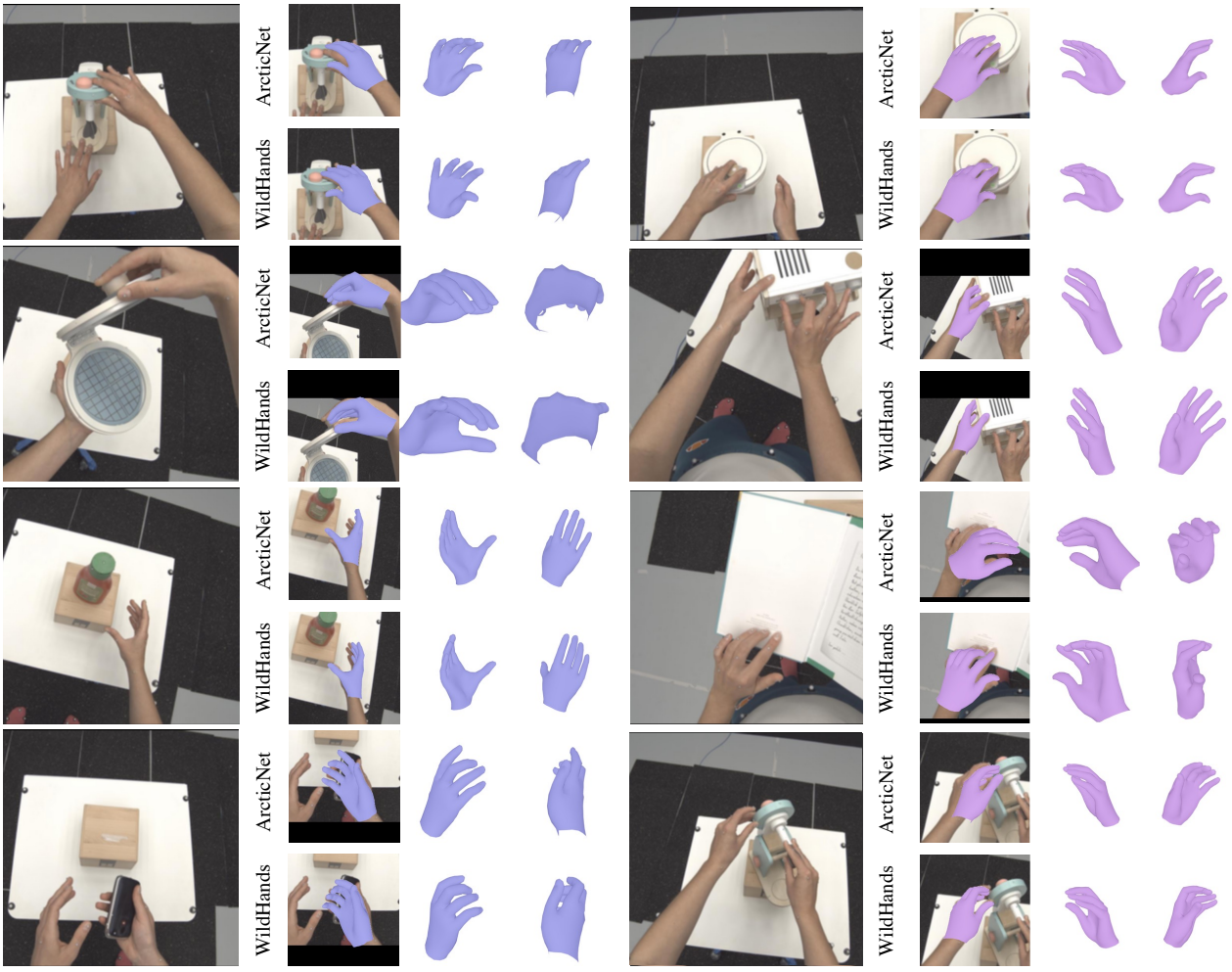

Figure 4. **Visualizations on ARCTIC**. Our WildHands model is able to predict better hand poses from a single image than ArcticNet-SF [6] in scenes involving interaction with articulated objects. We show projection of the predicted hand in the image and rendering of the hand mesh from 2 additional views.

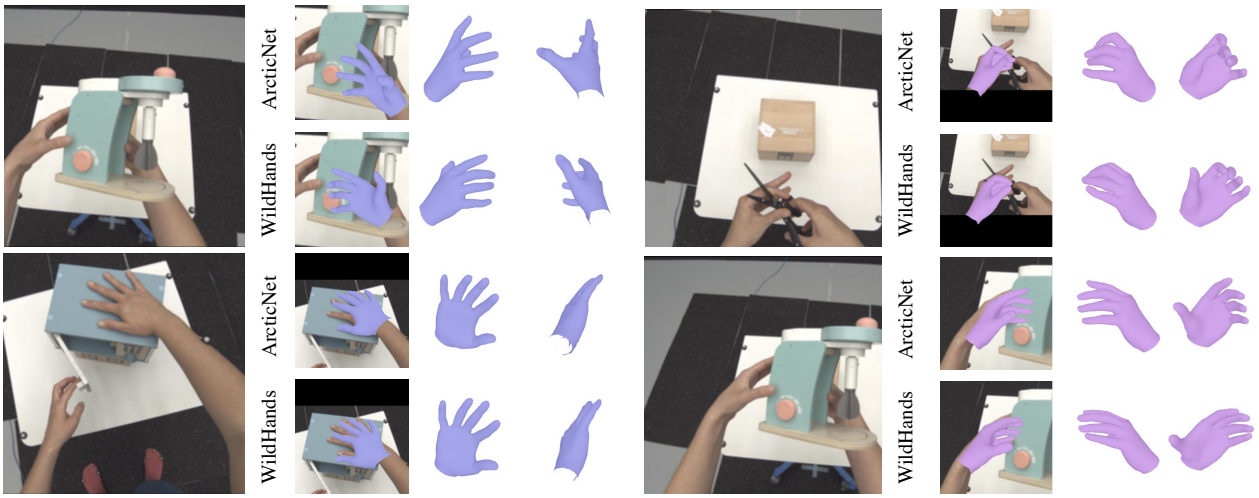

Figure 5. **Failure cases on ARCTIC**. We observe similar failure cases as Fig. 3, *i.e.* when fingers are barely visible (top row), wide palm poses (bottom row).

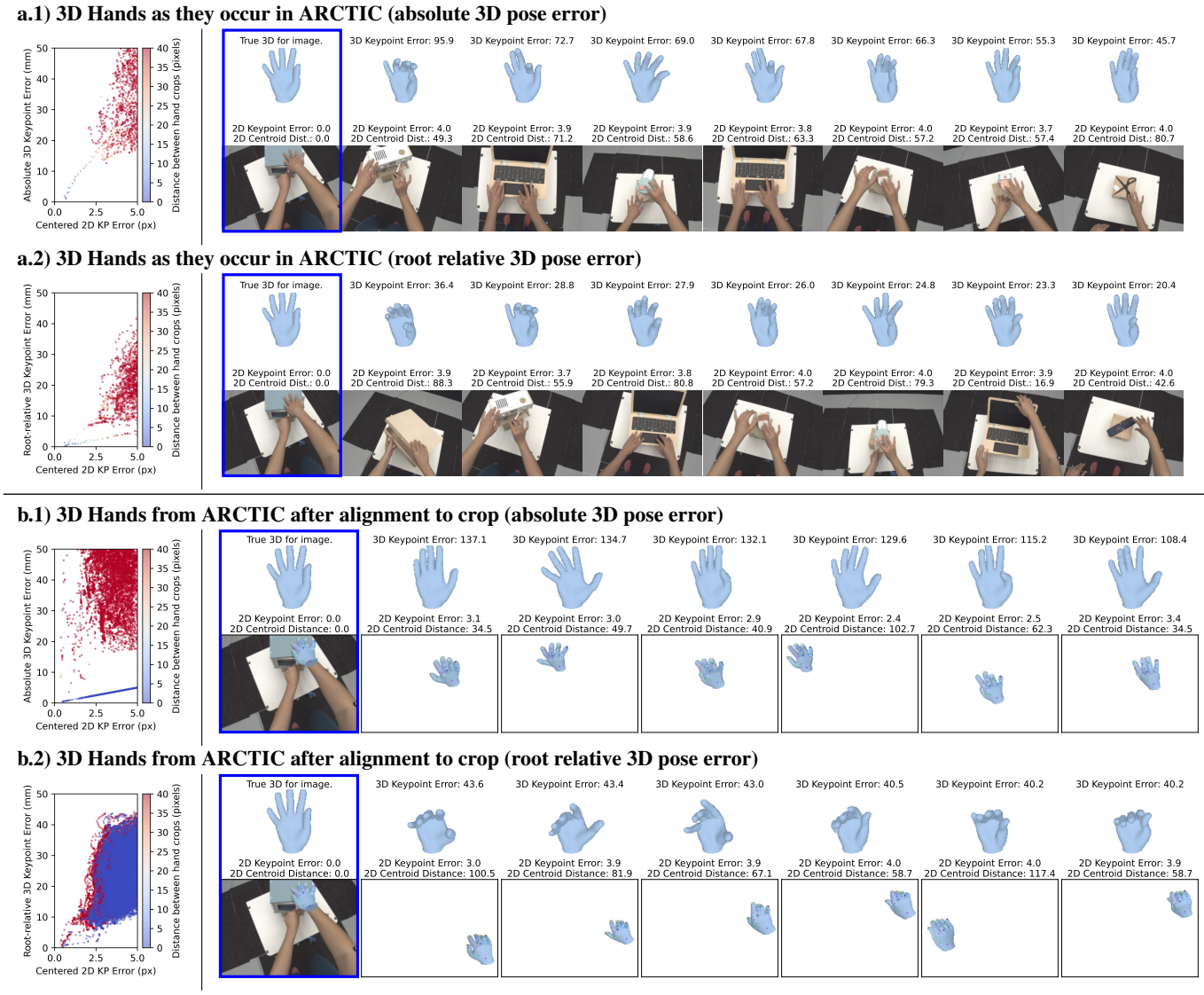

Figure 6. **Qualitative visualizations for hands with high 3D shape error (absolute in (a.1) and root-relative in (a.2)) but low 2D keypoint error (after centering).** The right hand in blue box is the reference hand w.r.t. which we measure 2D and 3D keypoint errors. **(b.1 and b.2)** show examples for the 3D hand shape and the corresponding location in the field of view such that the centered 2D projection of the hand keypoints match the reference in the blue box. Across all 4 settings, note the diversity in 3D hand shape and pose but the similarity in centered 2D hand pose. Scatter plots on the left are same as those in Fig. 3 in the main paper. Fig. 7 and Fig. 8 provide 2 more examples.

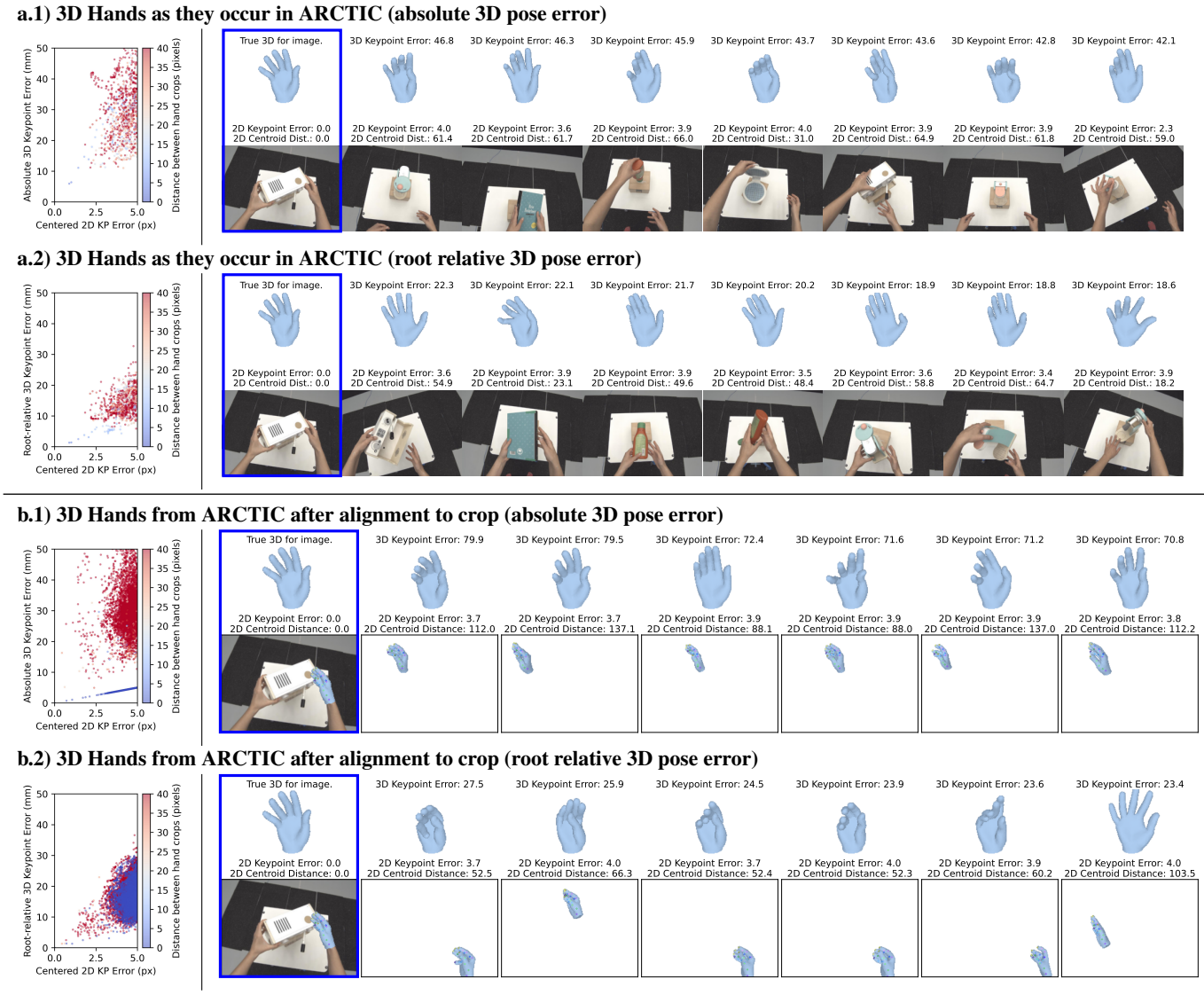

Figure 7. Another example similar to Fig. 6. **Qualitative visualizations for hands with high 3D shape error (absolute in (a.1) and root-relative in (a.2)) but low 2D keypoint error (after centering).** The right hand in blue box is the reference hand w.r.t. which we measure 2D and 3D keypoint errors. **(b.1 and b.2)** show examples for the 3D hand shape and the corresponding location in the field of view such that the centered 2D projection of the hand keypoints match the reference in the blue box. Across all 4 settings, note the diversity in 3D hand shape and pose but the similarity in centered 2D hand pose.

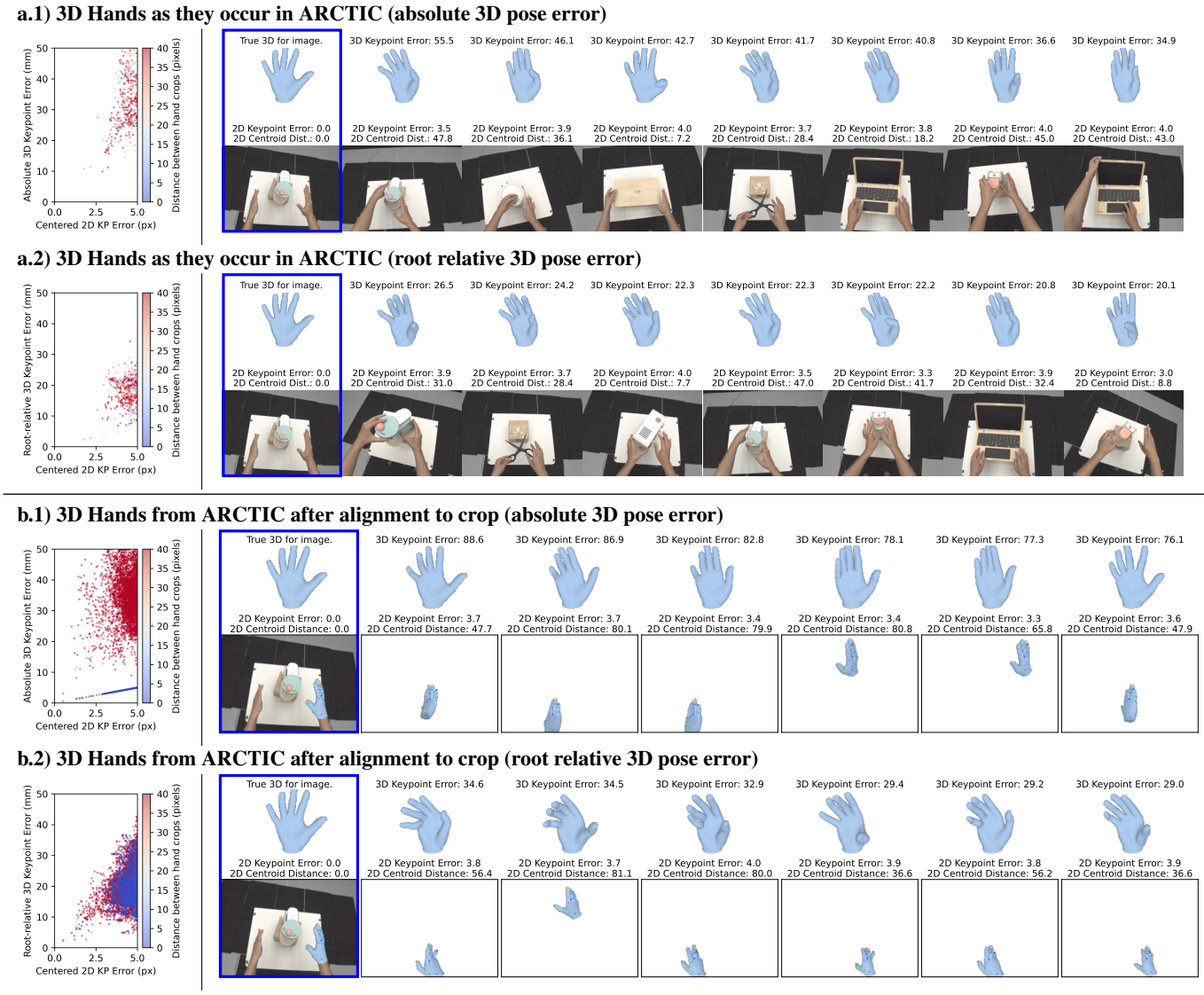

Figure 8. Another example similar to Fig. 6.**Qualitative visualizations for hands with high 3D shape error (absolute in (a.1) and root-relative in (a.2)) but low 2D keypoint error (after centering).** The right hand in blue box is the reference hand w.r.t. which we measure 2D and 3D keypoint errors. **(b.1 and b.2)** show examples for the 3D hand shape and the corresponding location in the field of view such that the centered 2D projection of the hand keypoints match the reference in the blue box. Across all 4 settings, note the diversity in 3D hand shape and pose but the similarity in centered 2D hand pose.

References

[1] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv*, 1803.08375, 2018. 1

[2] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[3] Tianyi Cheng, Dandan Shan, Ayda Sultan Hassen, Richard Ely Locke Higgins, and David Fouhey. Towards a richer 2d understanding of hands at scale. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 1, 2, 3

[4] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 2, 4

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 2, 3

[6] Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 3, 6

[7] Yana Hasson, Gül Varol, Dimitris Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 2

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2015. 1

[11] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[12] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015. 2

[13] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. A general differentiable mesh renderer for image-based 3d reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 1

[14] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010. 1

[15] Takehiko Ohkawa, Kun He, Fadime Sener, Tomas Hodan, Luan Tran, and Cem Keskin. Assemblyhands: Towards egocentric activity understanding via 3d hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12999–13008, 2023. 2

[16] Aditya Prakash, Arjun Gupta, and Saurabh Gupta. Mitigating perspective distortion-induced shape ambiguity in image crops. *arXiv*, 2023. 1, 2

[17] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020. 1

[18] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (ToG)*, 2017. 1

[19] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. Frankmocap: Fast monocular 3D hand and body motion capture by regression and integration. *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2021. 1, 2, 3, 5

[20] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2, 3