3D Hand Shape and Pose Estimation from a Single RGB Image (Supplementary Material)

Liuhao Ge[1]*, Zhou Ren[2], Yuncheng Li[3], Zehao Xue[3], Yingying Wang[3], Jianfei Cai[1], Junsong Yuan[4]
[1]Nanyang Technological University [2]Wormpex AI Research
[3]Snap Inc. [4]State University of New York at Buffalo
ge0001ao@e.ntu.edu.sg, zhou.ren@bianlifeng.com, yuncheng.li@snap.com,
zehao.xue@snap.com, ywang@snap.com, asjfcai@ntu.edu.sg, jsyuan@buffalo.edu

1. Qualitative Results

We present more qualitative results of 3D hand mesh reconstruction and 3D hand pose estimation for our synthetic dataset, our real-world dataset, STB dataset [3], RHD dataset [4], and Dexter+Object dataset [2], as shown in Fig. 1.

2. Details of Baseline Methods for 3D Hand Mesh Reconstruction

In Section 5.3 of our main paper, we compare our proposed method with two baseline methods for 3D hand mesh reconstruction: direct Linear Blend Skinning (LBS) method and MANO-based method. Here, we describe more details of these two baseline methods, as illustrated in Fig. 2.

In the direct LBS method, we train the network to regress 3D hand joint locations from the heat-maps and the image features with heat-map loss and 3D pose loss. As illustrated in Fig. 2 (b), the latent feature extracted from the input image is mapped to 3D hand joint locations through a multi-layer perceptron (MLP) network with three fully-connected layers. Then, we apply inverse kinematics (IK) to compute the transformation matrix of each hand joint from the the estimated 3D hand joint locations. The 3D hand mesh is generated by applying LBS with the predefined hand model and skinning weights. In this method, the 3D hand mesh is only determined by the estimated 3D hand joint locations, thus it cannot be adapted to various hand shapes. In addition, the IK often suffers from singularity and multiple solutions, which makes the solutions to transformation matrices unreliable. Experimental results in Figure 7 and Table 2 of our main paper have shown the limitations of this direct LBS method.

In the MANO-based method, we train the network to regress hand shape and pose parameters of the MANO hand model [1]. As illustrated in Fig. 2 (c), the latent feature extracted from the input image is mapped to hand shape and pose parameters θ, β through an MLP network with three fully-connected layers. Then, the 3D hand mesh is generated from the regressed parameters θ, β using the MANO hand model [1]. Note that the MANO mesh generation module is differentiable and is involved in the network training. The networks are trained with heat-map loss, mesh loss and 3D pose loss, which are the same as our method. Since the MANO hand model is fixed during training and is essentially LBS with blend shapes [1], the representation power of this method is limited. Experimental results in Figure 7 and Table 2 of our main paper have shown the limitations of this MANO-based method.

3. Details of the Task Transfer Method

In Section 5.4 of our main paper, we implement an alternative method ("full model, task transfer") for 3D hand pose estimation by transferring our full model trained for 3D hand mesh reconstruction to the task of 3D hand pose estimation. Here, we describe more details of our task transfer method. As illustrated in Fig. 3, we directly regress the 3D hand joint locations from the latent feature extracted by our full model using an MLP network with three fully-connected layers. We first train the MLP network with 3D pose loss on our synthetic dataset. When experimenting on STB dataset [3] with 3D pose supervision, we fine-tune the MLP network with 3D pose loss. When experimenting on STB dataset [3] without 3D pose supervision, we directly use the MLP network pretrained on our synthetic dataset. Experimental results in Figure 8 of our main paper show that our task transfer method is better than the baseline method which is only trained for 3D hand pose estimation, even though these two methods have the same pipeline. This indicates that the latent feature extracted by our full model is more discriminative and is easier to regress accurate 3D hand pose since our full model is trained with the

*This work was done when Liuhao Ge was a research intern at Snap Inc.

dense supervision of the 3D hand mesh that contains richer information than the 3D hand pose. In addition, although the estimation accuracy of our task transfer method is a little bit worse than that of our full model, our task transfer method is faster than our full model, since it does not generate 3D hand mesh. The runtime of our task transfer method is 15.1ms, while the runtime of our full model which estimate 3D hand pose from hand mesh is 19.9ms. Thus, in applications that only require 3D hand pose estimation but not 3D hand shape estimation, we can choose to use this task transfer method, which can maintain a comparable accuracy as our full model while runs at faster speed.

References

[1] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (TOG)*, 36(6):245, 2017.

[2] Srinath Sridhar, Franziska Mueller, Michael Zollhöfer, Dan Casas, Antti Oulasvirta, and Christian Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *ECCV*, 2016.

[3] Jiawei Zhang, Jianbo Jiao, Mingliang Chen, Liangqiong Qu, Xiaobin Xu, and Qingxiong Yang. 3D hand pose tracking and estimation using stereo matching. *arXiv preprint arXiv:1610.07214*, 2016.

[4] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single RGB images. In *ICCV*, 2017.

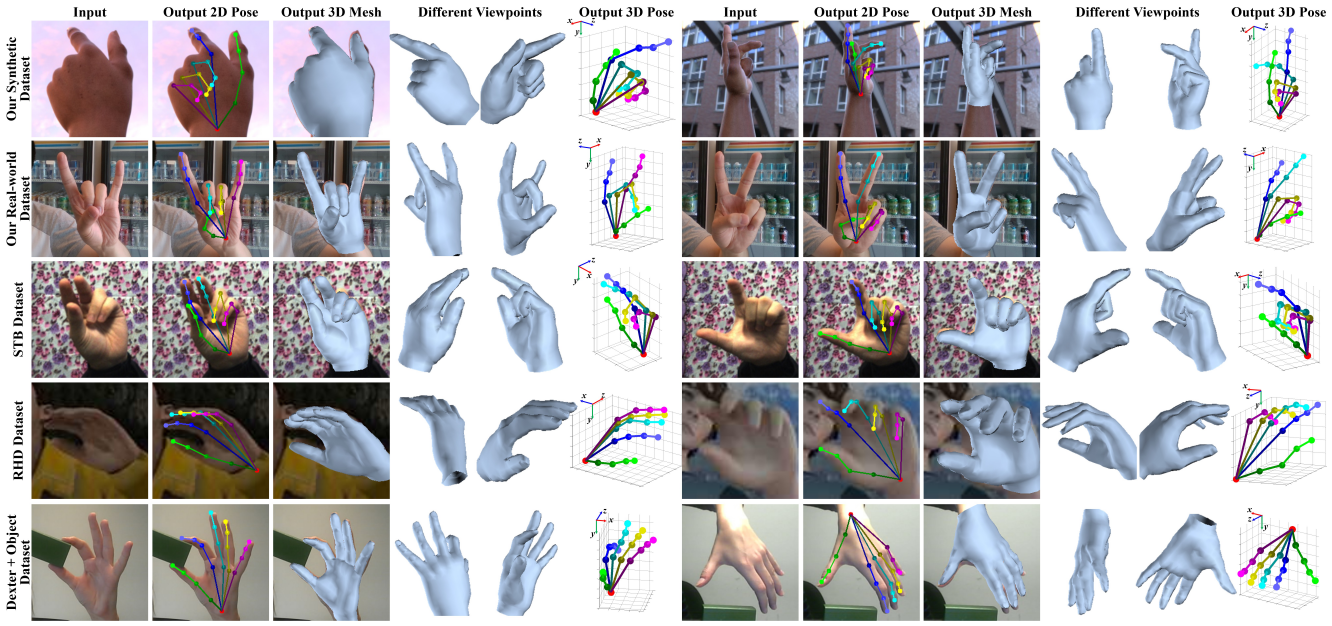

Figure 1: Qualitative results for our synthetic dataset (the first row), our real-world dataset (the second row), STB dataset [3] (the third row), RHD dataset [4] (the fourth row), and Dexter+Object dataset [2] (the last row).

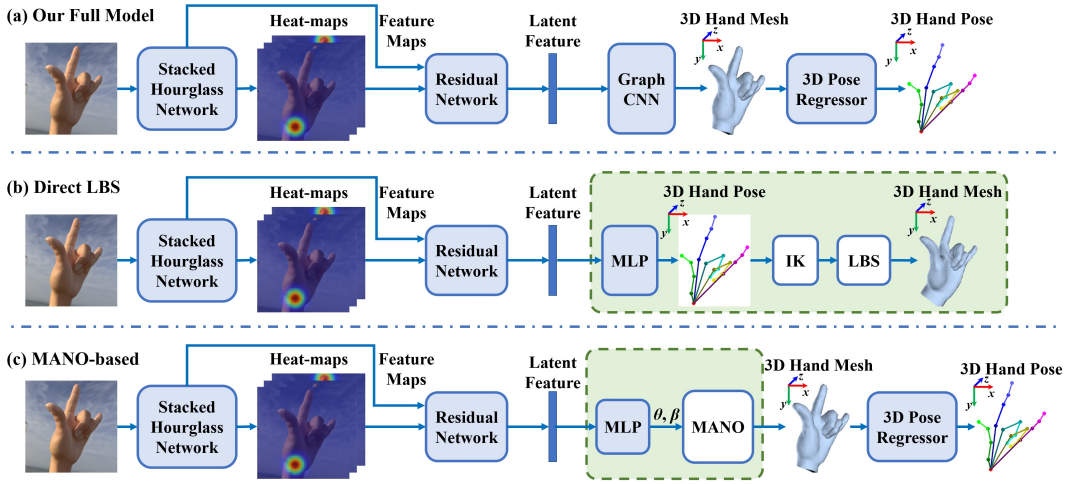

Figure 2: Pipelines of our proposed method and two baseline methods: direct LBS method and MANO-based method. The differences between the two baseline methods and our proposed method are highlighted in the green dashed line box.

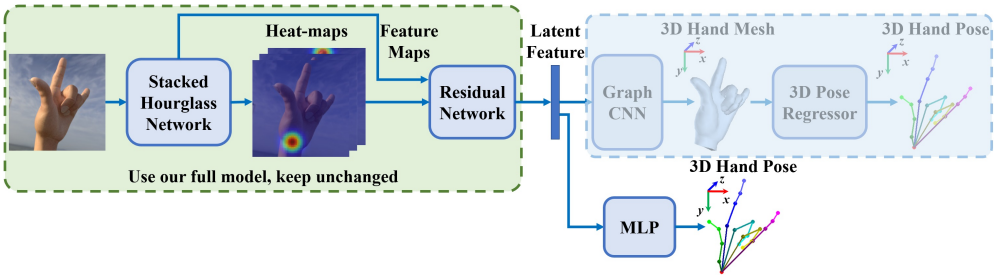

Figure 3: Illustration of our "full model, task transfer" method. We transfer our full model trained for 3D hand mesh reconstruction to the task of 3D hand pose estimation. Note that when training for the task of 3D hand pose estimation, the stacked hourglass network and the residual network are keep unchanged with our full model which is fully trained for the task of 3D hand mesh reconstruction.