Look Ma, No Hands!
Agent-Environment Factorization of Egocentric Videos
Supplementary Material

Anonymous Author(s)
Affiliation
Address
email

S1 Video

The accompanying video (vidm.mp4) in MP4 (AAC, H.264) format provides a narrated overview of the method and shows example predictions visualizing our factorized representation on in-the-wild videos. The video was tested to play well in Google Chrome, VLC, and QuickTime.

S2 Code

Additionally, the supplementary material contains two files "model.py" and "config.yaml". These contain the code for our video inpainting diffusion model, and the parameters used in its instantiation respectively.

S3 VIDM Training Details

For an overview of the VIDM model architecture, see Main Paper Section 4. In each block, the CNN layers are implemented as residual blocks [9] with SiLU non-linearities [10] and each attention layer does self-attention across all token from all input images using 32-channel GroupNorm normalization. Following [14], upsampling and downsampling operations are both implemented using residual CNN blocks with either an internal nearest mode $2\times$ upsampling operation or internal $2\times$ downsampling via average pooling. An initial convolution brings the feature dimension to 256, which is raised to a maximum of 1024 at the center of the U-Net. At the highest spatial resolution of 64×64 the self-attention layer is omitted, as attention with 16384 ($= 64 \times 64 \times 4$) tokens is computationally intractable for our available hardware. The largest attention layer occurs at a spatial resolution of 32×32 across four images for a total of 4096 tokens.

We trained VIDM using target images from Ego4D [6] and VISOR [4] (see Main Paper Sec. 4). Since no evaluation was done on Ego4D, no Ego4D data was held out. For VISOR, all data from participants *P37, P35, P29, P05*, and *P07* was held-out from training. This held-out data from these participants was used for reconstruction quality evaluation (Main Paper Section 5.1) and object detection (Main Paper Section 5.2) experiments. Table S1 lists hyper-parameters. Figure S2 shows sample training batches.

S4 Downstream Task Experimental Details

S4.1 Detection

We used off-the-shelf Mask R-CNN R_101_FPN_3x from Detectron2 [8, 17] trained on the COCO dataset [11] for evaluation. We used overlapping classes between the VISOR [4] annotations and

Table S1: VIDM Model and Training Hyper-parameters.

Hyper-parameter	**Value**
Learning Rate	4.8×10^{-5}
Batch Size	48
Optimizer	Adam
Diffusion Steps (training)	1000
Latent image Size	64×64
Number of VQ Embedding Tokens	8192
VQ Embedding Dimension	3
Diffusion Steps (inference)	200
Attention Heads	8

COCO for evaluation. These were: *apple, banana, bottle, bowl, broccoli, cake, carrot, chair, cup, fork, knife, microwave, oven, pizza, refrigerator, sandwich, scissors, sink, spoon, toaster*.

S4.2 Affordance Prediction

Dataset: We experiment on EPIC-ROI and GAO tasks from Goyal *et al.* [5]. EPIC-ROI uses the EPIC-KITCHENS dataset [3] and GAO uses YCB-Affordance [2] dataset. We consider a low data regime in our work and sample $1K$ images from these datasets to train the different models. For EPIC-ROI, we sample images with a probability inversely proportional to the length of the video. For GAO, we sample randomly. We use the same evaluation setting from [5].

Model: We use the same architecture from ACP [5] and replace the EPIC-ROI input images with images produced by our inpainting model (with hands removed) to incorporate our factorized representation. While ACP [5] masks out a patch at the bottom center of the image to hide the hand, we do not need any mask (neither for training nor for testing) since the hands have been removed via inpainting. The input is processed by ResNet-50 followed by different decoders for EPIC-ROI and GAO tasks.

Training: We train separate models for EPIC-ROI and GAO using the loss function and hyperparameters from ACP [5]. While it is possible to train a single model in multitask manner, we observe that the two tasks are not complementary to each other. We train using 3 seeds for each task and report the mean and standard deviation in the metrics.

S4.3 3D Reconstruction of Hand-held Objects

Dataset: We use ObMan [7] dataset which consists of $2.5K$ synthetic objects from ShapeNet [1]. We use the train and test splits provided by Ye *et al.* [18]. We divide the train split into train and val set. The train set consists of $134K$, val set $7K$ and test set $6.2K$ images. The dataset provides 3D CAD models for each object, which we use for training hand-held object reconstruction model from Ye *et al.* [18].

Model: We use the architecture from Ye *et al.* [18]. It uses FrankMocap [16] to extract hand articulation features from a single image using MANO [15] hand parameterization. These hand features are used as conditioning to a DeepSDF [12] model which predicts the object shape using implicit representation. This model also takes in pixel-aligned features and global image features along with hand features. To incorporate our factorized representation, we also extract global image features and pixel-aligned features from ObMan images showing only objects (with hands removed). These features are concatenated with the features from the input ObMan images and fed as input to the DeepSDF [12] decoder.

Training: Following [18], we use a normalized hand coordinate frame for sampling points and predicting SDFs. We sample 8192 points in $[-1, 1]^3$ for training, out of which half of them lie inside and the rest lie outside the object. At test time, 64^3 points are sampled uniformly in $[-1, 1]^3$. We

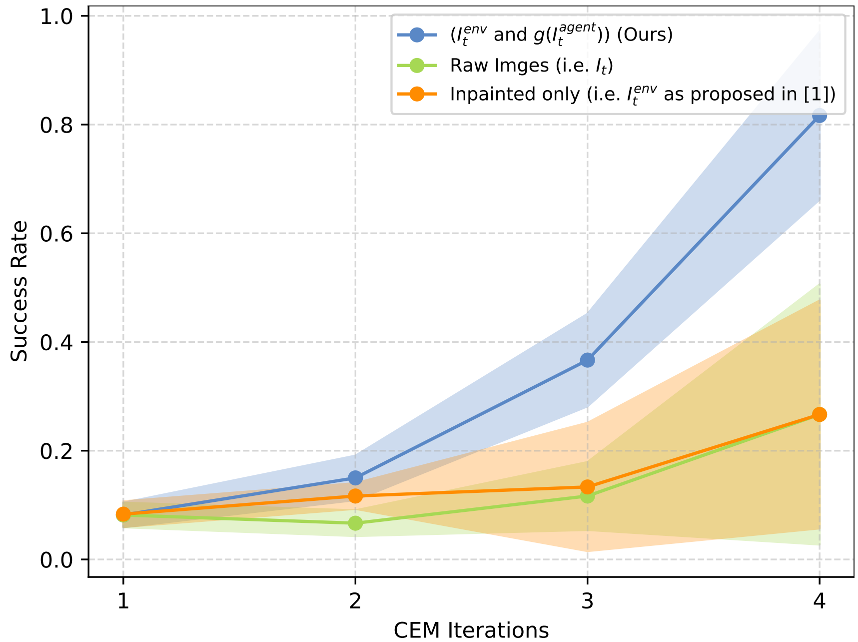


Figure S1: Success rate as a function of CEM iterations for the real-world experiment described in Main Paper Section 5.6. We report the mean and standard deviation across 3 runs for each method.

train the model in a supervised manner using 3D ground truth from ObMan [7] for 200 epochs with a learning rate of $1e-5$. Other hyper-parameters are used directly from [18].

S4.4 Error Bars for Real-World Policy Learning using Learned Rewards

In Figure S1, we report error bars for the real-world experiment (Main Paper Section 5.6) across additional runs. Across 3 runs for each method, we see that our method clearly performs the best (final mean success rate of 82% vs 27% for both baselines).

S5 Visualizations

In Figure S2, we include a visualization of a training batch for our method, showcasing supervision and generated masks. In Figure S3, we include additional visualizations of the predictions made by our method and baselines.

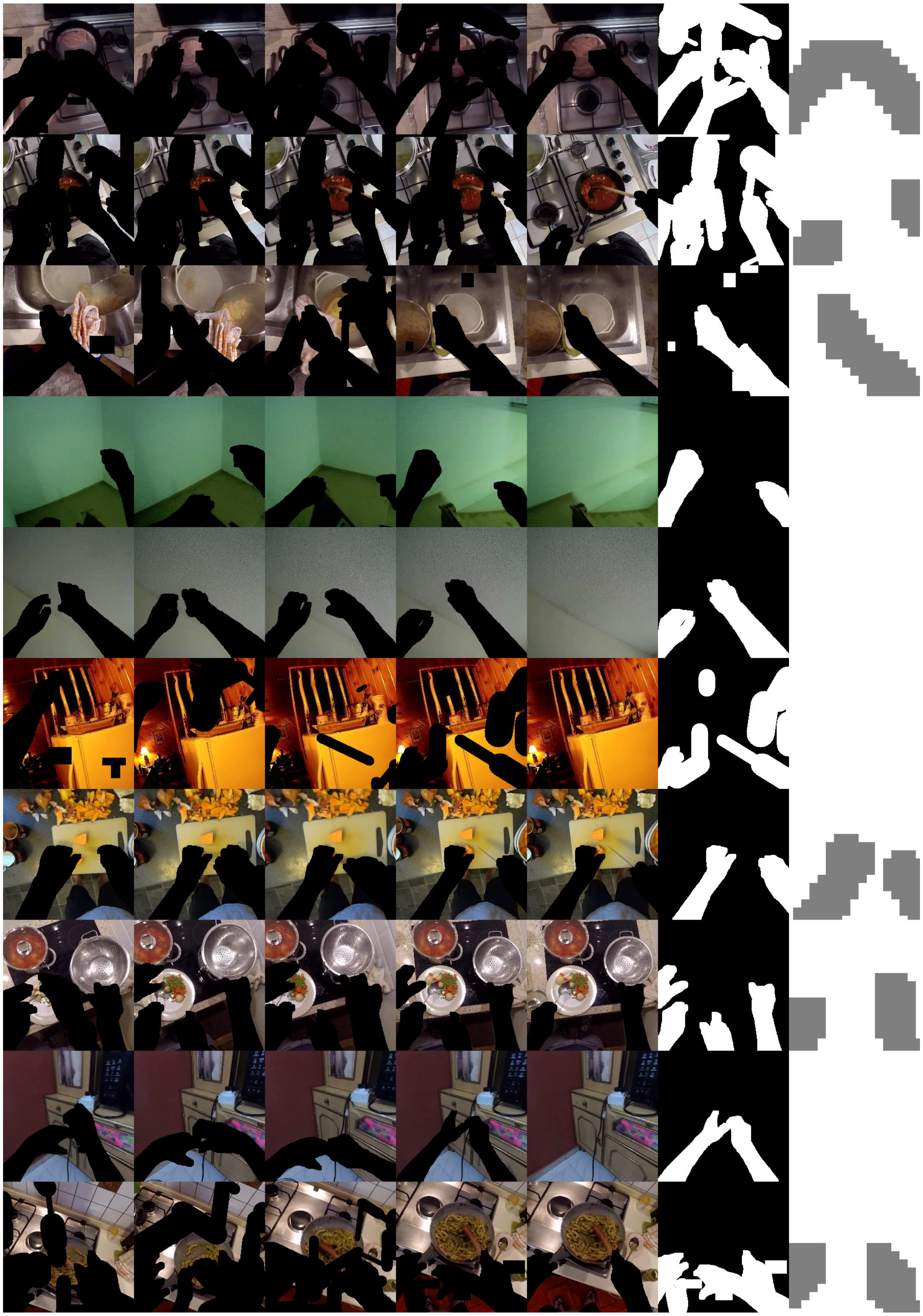
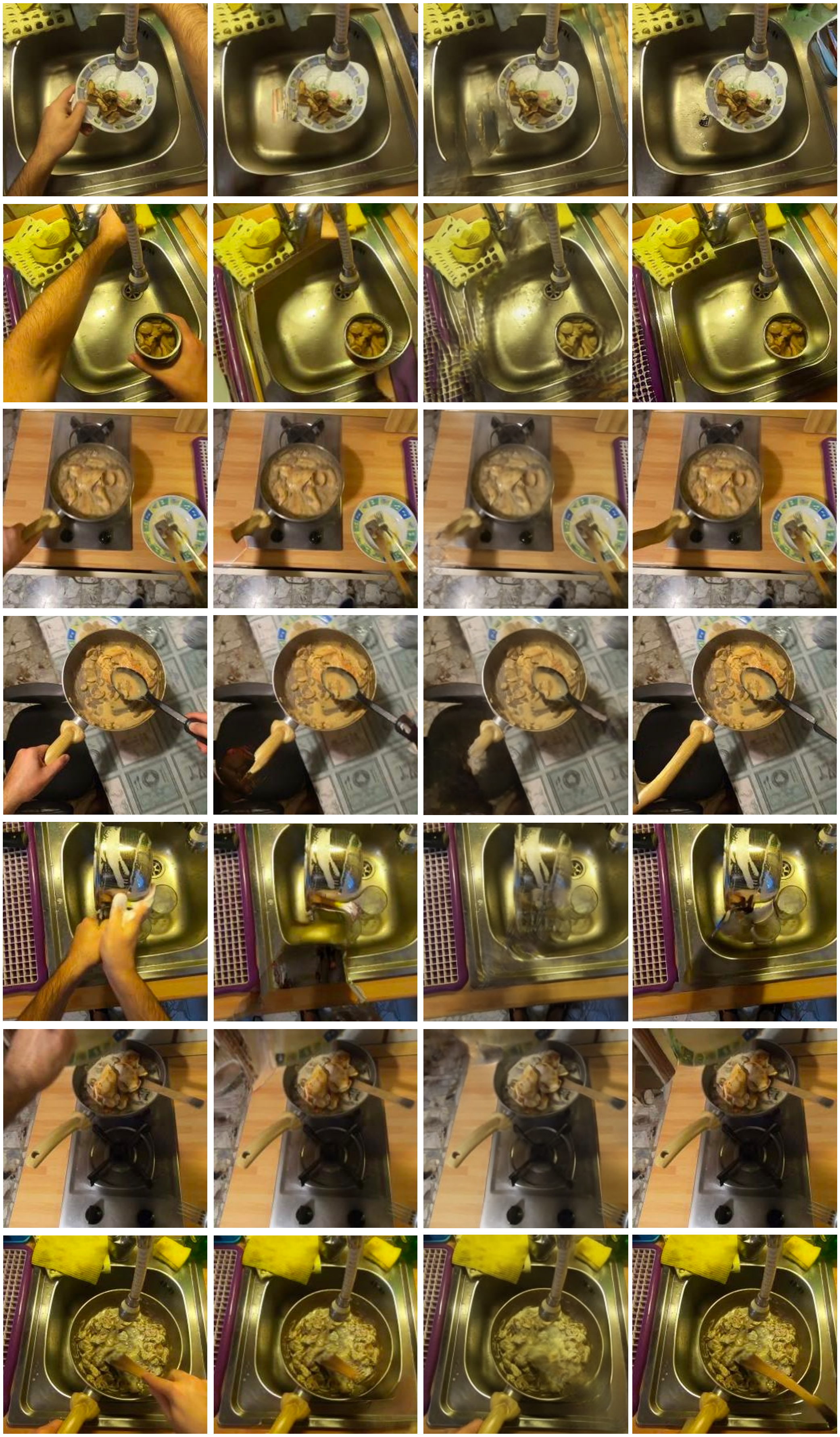


Figure S2: An example batch for training VIDM. Columns 1-4: Input images to the network. Column 5: target image for reconstruction. Column 6: Masked regions on the target image. Column 7: Pixels with loss propagated (white pixels have loss, gray pixels have no loss). Note that hands that are masked in the target image (column 5) have no loss on them. See Main Paper Section 4 for details.



a) Original Image b) LatentDiffusion FT [14] c) DLFormer [13] d) VIDM (Ours)

Figure S3: Additional visualizations of predictions from our method and baselines.

References

[1] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *arXiv*, 1512.03012, 2015. 2

[2] Enric Corona, Albert Pumarola, Guillem Alenyà, Francesc Moreno-Noguer, and Grégory Rogez. Ganhand: Predicting human grasp affordances in multi-object scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[3] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, , Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55, 2022. 2

[4] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Higgins, Sanja Fidler, David Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. In *Proceedings of the Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, 2022. 1

[5] Mohit Goyal, Sahil Modi, Rishabh Goyal, and Saurabh Gupta. Human hands as probes for interactive object understanding. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[6] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the World in 3,000 Hours of Egocentric Video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[7] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11807–11816, 2019. 2, 3

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[9] 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*, pages 770–778, 2016. 1

[10] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016. 1

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755. Springer, 2014. 1

[12] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[13] Jingjing Ren, Qingqing Zheng, Yuanyuan Zhao, Xuemiao Xu, and Chen Li. Dlformer: Discrete latent transformer for video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3511–3520, 2022. 5

[14] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 5

[15] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (ToG)*, 2017. 2

[16] 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. 2

[17] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 1

[18] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3d reconstruction of generic objects in hands. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3