3D Hand Pose Estimation from Single RGB Images Using Prior-knowledge and Mesh Supervision

Digang Sun
School of Computer Science and Engineering
South China University of Technology
Guangzhou, China 510006
`cssundg@mail.scut.edu.cn`

Ping Zhang
School of Computer Science and Engineering
South China University of Technology
Guangzhou, China 510006
`pzhang@scut.edu.cn`

Abstract

3D hand pose estimation from single RGB images is challenging because self-occlusions and the absence of depth make it difficult to estimate relative depth between hand joints and to produce biomechanically feasible hand poses. To address these issues, this paper proposes a Prior-knowledge and Mesh Supervision Network (PMSNet) to effectively combine the prior knowledge implied in the rigid articulated structure of the hand and the information contained in the hand mesh annotation. In our method, prior knowledge is forged into two categories. Implicit priors are learned from data and used to infer relative depth from 2D hand poses and hand textures. More importantly, we explicitly extract 2D hand poses and hand textures separately and concatenating them together. Explicit priors, including biomechanical constraints on an individual finger and multiple fingers, are embedded in a set of elaborately designed loss functions. Furthermore, hand meshes providing information in a higher order of magnitude than hand poses are employed as a supervision mechanism to fine-tune the network and it can be removed in inference stage so that time and space complexities of the network can be reduced. Experimental results, both quantitatively and qualitatively, show that our method is superior to start-of-the-arts with a considerable margin in terms of accuracy and generalization ability.

1 Introduction

3D hand pose estimation from single RGB images is more challenging than from depth ones due to lack of depth information. To address the problem of depth ambiguity, some approaches resort to images taken from multi-view [17, 21]. Some other methods take 3D hand pose estimation as a by-product of the hand shape reconstruction task [1, 29, 8]. In addition, to prevent from producing infeasible 3D hand pose as much as possible, some geometric or biomechanical constraints are proposed [28, 19].

The aforementioned methods have significantly improved the performance of 3D hand pose estimation. However, a multi-view sensing system needs multiple cameras placed in different angles, which could damage the convenience and naturalness in some applications, e.g., human-machine interaction. Deriving 3D hand poses from hand shape reconstruction tasks means we have to implement and use it during both training and inference stages, which might be heavy for hand pose estimation. On the other hand, the knowledge implied in the articulated structure of the hand, which could be beneficial to producing more accurate and feasible 3D hand poses, is still worth further explorations. Moreover, there still is some room for integrating various kinds of prior knowledge in a effective and efficient way.

Preprint. Under review.

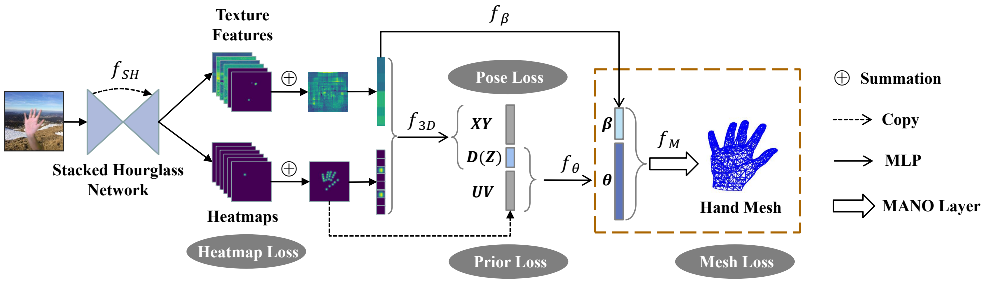


Figure 1: Architecture of the proposed Prior-knowledge and Mesh Supervision Network (PMSNet). It mainly comprises a stacked hourglass network as the backbone, a relative depth regressor, and a MANO layer (in the dotted-line bounding box). The MANO layer will be removed when doing inference.

In this paper, we propose a novel Prior-knowledge and Mesh Supervision Network (PMSNet) and design a set of loss functions to embed prior knowledge so that we can train the network in an end-to-end fashion. We observed that it is not sufficient to regress relative depth between hand joints using only 2D coordinates of them, since there are multiple 3D hand poses corresponding to one 2D pose (see Fig. 2). We also observed that the hand texture could contribute to performance gains. Therefore, we argue that to boost the accuracy of 3D hand pose estimation, it is necessary to combine 2D hand poses and hand textures together. As we know, the hand containing bones and joints has a rigid articulated structure. Consequently, we can, to some degree, infer the relative depth between hand joins via the rule that smaller distance of two adjacent joints in the image plane means larger difference in depth. As for prior knowledge, although some geometric and biomechanical constraints have been proposed, more detailed researches are still needed so as to make them be complementary to other components. Furthermore, hand meshes, which are used in hand shape reconstruction tasks, can also be employed as a higher level supervision for hand pose estimation to get better results. It's worth noting that this mechanism can be removed when the network put into practice so that the cost of storage space and inference time can be reduced.

To better infer relative depth between hand joints, we explicitly extract 2D hand poses and hand textures separately and fuse them with concatenation operation. To the best of our knowledge, we are the first to adopt this strategy while conventional methods usually use convolutional layers to transform intermediate features to an ordinary vector and feed it into fully-connected layers. The architecture of our proposed PMSNet is shown in Fig. 1. It mainly consists of three parts: (i) a stacked hourglass network [15] as the backbone to extract multi-scale features and fuse them in a multi-stage manner; (ii) a regressor that infers relative depth between hand joints from 2D hand poses and hand textures; (iii) a differentiable MANO [16] layer [7] that generates hand meshes according to the MANO model.

We test our method on the STB [27], RHD [31], and FreiHAND [32] datasets, which are commonly used in 3D hand pose estimation. We also test our method on a custom hand gesture dataset for cross-dataset evaluation. We compare our method with other state-of-the-arts in terms of evaluation accuracy and generalization ability. Experimental results show that our method outperforms state-of-the-arts with a considerable margin. For example, the area under the 3D PCK curve of our method on the STB dataset is 0.933 while that of method [5] is 0.791.

2 Related work

Hand pose estimation from RGB images 3D Hand pose estimation from RGB images especially a single one is challenging due to the absence of depth information. With the emergence and advancement of deep learning which is capable of automatically extracting features from training data, numerous methods [31, 2, 26, 18, 20, 23, 30] use deep neural networks to extract features from single RGB images of the hand. Zimmermann and Brox [31] learn implicit priors from training data. The network consists of three parts, which are used to segment hand, extract 2D coordinates and derive 3D hand pose, respectively. Spurr et al. [18] construct a unified latent space using multiple modalities to encourage similar poses with different modality to be embedded close to each other. Iqbal et al.

[9] propose 2.5D heatmaps which consists of 2D heat-maps and a depth map for each key-point for depth prediction. Mueller et al. [14] use a geometrically consistent translation network to generate hand images that follow the same statistical distribution as real-world images. A straightforward way to mitigate the issue of depth ambiguity is using images taken from different views [17, 21]. Sridhar et al. [21] employ multi-view images to match a pose while Simon et al. [17] use the image where joint detection is easy to improve joint detectors that work on difficult images.

Hand shape estimation from RGB images Most of the hand shape estimation methods [1, 29, 8, 12, 3] are based on the MANO hand model [16], which can generate hand meshes from the shape and pose parameters. Boukhayma et al. [1] present an end-to-end method for hand shape and pose estimation from single color images in the wild. Zhang et al. [29] use a multi-task learning framework to estimate the 2D/3D hand pose, hand mask, and MANO hand mesh. Chen et al. [3] propose I2UV-HandNet that takes a low-resolution UV position map which is derived from an unfolded MANO hand mesh as input and output a high-resolution one. Many recent works [8, 12] focus on more challenging scenarios related to hand-object interaction. Different from these methods based on MANO, Ge et al. [5] propose an end-to-end trainable hand mesh generation approach using Graph CNN [4].

Hand shape and/or pose estimation using geometric and biomechanical constraints To produce as feasible estimation result as possible, some works [28, 19, 22] adopt geometric or biomechanical limits. Zhang et al. [28] introduce two geometric rules to restrict the position of the joints in an individual finger. In [22], priors consist of the lower bound of the variational auto-encoder (VAE) and the bone lengths as well as self-occlusions of the hand, while Spurr et al. [19] adopt palmar structure, bone lengths, and joint angles as biomechanical constraints. In contrast, Zimmermann and Brox [31] utilize deep neural networks to learn implicit priors from data rather than explicitly defining constraints.

3 Method

3.1 Overview

Given an input RGB image I, the purpose of 3D hand pose estimation is to extract 3D coordinates of hand joints. However, this type of task generally suffers from the absence of depth information. How to accurately infer relative depth between hand joints and lead to biomechanically feasible 3D poses is a core issue.

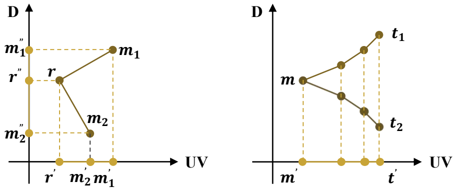


Figure 2: Visualization of prior geometric knowledge implied in the articulated structure of the hand. (left) For the same bone, a smaller length in the image plane generally means a larger depth margin between joints. (right) For the same finger, there exist at least two 3D poses corresponding to one 2D pose.

As we know, the bones and joints of the human hand form a rigid articulated structure, of which the priors could provide opportunities for us to improve the evaluation result. Fig. 2 presents a visualization of prior geometric knowledge implied in the hand. Taking the bone between the wrist and the MCP of the middle finger denoted as $\overrightarrow{rm}$ as an example, we can infer that a smaller value of $\|\overrightarrow{rm}\|$ in the image plane (UV) commonly means a larger margin in depth. Specifically, in the left of Fig. 2, the bone $\overrightarrow{rm}$ is presented in two different orientations as $\overrightarrow{rm_1}$ and $\overrightarrow{rm_2}$, i.e., $\|\overrightarrow{rm_1}\| = \|\overrightarrow{rm_2}\|$. $\overrightarrow{r'm'_1}$ and $\overrightarrow{r'm'_2}$ are their projections on the image plane and $\overrightarrow{r''m''_1}$ and $\overrightarrow{r''m''_2}$ are their projections on the depth axis. It can be observed that $\|\overrightarrow{r'm'_1}\| > \|\overrightarrow{r'm'_2}\|$ while $\|\overrightarrow{r''m''_1}\| < \|\overrightarrow{r''m''_2}\|$. This observation suggests it is possible to infer relative depth between hand joints from their 2D positions if we know them accurately. It is also possible and, more importantly, necessary to further learn a mapping between 2D coordinates and relative depth of the joints from data because there are some differences of the hand structure between individuals. Unfortunately, however, there exist at least two 3D poses can project to one 2D pose. As shown in the right of Fig. 2, two poses (denoted as mt_1 and mt_2) formed by a finger bent towards opposite directions have the same projection ($m't'$) on the image plane, which means we are unable to determine whether a joint is in front or back of its adjacent joints. Additional information is hence needed to deal with this problem. We are aware, inspired by the cognitive psychology of human beings, that the two 3D poses that correspond to the same 2D pose probably make the hand render different textures of and show different effects with illuminations on the skin. These differences could make human beings able to distinguish between the two 3D poses. Basing on these observations, we propose to explicitly extract 2D hand poses and hand textures separately and combine them together to infer 3D poses.

Furthermore, the biomechanical position and angle constraints on an individual finger and on fingers could also be beneficial to evaluation. Therefore, the prior can accordingly be categorized into two aspects; the first one is implicitly implied in the hand and can be learned from data while the second one can explicitly be defined. To integrate these tow types of priors appropriately, we use deep neural networks to learn a mapping $f_p : \mathbb{R}^2 \rightarrow \mathbb{R}$ that embeds implicit priors and can be used to regress relative depth d between joints from their 2D coordinates J^{2D}, i.e., $d = f_p(J^{2D})$. To deal with the depth ambiguity, we propose to extract intermediate hand texture features F_t as a supplementary information to 2D poses. As a result, we have $d = f_p(J^{2D}; F_t)$, where ; indicate the concatenation operation. On the other hand, we define a set of loss functions $\mathcal{L}$ to embed explicit priors, which can act as regularizers during network training.

Although in this paper we focus on hand pose estimation, the knowledge contained in hand shape reconstruction methods could be useful for our task. We thus adopt the hand mesh generated by a differentiable MANO layer and corresponding loss functions as a supervision mechanism to fine-tune the network. It is worth noting that this mechanism will be removed when our method is put into practice so that storage space and inference time of it can be reduced.

3.2 Network architecture

The architecture of our proposed deep neural network is shown in Fig. 1. It consists of (i) a two-stacked hourglass network as the backbone producing heatmaps of the hand joints and texture feature maps of the hand (ii) a multilayer perceptron (MLP) network that regresses 3D coordinates of hand joints from their 2D coordinates and hand texture features (iii) an MLP network that converts 3D coordinates into rotate angles (θ) of the MANO hand model (iv) an MLP network that regresses shape parameters (β) of the MANO model from the intermediate features (v) a differentiable MANO layer that generates a hand mesh according to pose and shape parameters.

Given an input image $I \in \mathbb{R}^{256\times256\times3}$, the hour glass network [15] f_{HG} extract features $F = (F_{hm}; F_t) = f_{HG}(I)$, where $F_{hm} \in \mathbb{R}^{64\times64\times21}$ are heatmaps and $F_{hm} \in \mathbb{R}^{64\times64\times512}$ represent hand texture features, respectively. Heatmaps are then added channel-wise to form a single heatmap, which are further flattened to a heatmap vector $\boldsymbol{v}_{hm} \in \mathbb{R}^{4096}$. Similarly, hand texture features are also converted to a texture vector $\boldsymbol{v}_t \in \mathbb{R}^{4096}$. An MLP network f_{3D}, which consists of 4 fully-connected layers, takes the concatenation of $\boldsymbol{v}_{hm}$ and $\boldsymbol{v}_t$ as input, and outputs 3D coordinates of hand joints $\boldsymbol{v}_{3D} = f_{3D}(\boldsymbol{v}_{hm}, \boldsymbol{v}_t)$, where $\boldsymbol{v}_{3D} \in \mathbb{R}^{63}$ is the concatenation of x, y, and z coordinates. Meanwhile, 2D positions of hand joints in image plane are obtained from heatmaps using $(u, v) = argmax_{(u,v)}\mathcal{H}(u, v)$, where $\mathcal{H}$ denotes the heatmap, and u, v is the pixel position in $\mathcal{H}$. We here use (u, v) rather than (x, y) to integrate with z, i.e., depth information d, to generate final 3D hand joint positions. Accordingly, prior biomechanical constraints will be imposed on (u, v, z). The reason that, in addition to (u, v), we evaluate (x, y) simultaneously is to preserve the proportion of (x, y) to z and then replace (x, y) with (u, v) proportionally in scenarios where camera intrinsic parameters are unknown.

The premise of using a MANO layer to generate hand meshes is obtaining pose ($\beta \in \mathbb{R}^{45}$) and shape ($\theta \in \mathbb{R}^{10}$) parameters according to the MANO model. On the one hand, inspired by [10] which uses eight fully-connected layers as a non-linear mapping network to map vectors from the input to an intermediate latent space, we use an MLP network f_θ, which also consists of eight fully-connected layers, to transform 3D positions J_{3D} to rotation angles $\theta = f_\theta(J_{3D})$ of hand joints. Specifically, six of these eight fully-connected layers are constructed as three residual connection blocks [25] and each block contains two layers. On the other hand, another MLP network f_β, which comprised of four fully-connected layers, is employed to extract shape parameter β from intermediate texture features F_t of the hand, i.e., $\beta = f_\beta(F_t)$. A differentiable MANO layer f_M is then utilized to produce hand meshes $\mathcal{M} \in \mathbb{R}^{N\times3}$ from both pose and shape parameters, i.e., $\mathcal{M} = f_M(\beta, \theta)$, where N is the number of vertices in the hand mesh and $N = 778$ for the MANO model.

3.3 Loss functions

To train the network effectively and efficiently, a set of loss functions are introduced to supervise different parts of it (see Fig. 1). They can generally be separated into three groups: (i) pose-related, (ii) prior-related, and (iii) mesh-related.

Pose-related loss This part consists of 2D heatmap loss and 3D pose loss. The ground truth heatmap is defined as a 2D Gaussian with a standard deviation of 1 px centered on the ground truth 2D joint location. 2D heatmap loss is defined as:

$$\mathcal{L}_{HM} = \Sigma_{j=1}^{J} \|\hat{\mathcal{H}}_j - \mathcal{H}_j\|_2^2, \tag{1}$$

where $\hat{\mathcal{H}}_j \in \mathbb{R}^{64\times 64}$ and $\mathcal{H}_j \in \mathbb{R}^{64\times 64}$ denote the estimated and ground-truth heatmaps, respectively. 3D pose loss is defined as:

$$\mathcal{L}_J = \Sigma_{j=1}^{J} \|\hat{\phi}_j^{3D} - \phi_j^{3D}\|_2^2, \tag{2}$$

where $\hat{\phi}_j^{3D} \in \mathbb{R}^3$ and $\phi_j^{3D} \in \mathbb{R}^3$ are the estimated and ground-truth 3D joint positions, respectively.

Prior-related loss This part comprises four loss items that represent the priors of an individual finger and between fingers. Similar to [28], we propose two extended geometric constraints to better restrict the position of the joints and the bending direction of the finger. Different from [28], we additionally include the wrist as the root joint of each finger (except the thumb). The wrist and other joints (from the MCP to the tip) of the finger are denoted as r, a, b, c , and d, respectively. We divide the extended five joints (including the wrist) of a finger into three segments: (r, a, b), (a, b, c), and (b, c, d). Firstly, these five joints should approximately be in the same plane. Accordingly, the loss function is formulated as:

$$\mathcal{L}_p = \alpha_p \langle \overrightarrow{ra} \times \overrightarrow{ab}, \overrightarrow{bc}\rangle + (1-\alpha_p)\langle \overrightarrow{ab} \times \overrightarrow{bc}, \overrightarrow{cd}\rangle, \tag{3}$$

where α_p is a hyperparameter; $\overrightarrow{ra}$, $\overrightarrow{ab}$, $\overrightarrow{bc}$, and $\overrightarrow{cd}$ represent the vector from joint r to a, a to b, b to c, and c to d, respectively; "$\langle\rangle$" and "$\times$" are inner and cross product operations of two vectors. Secondly, the three segments of a finger should bend towards the same direction. Corresponding loss function can be defined as:

$$\mathcal{L}_{df} = \alpha_{df}(\|\overrightarrow{bc}\| - \langle \overrightarrow{bc}, \frac{\overrightarrow{ab}}{\|\overrightarrow{ab}\|}\rangle) + (1-\alpha_{df})(\|\overrightarrow{cd}\| - \langle \overrightarrow{cd}, \frac{\overrightarrow{bc}}{\|\overrightarrow{bc}\|}\rangle), \tag{4}$$

where α_{df} is a hyperparameter. Thirdly, the length of the bones of a finger should follow the rule that the bone near the wrist is longer than the one near the tip. The loss function is defined as:

$$\begin{cases} \mathcal{L}_{len} = \frac{1}{2}[((|\delta_{rab}| - \delta_{rab}) + (|\delta_{abc}| - \delta_{abc}) + (|\delta_{bcd}| - \delta_{bcd})], \\ \delta_{rab} = \|\overrightarrow{ra}\| - \|\overrightarrow{ab}\|, \\ \delta_{abc} = \|\overrightarrow{ab}\| - \|\overrightarrow{bc}\|, \\ \delta_{bcd} = \|\overrightarrow{bc}\| - \|\overrightarrow{cd}\|. \end{cases} \tag{5}$$

Fourthly, the bending direction of all fingers (except the thumb) should be the same. We here introduce a concept of hand bending direction, which is determined by a simple majority of the bending direction of the four fingers. If there does not exist a majority, we will not handle it temporarily. We first define the bending degree of a finger as:

$$d = (\|\overrightarrow{ra}\| + \|\overrightarrow{ab}\| + \|\overrightarrow{bc}\| + \|\overrightarrow{cd}\|)/\|\overrightarrow{rd}\|. \tag{6}$$

According to the simple majority principle, the loss function can thus be defined as $\mathcal{L}_{dh} = d$. As a result, the prior loss in total is $\mathcal{L}_{prior} = \mathcal{L}_p + \mathcal{L}_{df} + \mathcal{L}_{len} + \mathcal{L}_{dh}$.

Mesh-related loss Similar to [24, 5], the mesh loss consists of vertex loss, edge loss, and normal loss. The vertex loss, which is used to restrict 3D positions of mesh vertices, is defined as:

$$\mathcal{L}_v = \Sigma_{i=1}^{N} \|\hat{\boldsymbol{v}}_i^{3D} - \boldsymbol{v}_i^{3D}\|_2^2, \tag{7}$$

where $\boldsymbol{v}_i^{3D}$ and $\boldsymbol{v}_i^{3D}$ are the estimated and ground-truth 3D position of the mesh vertices. The normal loss is employed to ensure surface normal consistency and is defined as:

$$\mathcal{L}_n = \Sigma_t \Sigma_{(i,j)\in F(t)} \|\langle \hat{\boldsymbol{v}}_i^{3D} - \hat{\boldsymbol{v}}_j^{3D}, n_t\rangle\|_2^2, \tag{8}$$

where t is the index of a triangular face F in the mesh; (i, j) are the indices of vertices that form an edge in a face; n_t is the normal of a ground-truth face. The edge loss is designed to impose edge length consistency, which is formulated as:

$$\mathcal{L}_e = \Sigma_{i=1}^{E}(\|\hat{\boldsymbol{e}}_i\|_2^2 - \|\boldsymbol{e_i}\|_2^2), \tag{9}$$

where $\hat{\boldsymbol{e}}_i$ and $\boldsymbol{e_i}$ are the estimated and ground-truth edge. The mesh loss in total is $\mathcal{L}_{mesh} = \mathcal{L}_v + \mathcal{L}_n + \mathcal{L}_e$.

The overall loss function of our method is:

$$\mathcal{L}_{total} = \lambda_{hm}\mathcal{L}_{hm} + \lambda_J\mathcal{L}_J + \lambda_{prior}\mathcal{L}_{prior} + \lambda_{mesh}\mathcal{L}_{mesh}, \tag{10}$$

where λ_{hm}, λ_J, λ_{prior}, and λ_{mesh} are hyperparameters. In our experiment, we set $\lambda_{hm} = 5$, $\lambda_J = 1$, $\lambda_{prior} = 0.1$, and $\lambda_{mesh} = 0.1$.

3.4 Implementation

Dataset The Rendered Hand Dataset (RHD) [31] and the Stereo Hand Pose Tracking Benchmark (STB) [27] are two widely used datasets in 3D hand pose estimation tasks. The former is a synthetic dataset while the latter a real one. These two datasets both provide 3D hand pose annotations. In addition to pose data, some newly published datasets [32, 13] further offer hand shape annotations. InterHand2.6M [13] is a large-scale real RGB-based 3D hand pose dataset, including both single and interacting hand sequences under various poses from multiple subjects. FreiHAND [32] additionally contains interaction with objects.

For the STB dataset, we select images with background 1 to 5 as training data and images with background 6 as evaluation data. The RHD dataset is split into training and evaluation parts according to [31]. Although the original FreiHAND dataset contains a sub-dataset for evaluation, it lacks pose and mesh annotations. Therefore, we divide the original dataset into four parts, each part includes 32560 images, and take the second part as the train dataset while the first 3000 images of the third part as the evaluation dataset. All hand images are cropped and resized to 256×256 pixels.

Data preprocess Similar to [31], we estimate a scale-invariant and root-relative 3D hand pose. We select the wrist r as the root joint and the distance between the wrist and the MCP of the middle finger as the scale s of the hand. The relative and normalized 3D coordinates of the joints are given by

$$\widetilde{J_i} = (J_i - J_r)/s, \tag{11}$$

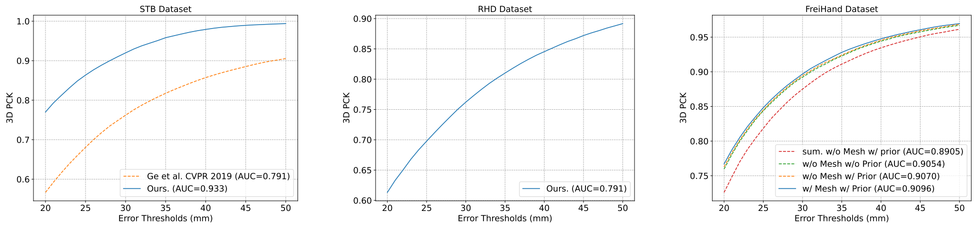
where $J_i \in \mathbb{R}^3$ is the original coordinates of hand joints. The mesh is also normalized in a similar way. We select vertices with the index of 33 and 370 as the wrist and the MCP of the middle finger, respectively. The relative and normalized 3D coordinates of mesh vertices are given by

$$\widetilde{\boldsymbol{v}}_i = (\boldsymbol{v}_i - \boldsymbol{v}_r)/s, \tag{12}$$

where $\boldsymbol{v}_i \in \mathbb{R}^3$ is the original coordinates of mesh vertices.

Data augmentation For images in all training datasets, we perform scaling (0.9 - 1.1) and rotation with an angle in $\{0°, 90°, 180°, 270°\}$. The 2D and 3D coordinates of hand joints and mesh vertices are also rotated accordingly. Moreover, we randomly change an image by color-jittering with the following configurations: brightness (0.9 - 1.1), contrast (0.85 - 1.15), saturation (0.9 - 1.1), hue (0.9 - 1.1), and apply randomly chosen Gaussian blur on images.

Training Our hand pose estimation method is implemented using PyTorch framework. The network is trained using Adam [11] optimizer with a batch size of 16 on a single GTX2080Ti GPU. The learning rate warmup strategy [6] is used in the first epoch to make the training process stabler. To train the neural network efficiently, we divide it into three main parts and train it in a gradually-extended manner. At first, we train the two stacked hourglass subnetwork using the heatmap loss with a learning rate of 0.001. We then further include 3D coordinates regressor using the 3D pose and prior-related loss with a learning rate of 0.0003. Finally, we train the whole network using all loss items with a learning rate of 0.0001.



(a) (b) (c)

Figure 3: (a) Comparison with state-of-the-art Ge et al. [5] on the STB dataset. (b) 3D PCK curve on the RHD dataset of our method. (c) Ablation study results for feature fusion manner and loss items.

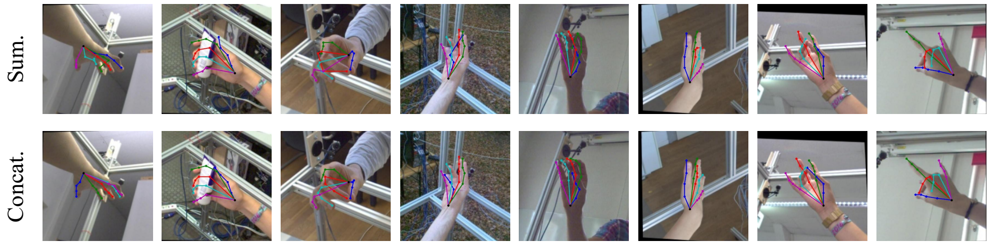


Figure 4: Qualitative comparison on the FreiHAND dataset between feature fusion manners: summation (top) and concatenation (bottom).

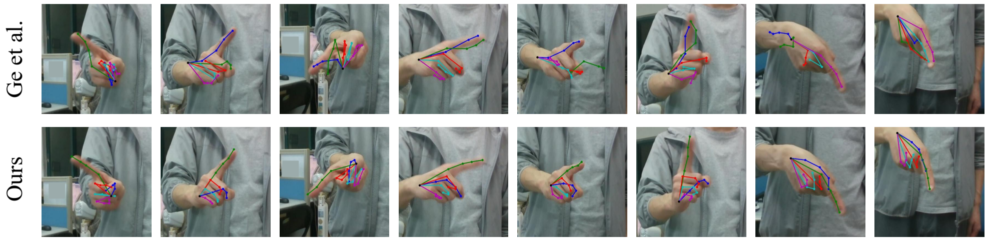


Figure 5: Qualitative comparison on a custom hand gesture dataset between method [5] (top) and ours (bottom).

4 Experiments

We evaluate our method on the STB, RHD, and FreiHAND datasets as well as a custom hand gesture dataset. For the STB and RHD datasets, we crop hand images and simultaneously record the position and size of corresponding bounding boxes. When carrying out pose evaluation, we transform the estimated hand pose back onto the original image coordinate system according to the bounding box and then calculate the error between the estimated and ground truth hand poses. Similar to [5], we report evaluation results with following metrics: (i) 3D PCK: the percentage of correct keypoints of which the Euclidean error distance is below a threshold; (ii) AUC: the area under the 3D PCK curve.

4.1 Comparison with start-of-the-arts

Firstly, we compare our method with other state-of-the-arts on the STB dataset. Although numerous methods have been proposed in recent three years, few of them provide currently available code and pre-trained models except for Ge et al. [5]. We train our method on the InterHand, FreiHAND, and STB dataset without mesh supervision as the STB dataset does not provide mesh annotations. We observed from [5, 28] that several recently proposed methods have achieved approximately saturated results on the STB dataset. However, when we evaluate a publicly available model of method [5], which is fine-tuned on the STB dataset, with the same settings as applied to our method, the result is far from saturated. Comparison between method [5] (fine-tuned on the STB dataset) and our method is shown in Fig. 3(a). It can be seen that our method is superior to [5] at all error thresholds with a large margin. The area under the PCK curve of method [5] and ours are 0.809 and 0.942, respectively.

We also evaluate our method on the RHD dataset. The network is trained on the InterHand, FreiHAND, and RHD dataset without mesh supervision. Since the model and code of other methods on this dataset are not publicly available, we here only report the result of our method, which is shown in Fig. 3(b).

In addition, we assess our method and method [5] on a custom hand gesture dataset to compare their cross-dataset generalization ability. Some results are demonstrated in Fig. 5. These hand images are of practical significance as they can be used to indicate various moving directions by the index finger in human-robot interaction. It can be seen that the result of our method is much better than that of Ge et al. [5]. An obvious mistake made by method [5] is it tends to incorrectly recognize the index finger, which should be coloured green, as the thumb, which should be coloured blue.

As shown in Table 1, we also compare the complexity between our method and method [5]. It can be observed that our method is simpler than [5] in terms of storage space, parameter count, and FLOPs. This further demonstrates the superiority of our method.

4.2 Ablation study

In this paper, we propose to regress relative depth of hand joints from the combination of 2D hand poses and hand textures and use concatenation rather than summation to fuse them together. In addition, we mainly introduce prior-related and mesh-related loss items. To better understand the contribution of these components in our method, we evaluate it with different configurations. We set the full model **(w/ prior-related loss, w/ mesh-related loss)** as the baseline, and compare it with baseline-pm **(w/o prior-related loss w/o mesh-related loss)** and baseline-m **(w/ prior-related loss w/o mesh-related loss)**. We also assess the difference between concatenation and summation as the feature fusion manner. The network is trained on the InterHand and FreiHAND dataset since they both provide mesh annotations.

Table 1: Comparison of space and time complexity between methods.

Method	Size (MB)	Params (M)	FLOPs (G)
Ge et al. [5]	87.6	21.76	16.34
Ours	85	17.33	13.69

Ablation study results for feature fusion manner and loss items are shown in Fig. 3(c). It can be seen that the concatenation of 2D hand poses and hand textures significantly improve accuracy because explicitly extracting 2D hand poses and hand textures separately and concatenating them together can better fuse features and exploit the capacity of the network. Some evaluation results of these two feature fusion manners on the FreiHAND dataset are shown in Fig. 4, demonstrating partial as well as global improvement. Prior-related and mesh-related loss items all play their roles in performance gains; the former can exert necessary geometric constraints on and thus be helpful in producing more feasible 3D hand poses while the latter provide a more detailed mesh-level supervision over hand pose fine-tuning, which is also beneficial to accuracy enhancement.

5 Conclusion

In this paper, we address two core issues involved in 3D hand pose estimation: (i) improving relative depth evaluation between hand joints; (ii) producing more feasible 3D hand poses. We propose a Prior-knowledge and Mesh Supervision Network (PMSNet) to integrate prior knowledge and mesh supervision effectively. The prior knowledge is divided into implicit and explicit aspects; the former is directly learned from data and the latter is embedded in a set of loss functions. To better infer relative depth between hand joints, we explicitly extract 2D hand poses and hand textures separately and fuse them with concatenation operation. In addition, we employ hand meshes as a supervision mechanism to fine-tune the network. Experimental results show that our method outperforms state-of-the-arts with a considerable margin in terms of accuracy and generalization ability. However, the contribution of the mesh supervision mechanism is, to some extent, lower than expected. Further studies are required to better understand and exploit this mechanism.

Acknowledgments and Disclosure of Funding

This work was supported by the Key-Area Research and Development Program of Guangdong Province (2019B090915002).

References

[1] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3d hand shape and pose from images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10843–10852, 2019.

[2] Yujun Cai, Liuhao Ge, Jianfei Cai, Nadia Magnenat Thalmann, and Junsong Yuan. 3d hand pose estimation using synthetic data and weakly labeled rgb images. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):3739–3753, 2020.

[3] Ping Chen, Yujin Chen, Dong Yang, Fangyin Wu, Qin Li, Qingpei Xia, and Yong Tan. I2uv-handnet: Image-to-uv prediction network for accurate and high-fidelity 3d hand mesh modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12929–12938, 2021.

[4] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[5] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10833–10842, 2019.

[6] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[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.

[8] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 571–580, 2020.

[9] Umar Iqbal, Pavlo Molchanov, Thomas Breuel Juergen Gall, and Jan Kautz. Hand pose estimation via latent 2.5 d heatmap regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 118–134, 2018.

[10] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] Taein Kwon, Bugra Tekin, Jan Stühmer, Federica Bogo, and Marc Pollefeys. H2o: Two hands manipulating objects for first person interaction recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10138–10148, 2021.

[13] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2. 6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. In *European Conference on Computer Vision*, pages 548–564. Springer, 2020.

[14] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[15] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[16] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022.

[17] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017.

[18] Adrian Spurr, Jie Song, Seonwook Park, and Otmar Hilliges. Cross-modal deep variational hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 89–98, 2018.

[19] Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Otmar Hilliges, and Jan Kautz. Weakly supervised 3d hand pose estimation via biomechanical constraints. In *European Conference on Computer Vision*, pages 211–228. Springer, 2020.

[20] Adrian Spurr, Aneesh Dahiya, Xi Wang, Xucong Zhang, and Otmar Hilliges. Self-supervised 3d hand pose estimation from monocular rgb via contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11230–11239, 2021.

[21] Srinath Sridhar, Antti Oulasvirta, and Christian Theobalt. Interactive markerless articulated hand motion tracking using rgb and depth data. In *Proceedings of the IEEE international conference on computer vision*, pages 2456–2463, 2013.

[22] Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao. Self-supervised 3d hand pose estimation through training by fitting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10853–10862, 2019.

[23] Jiayi Wang, Franziska Mueller, Florian Bernard, Suzanne Sorli, Oleksandr Sotnychenko, Neng Qian, Miguel A Otaduy, Dan Casas, and Christian Theobalt. Rgb2hands: real-time tracking of 3d hand interactions from monocular rgb video. *ACM Transactions on Graphics (ToG)*, 39(6): 1–16, 2020.

[24] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 52–67, 2018.

[25] Zeye Wu and Wujun Che. 3d human pose lifting: From joint position to joint rotation. In *Chinese Conference on Image and Graphics Technologies*, pages 228–237. Springer, 2019.

[26] Baowen Zhang, Yangang Wang, Xiaoming Deng, Yinda Zhang, Ping Tan, Cuixia Ma, and Hongan Wang. Interacting two-hand 3d pose and shape reconstruction from single color image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11354–11363, 2021.

[27] Jiawei Zhang, Jianbo Jiao, Mingliang Chen, Liangqiong Qu, Xiaobin Xu, and Qingxiong Yang. A hand pose tracking benchmark from stereo matching. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 982–986. IEEE, 2017.

[28] Xiong Zhang, Qiang Li, Hong Mo, Wenbo Zhang, and Wen Zheng. End-to-end hand mesh recovery from a monocular rgb image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2354–2364, 2019.

[29] Xiong Zhang, Hongsheng Huang, Jianchao Tan, Hongmin Xu, Cheng Yang, Guozhu Peng, Lei Wang, and Ji Liu. Hand image understanding via deep multi-task learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11281–11292, 2021.

[30] Zimeng Zhao, Xi Zhao, and Yangang Wang. Travelnet: Self-supervised physically plausible hand motion learning from monocular color images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11666–11676, 2021.

[31] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *Proceedings of the IEEE international conference on computer vision*, pages 4903–4911, 2017.

[32] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 813–822, 2019.