Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Design of 3D Hand Mesh Reconstruction from Monocular Image

Tsung-Han Tsai

han@ee.ncu.edu.tw

National Central University

Jia-Yang Jiang

National Central University

Research Article

Keywords: CNNs, Monocular Hand pose Estimation, Monocular Hand shape Estimation

Posted Date: November 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8341709/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Design of 3D Hand Mesh Reconstruction from Monocular Image

Tsung-Han Tsai
Dept. of Electrical Engineering
National Central University
Taoyuan City, Taiwan
han@ee.ncu.edu.tw

Jia-Yang Jiang
Dept. of Electrical Engineering
National Central University
Taoyuan City, Taiwan
jjyp@ee.ncu.edu.tw

***Abstract*—In recent years, as the impact of deep learning on people's lives has grown, more and more attention has been paid to the development of this field. The task of human hand pose and shape estimation from an RGB image has been a long-standing problem in the field of computer vision. Unlike common hand posture prediction, which only predicts the coordinates of the skeletal points of the hand, this task restores the original shape of the hand. Many places will apply this task such as augmented reality and virtual reality, but the task is still very challenging because the hand occupies a relatively small part of the image area, and the hand movements are flexible and easy to block. In this paper, we propose an end-to-end network for 3D hand mesh reconstruction from RGB images. Using ResNet-50 as the encoder, we extract image features and generate 2D heatmaps and masks through convolutional layers to improve parameter regression. The model parameters are iteratively refined with fully connected layers. To correct unnatural hand poses, we apply a graph convolutional network to adjust mesh coordinates, combining these adjustments with the initial MANO-generated model. Experiments demonstrate effective 3D hand mesh reconstruction from monocular images.**

***Index Terms*—CNNs, Monocular Hand pose Estimation, Monocular Hand shape Estimation**

I. Introduction

Hands are one of the important tools for interaction with the world so the prediction of human hand pose and shape is very important for applications such as AR, VR, and human-computer interaction in games. Accurate hand mesh prediction allows users to naturally interact with virtual environments using gestures, such as grabbing, dragging, and rotating objects. By precisely tracking hand movements, it can significantly enhance the immersion and realism of the user experience in games. Regarding accuracy requirements, low latency is essential for real-time feedback and a natural interaction experience, ensuring synchronization between hand movements and the virtual hand display. Nowadays, many works on 3D hand analysis focus more on the study of hand pose, predicting the skeletal points of the hand [1][2][3][4], actually many applications only need to predict the shape of our hand. Of course, the prediction of the 3D pose and shape of the human hand is very challenging. Many approaches require the use of simultaneous multiview images or images with depth information [5][6][7][8], so the prediction of the 3D pose and shape of the human hand is particularly challenging in the case of monocular RGB images, not only because of the depth ambiguity but also because of the self-occlusion of the hand. The reasons for this include not only the depth blur, and ego occlusion of the hand, but also the flexibility of hand movements. Considering that we encounter monocular RBG images more often in our lives, 3D hand mesh reconstruction from monocular RBG images is a long-term task in the computer vision field.

The prediction of 3D pose and shape of the human hand is divided into two main approaches in today's research. The first is the model-based approach, using a parametric model such as MANO [9]. The neural network only needs to learn the shape and pose parameters of the generated model. The network architecture of this approach is straightforward: it includes an encoder for extracting features from the input RGB image and a decoder for regressing the model parameters. The advantage of this approach is obvious it has good robustness in complex environments and environmental changes. The disadvantage is also obvious, that is, it is limited by the parametric composition of the model. As a result, for some hand shapes and postures, there is no way to achieve a complete recovery. Simply put, the parametric model is a strong structure so that it can still produce a natural hand shape even when the hand is blocked or overlapped. But at the same time, it is not conducive to making special hand shapes and postures due to the limitations of the parametric model. The second method is the non-model-based method. Because of the many fixed points on the mesh, the non-model-base method does not apply to the neural network methods to regress the coordinates of the mesh vertices. This approach can effectively compensate for the shortcomings of the model-based approach but sometimes the model is not very efficient.

The prediction of human hand pose and shape from the network architecture can be subdivided into two types, the first is the one-stage approach which does not generate intermediate representations in the middle process of the model. This architecture usually uses a conventional backbone network such as ResNet [10] to extract various features of the input RGB

The authors are with the Department of Electrical of Engineering, National Central University, Zhongli, Taiwan.

Tsung-Han Tsai is with National Central University, Zhongli, Taiwan (email: han@ee.ncu.edu.tw).

Jia-Yang Jiang is with National Central University, Zhongli, Taiwan (e-mail: jjyp.ee.ncu.edu.tw).

images in the encoder part and then uses the decoder to return the model parameters we need. Therefore, this architecture is dedicated to design better coders and decoders to achieve our task requirements. The second approach is similar to the multi-stage approach. Instead of using the input RGB image directly for prediction, this approach will first generate intermediate representations such as hand contours, 2D heatmap, etc., and then use these intermediate representations for regression to generate the final model parameters. This multi-stage approach usually uses the convolution layer to generate the intermediate representations first. It can simplify and guide the subsequent parameter regression and reduce the difficulty of parameter regression. The intermediate representations can effectively eliminate the effect of background and brightness on the regression. Therefore, the intermediate representations map provides a new starting point and effectively reduces the difficulty of hand pose and shape prediction.

Over the years, due to the rise of AR and VR, the task of predicting 3D hand pose and shape has increasingly come into the limelight, and now there has been very obvious progress. Among them, images with depth information or multi-angle images as input have reached a considerable degree of accuracy, but, more often in our lives, we use monocular RGB images as input for 3D hand pose and shape prediction.

Since the prediction of 3D hand pose and shape can be divided into two approaches, model-base and non-model-base, in this paper, we try to combine the advantages of both approaches to avoid the disadvantages of both approaches. Our approach starts with extracting the features of the input RGB image and then using the convolution layer to generate intermediate representations such as hand contours, 2D heatmap, etc., and then regressing the corresponding hand model parameters (MANO) to generate a preliminary 3D hand mesh, and then adding the hand features in the image to this rough 3D hand mesh to enter the graphical convolution network for learning. We reduce the difficulty of regression model parameters by a multi-stage method to generate intermediate representations first, and then use the graphical convolution network to generate the offset of the vertex coordinates on the 3Dmesh. Our main contributions are addressed as follows:

- We propose the use of graphical convolutional networks to correct the hand meshes generated by parametric models, which effectively improves the accuracy of 3D hand pose and shape prediction.
- We generate the hand contours, hand 2D heatmap, and other intermediate forms for the subsequent regression as input to effectively reduce the difficulty of the subsequent regression parameters. Compared with the original input RGB image, it effectively reduces the interference of the background and reduces the impact of brightness.
- We evaluated the predictions of 3D hand pose and shape on the FreiHAND dataset [11] and obtained good results.

This paper is organized as follows. Section I is the introduction. In Section II, we discuss the related works. The details of our proposed neural network architecture are highlighted in Section III. Experimental results are shown in Section IV. Finally, the conclusions are given in Section V.

II. Related Works

Due to the large scope of relevant fields, it is difficult to comprehensively summarize all the work. We will only discuss the work closely related to our framework setting. In this chapter, we only discuss the methods of estimating 3D hand posture using a monocular camera. These methods are closely related to our work.

A. 3D hand pose estimation

The task of 3D hand pose prediction aims to regress the 3D positions of hand joints from RGB or depth images. Recently, 3D hand pose prediction has been mainly based on deep learning methods [1][12][13][14][15][16][17][18][19][20][21]. Zimmermann et al. [1] implemented the early deep learning techniques to estimate a 3D hand pose from a single RGB image . [22] further predicted the posture of the hand object. In a recent study, Transformer [23] was found to be very suitable to be used to improve the accuracy of hand pose estimation [24][25]. Among them, [24] proposed that 3D hand pose prediction is generally divided into two parts firstly 2D skeleton point detection to find out the location of skeleton points on the image, and then conversion from 2D to 3D. [26] proposed a new 3D hand pose estimation method called CCPI, which aims to improve the system's generalization ability and performance through dynamic progressive inference and cross-constraints.

B. Depth base 3D hand pose and shape estimation

Since the depth cameras are gradually starting to come into our lives and the number of depth cameras around us is increasing, many works on estimating hand meshes based on depth images are started. The early depth-based work [27][28] matches a deformable hand mesh with a depth image by iterative computational optimization. However, the drawbacks of such an approach are obvious, for example, the computational effort is very large and the accuracy of the model is very dependent on the manually set similarity function. Later [29] proposed to combine CNN with stochastic decision forest and inverse kinematics to estimate the hand pose in real-time from a single depth image. In [30][31], a feedback loop is designed to correct the prediction errors using the feedback loop, where the feedback loop is also needed to train the neural network, and the advantage of this design is that it solves the problem of not having to manually create 3D models as input. [31] suggested additional consideration of the function and importance of each finger. [33] presented a joint estimation of hand shape and posture. Some works exploit some other representations such as point clouds [34] and three-dimensional voxels [35], which can be transformed from depth images. [36] Real-time tracking of hand interactions with the depth camera.

C. Monocular base 3D hand pose and shape estimation

To address the shortcomings in the task of 3D hand mesh reconstruction based on depth images for human beings people

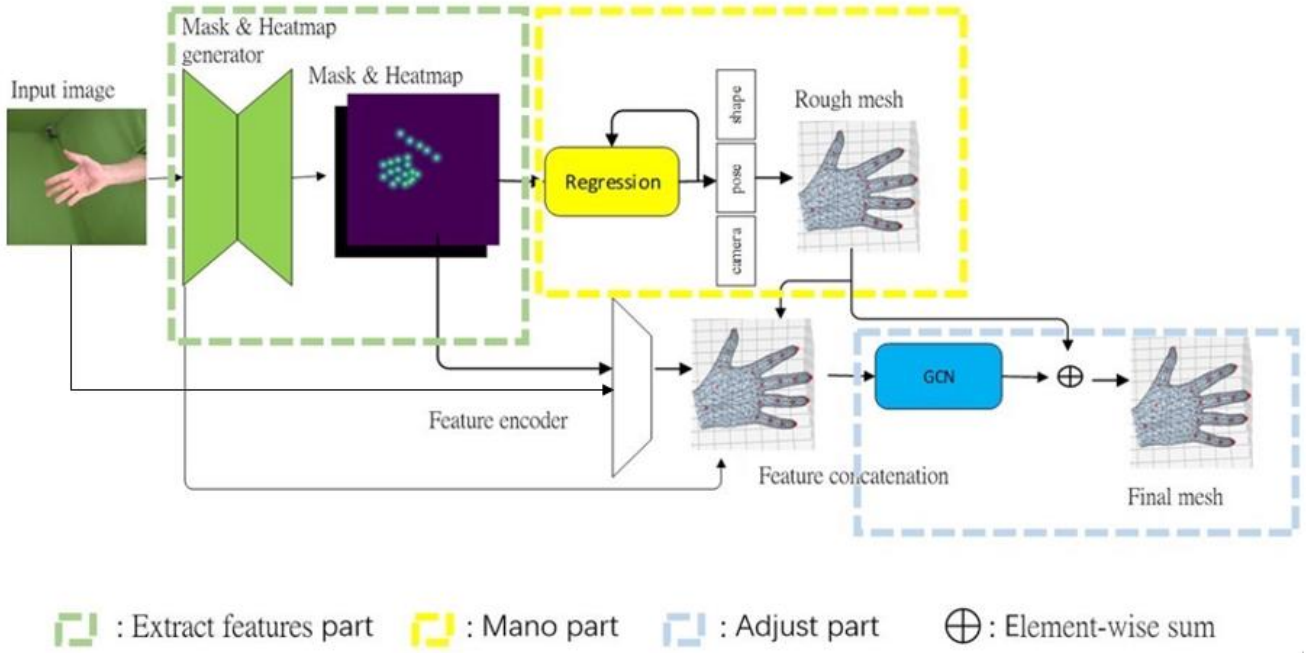


Fig. 1. Our proposed framework

are also investigating 3D hand mesh prediction from monocular RGB images, which is more challenging than depth-based settings due to depth ambiguity. Aiming to reconstruct 3D hand mesh using a monocular RGB camera [37][38]. One of the more popular approaches is to use the model-based approach to allow the network to regress model parameters such as the MANO model parameters. In other words, the model only needs to output a vector of low latitude, and the MANO model has a total of 61 parameters. So let's discuss some papers that use the model-based approach. In [39], the parameters of the MANO model are regressed and the predicted MANO model is projected into the image space using a reprojection to obtain a 2D pose of the hand, which effectively overcomes the problem of sparse 3D hand pose data set. The approach of this paper is similar to that of [40]. [41] is also a model-based approach to regression of MANO hand model parameters called Hand Mesh Recovery HAMR (HAnd Mesh Recovery) framework. Both [41] and [42] used a projection of hand contours to enhance training. [43] first predicted the 3D coordinates of the hand joint, and then used the joint prediction as before generating the hand mesh through the inverse kinematics network. Since the model-base approach, there is a model base in which the hand mesh prediction is helpful, but in some cases, it also binds the hand mesh prediction, and the hand shape of some postures is not reconstructed. To solve this problem, we have a non-model-base approach, which usually uses a graphical neural convolution network or a transformer to directly predict the 3D coordinates of the vertices on the hand mesh, instead of regressing the hand model parameters. We know that graphical neural convolutional networks are designed for the aggregation of neighboring features based on a mesh topology, so they are less efficient for long-range features. To overcome this drawback, [44] proposed to use of Graph-CNN to regress hand network vertices in a coarse-to-fine manner. [45] proposed that most of the existing work on hand mesh reconstruction is focused on improving the accuracy of hand gestures. Therefore, there is no requirement to align their generated hand meshes with the image, but this alignment is crucial for AR applications. [46] used the transformer approach and achieved quite excellent results on both human mesh and hand mesh reconstruction.

D. 3D hand pose and shape estimation with transformer

Recent studies show a noteworthy enhancement in performance across various natural language processing tasks through the implementation of the transformer architecture [47]. This success is primarily ascribed to the self-attention mechanism of the transformer, proving highly effective in modeling dependencies or interactions, regardless of their distance in both input and output. By capitalizing on these dependencies, the transformer excels in conducting a soft search for relevant tokens and making predictions based on crucial features.

However, recent research has also observed the adoption of the transformer architecture in 3D pose and mesh reconstruction [48]-[49]. Despite the significant potential of the transformer architecture in enhancing model performance, it's crucial to acknowledge its complexity. Transformer models typically entail a large number of parameters, which may lead to computationally expensive and time-consuming training and inference processes. Therefore, when opting for the transformer architecture, it's essential to strike a balance between performance enhancement and computational cost and ensure the efficient utilization of computational resources to achieve optimal results.

III. Proposed Approach Overview

The network architecture proposed in this paper is shown in Fig. 1. Our method consists of three main modules. First is the Extract features part to extract image features, then is the MANO part to regress model parameters, and finally is the adjustment part to correct the hand mesh coordinates.

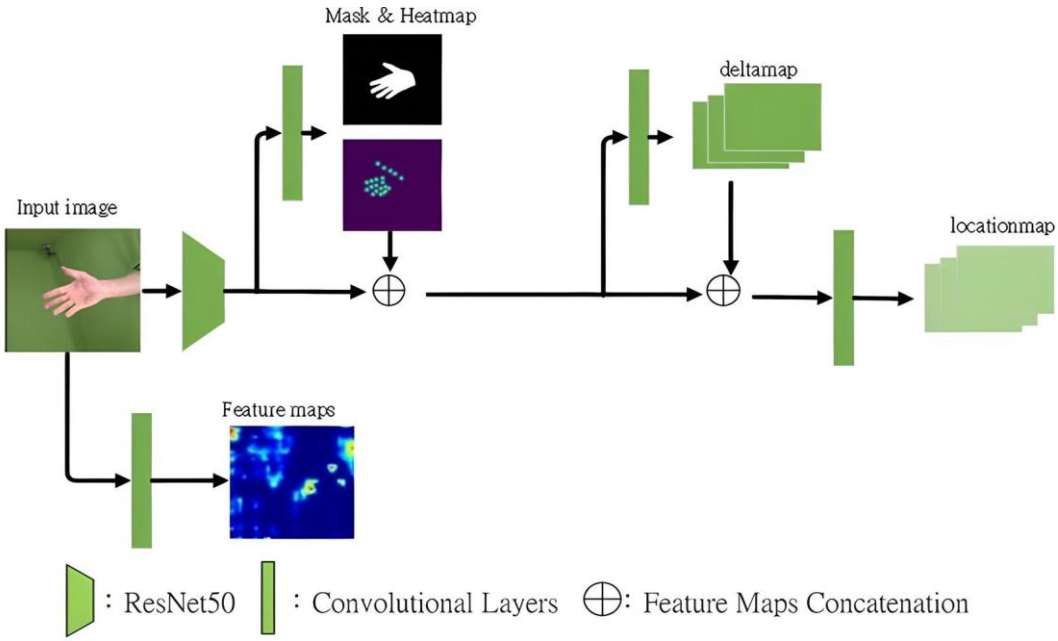


Fig. 2. Extract features part framework

A. Extract features part

Based on the monocular RGB image 3D hand mesh prediction is difficult, so to do this task well, we should do all the work at the beginning of the feature extraction stage. Here, we perform feature extraction using the specific network structure as shown in Fig. 2. Our inspiration comes from [50], it is certainly possible to directly use the input images for regression, but it is not a very good way. Generating a kind of intermediate representation as we did can better simplify the input and lead the later network to better regress the parameters. Generally speaking, intermediate representations are generated by some convolutional networks after the backbone network, and they usually represent 2D features or 3D features of the input image. They include but are not limited to mask images, 2D heatmap, 3D heatmap, 2D skeleton point location 3D skeleton point location, etc. The role of the mask image, for example, is very obvious. The hand mask image simplifies the input image very effectively, eliminates the interference of the background in the image, and also avoids the influence of the brightness of the environment on the model. Another example is the intermediate morphology characterized by the skeleton point coordinates, which also very obviously reduces the difference in morphology between the task input and output. Imagine a hand mesh from an RGB image, but now there is an additional hand skeleton point in the middle, which also helps to improve the model. At the same time, the characteristics of the intermediate morphology can also guide the model to accomplish the prediction task. For those who have related papers, 2D heatmap or 3D heatmap is used in combination with the input image so that the network can target the hand part of the image for feature extraction. In short, introducing an intermediate form of features for regression instead of directly using the input image can effectively improve the capability of

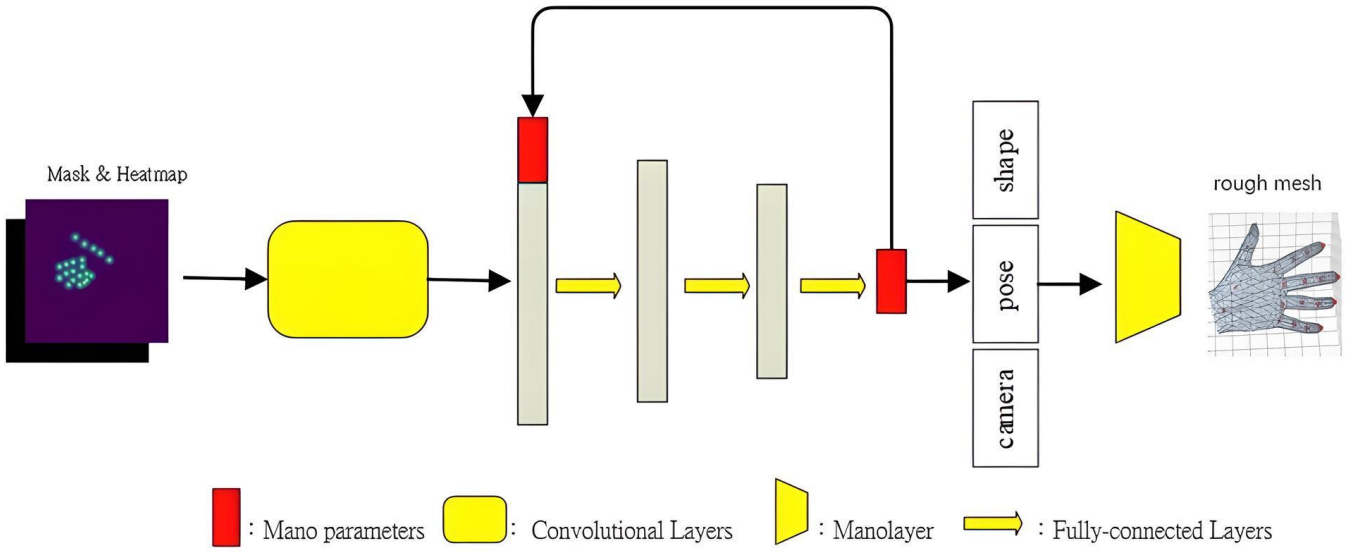


Fig. 3. MANO part framework

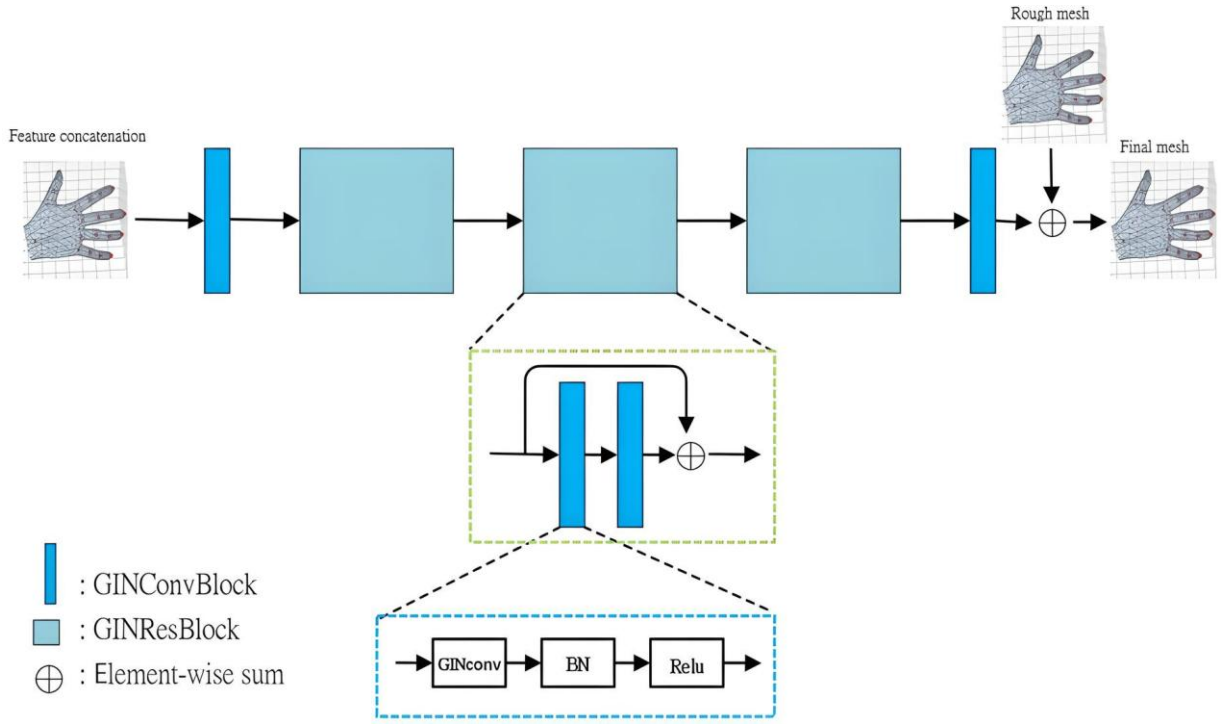


Fig. 4. Adjust part framework

the network model. Our approach is borrowed from [43], where the input RGB image is extracted by ResNet-50 and the 2D heatmap and mask images of the hand are generated using the convolutional layer, and then the 2D heatmap and ResNet-50 extracted features are added into the convolutional layer to obtain the Deltamap. The four intermediate representations including hand 2D heatmap, mask image, Deltamap, and Locationmap are all supervised in our paper. Adding the intermediate representations of Deltamap and Locationmap is worth mentioning, as it will significantly enhance the model's accuracy.

B. MANO part

The MANO part is relatively simple, and the network architecture we borrowed from [51] is shown in Fig. 3. It should be said that a significant part of the model-based approach in 3D hand mesh prediction based on monocular RGB images is based on the iterative error feedback approach proposed in the previous work [52] to generate model parameters. The MANO part receives the 2D heatmap, mask, and feature map of the hand from the previous network organizes this information using a convolutional layer, and then enters the iterative error feedback module, which is a three-layer fully connected layer with an output of 61 values. Among these, the 61 parameters include camera, pose, and shape. Camera parameters project 3D skeleton points onto the 2D image plane, pose parameters define the hand model's posture, and shape parameters determine finger thickness and length. The input of the iterative error feedback module is the feature map of the previous convolutional layer plus 61 parameters of the MANO model, and the 61 parameters corresponding to the sample hand type T-pose in the MANO hand model mentioned earlier in this paper are used as initial values in the first round of iterative computation. After starting the iterative computation with error feedback, the 61 parameters of the last level of output are returned to the beginning as input, thus achieving the effect of iterative computation. Then why do these parameters need to be iterated instead of being directly regressed? It is because the parameters of MANO are not so easy to regress, especially the pose parameters, which need to be corrected step by step by iterative error feedback. The manolayer is a differentiable Pytorch layer that maps the pose and shape parameters to the hand skeleton points and hand mesh so that we can get the rough mesh mentioned in the paper.

C. Adjust part

In this section of the Adjust part, we learn to adopt the idea of graphical convolutional neural networks from [53], where we use graphical neural convolutional networks to correct our 3D mesh hand coordinates. We start by adding all the features to our previously generated crude mesh as input to the graphical neural network. For the selection of the graph convolution layer, we use GIN (Graph Isomorphism Network) [54] because of its more powerful representation compared to other approaches, as defined in Equation 1, where x_i is the feature of the ith point on the 3D hand mesh. $x_i^{'}$ is the feature of this point after updating. $N(i)$ is the set of index values of all vertices on the 3D hand mesh. $MLPs(\cdot)$ is the multi-layer perceptron.

$$x_i^{'} = MLPs(x_i + \sum_{j \in N(i)} x_j) \quad (1)$$

We use the local feature processing unit and the full domain feature processing unit in [53] to process the features in front of the network, including image features, image shallow features, image deep features, skeleton point features, and 3D hand mesh features. These features are added to the crude hand mesh and then entered into the graphical neural convolution network, as

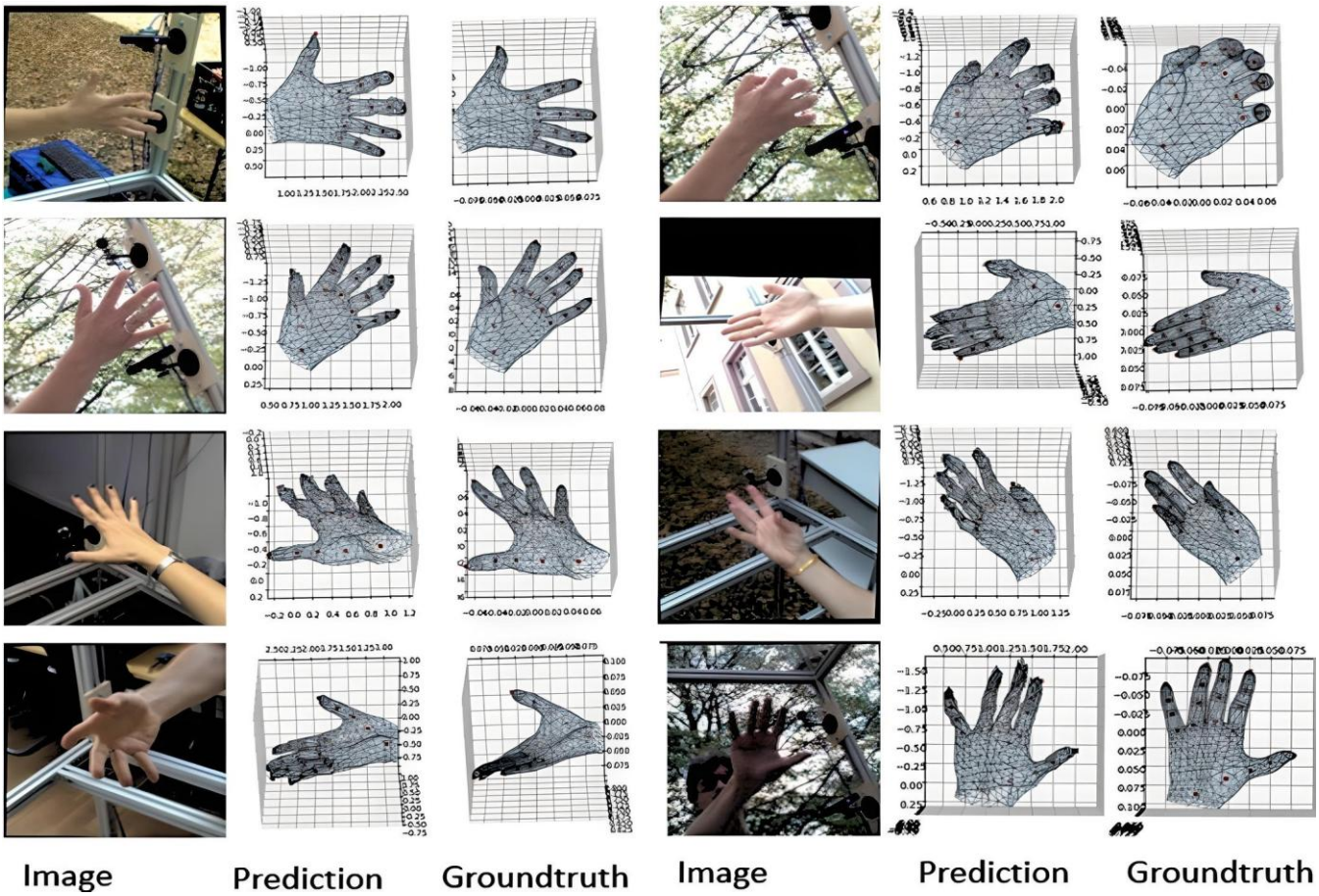


Fig. 5. The results of our model tested on the FreiHAND dataset are visualized in the figure, the left side is the input image, the middle is the model prediction result, and the right side is the Ground truth

shown in Fig. 4. One *GINConvBlock* is composed of a set of *GINconv*, *BN*, and *Relu*, while the *GINResBlock* is a residual module composed of two *GINConvBlock*s.

D. Training & Inference

In the loss function section, we use several loss functions to constrain our network for training. These include L_E , L_{ref}, L_V, L_{M2J}, L_{MASK}, and L_{J3D}. In the following, we will introduce the loss functions in this paper. Firstly, L_E includes three loss functions, namely L_H, L_D and L_L, which represent the loss functions of 2D heatmap, deltamap and locationmap respectively. All of them use the Frobenius norm, where L_D and L_L use the ground truth of heatmap as the weighted value. Equations 2, 3 ,and 4 are shown respectively. H^{GT} , D^{GT} and L^{GT} are generated by 3D skeleton points, $\|\cdot\|_F$ is the Frobenius norm, and $\odot$ is the matrix product between elements. Equation 5 represents L_E where λ_H is 100.

$$L_H = \|H^{GT} - H\|_F^2 \tag{2}$$

$$L_D = \|H^{GT} \odot (D^{GT} - D)\|_F^2 \tag{3}$$

$$L_L = \|H^{GT} \odot (L^{GT} - L)\|_F^2 \tag{4}$$

$$L_E = \lambda_H * L_H + L_D + L_L \tag{5}$$

L_{ref} includes two loss functions, L_{pose} and L_{shape}, which represent the loss function of the MANO pose parameter and the loss function of the MANO shape parameter as shown in Equations. 6 and 7, respectively. The L2 parametrization is used and the performance advantage is calculated in a way similar to the Mean Square Error Loss. The definition is the open square of the sum of squares of all elements of the vector. θ^{GT} and β^{GT} are Ground truth of pose and shape parameters respectively, $\|\cdot\|_2^2$ is L2 norm, and equation 8 represents L_{ref} where λ_{pose} is 0.1.

$$L_{pose} = \|\theta^{GT} - \theta\|_2^2 \tag{6}$$

$$L_{shape} = \|\beta^{GT} - \beta\|_2^2 \tag{7}$$

$$L_{ref} = \lambda_{pose} * L_{pose} + L_{shape} \tag{8}$$

L_V includes two loss functions L_{final} and L_{rough} representing the loss function of the final hand mesh generated in the model and the loss function of the crude hand mesh as shown in Equations 9 and 10, respectively. The L1 parametrization is used, which is defined as the sum of the absolute values of each element in the vector. V^{GT} is the Ground truth of 3D hand mesh, V^R and V^F are the rough hand mesh and the final hand mesh generated by the model, $\|\cdot\|_1$ is the L1 parametrization, Equation 11 represents L_V where λ_{rough} is 10, λ_{final} is 10.

$$L_{rough} = \|V^{GT} - V^R\|_1 \tag{9}$$

$$L_{final} = \|V^{GT} - V^F\|_1 \tag{10}$$

$$L_V = \lambda_{rough} * L_{rough} + \lambda_{final} * L_{final} \tag{11}$$

Table I：Accuracy comparison table for predicted pose in FreiHAND dataset

Authors (Publication)	Pose Error (mm) ↓	Pose AUC ↑	Mesh AUC ↑	F-score at 5mm ↑	F-score at 15mm ↑	Parameters	FPS
Handar(ICCV'21)[53]	-	0.868	0.866	0.724	0.981	426M	39.1
Pose2Mesh(ECCV'20)[44]	0.74	-	-	0.683	0.973	75M	22
Boukh. Et al(CVPR'19)[39]	1.32	0.351	0.738	0.427	0.894	-	10
Kan et al.(AAAI'24)[26]	-	-	-	-	**0.987**	198.74M	-
Wang et al.(ICMA'23)[58]	-	-	-	0.728	**0.981**	109M	-
LIN et al.(ECCV'22)[49]	-	-	-	-	0.982	153.0M	20
CHO et al.(CVPR'21)[48]	-	-	-	-	0.981	230.4M	5
MobileHand(ICONIP'20)[59]	-	-	-	0.439	0.902	-	110
Hasson et al.(CVPR'19)[57]	1.33	0.737	0.736	0.429	0.907	-	20
MANO Fit(ICCV'19) [9]	1.37	0.73	0.729	0.415	0.884	-	-
Mean Shape(ICCV'19) [11]	1.71	0.662	0.674	0.3	0.808	-	-
Ours	**1.26**	**0.752**	**0.74**	**0.459**	0.922	**56.33M**	**57.3**

L_{M2J} , L_{MASK} and L_{J3D} denote the loss function of 3D skeleton points generated by the predicted hand mesh, the loss function of the mask image, and the loss function of 3D skeleton points generated by the MANO model, respectively. As shown in Equations 12, 13 ,and 14, κ is the regression matrix of skeleton points defined by MANO, J^{GT} is the Ground truth of 3D skeleton points, M^{GT} is the Ground truth of hand mask, and M and J are the predicted hand mask and 3D skeleton points, respectively. The configuration of our final target loss function is shown in Equation 15, where λ_{M2J}, λ_{MASK} and λ_{J3D} are all 10.

$$L_{M2J} = \|J^{GT} - \kappa V^{GT}\|_2^2 \tag{12}$$

$$L_{MASK} = \|M^{GT} - M\|_1 \tag{13}$$

$$L_{J3D} = \|J^{GT} - J\|_2^2 \tag{14}$$

$$L_{TOTAL} = L_E + L_{ref} + L_V + \lambda_{M2J} * L_{M2J} + \lambda_{MASK} * L_{MASK} + \lambda_{J3D} * L_{J3D} \tag{15}$$

IV. Experimental Results

This section of the thesis describes the results of the approach proposed in this paper compared to those of other papers, as well as an analysis of the performance improvement of the different modules used in the model of this paper, and the configuration of the parameters we used to train the network. Our paper will be tested on the well-known public dataset FreiHAND. This chapter consists of four subchapters in total. They are dataset, training details, comparison of experimental results, and ablation experiments.

A. Datasets

Estimating 3D hand meshes from monocular RGB images is a difficult problem that relies on a good dataset for training. The FreiHAND dataset is a large-scale, multi-viewpoint hand dataset to solve this problem. hand movements performed by 32 people. The training set contains 32560 unique samples, which are post-processed in four different ways to remove the green screen background. Simply put, the backgrounds of the same hand shapes are different, some are green screen backgrounds and some are outdoor scenes, which effectively improves the accuracy of the models in outdoor recognition after training. The FreiHAND dataset provides RGB images (224*224 pixels), a hand segmentation mask (224*224 pixels), an intrinsic camera matrix, a hand scale, and a 3D hand annotation based on Mano.

To further investigate the generalization capability of the model, we include HO3D-V2 as an additional test set. This dataset contains 11,524 test samples and allows us to assess the model's performance across different scenarios.

B. Training and Implementation details

We use two Nvidia Geforce RTX 3080 Ti GPUs simultaneously in the PyTorch [56] deep learning framework. RGB images from the FreiHAND dataset as input, and first we preprocess the input images by adding Gaussian blur, and random color dithering (brightness 50%, contrast 50%, saturation 50%, hue 50%, and normalization). Contrast 50%, Saturation 50%, Hue 50%) and normalization. Finally, the image size is changed to 128*128 pixels, which effectively enhances the performance of the model in different brightness environments and under environmental disturbances. In terms of model selection and parameter settings, the model is trained with 40 epochs and the batch size is set to 32. ResNet-50 is used to extract the input image features for our backbone network. The use of *rmsprop* (root mean square prop) [53] as an optimizer can mitigate the problem of fast decreasing learning rate, and the introduction of root mean square can reduce the oscillation, this method is a good solution to the problem of premature ending in deep learning, and the learning rate is changed over time. The learning rate is set to 0.001 at the

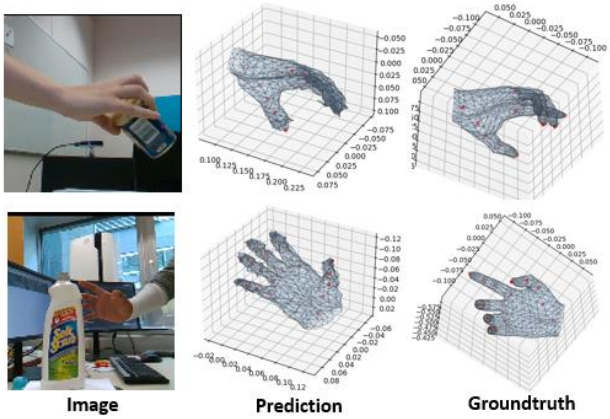


Fig. 6. The results of our model tested on the HO3D V2 dataset are visualized in the figure, the left side is the input image, the middle is the model prediction result, and the right side is the Ground truth

Table II：The results of the ablation experiments of the paper network on whether to use Deltamap and Locationmap in the intermediate morphological feature map section

Method	Pose error	Pose AUC	Mesh AUC	F-score at 5mm	F-score at 15mm
With Deltamap and Locationmap	**1.26**	**0.752**	**0.74**	**0.459**	**0.922**
Without Deltamap and Locationmap	1.40	0.72	0.7	0.414	0.9

Table III：The results of ablation experiments in the thesis network on the availability of Adjust part to assist in the correction of 3D hand mesh coordinates

Method	Pose error	Pose AUC	Mesh AUC	F-score at 5mm	F-score at 15mm
With Adjust part	**1.26**	**0.752**	**0.74**	**0.459**	**0.922**
Without Adjust part	1.58	0.68	0.68	0.375	0.86

Table IV：The results of thesis network ablation experiments on whether to use L_M2J to help network model training

Method	Pose error	Pose AUC	Mesh AUC	F-score at 5mm	F-score at 15mm
With all loss	**1.26**	**0.752**	**0.74**	**0.459**	**0.922**
Without L_{M2J}	1.37	0.73	0.71	0.425	0.903

Table V：The generalization experiments on HO3D-V2

Test set	Pose error	Pose AUC	Mesh AUC	F-score at 5mm	F-score at 15mm
FreiHAND	1.26	0.752	0.74	0.459	0.922
HO3D-V2	1.76	0.653	0.639	0.322	0.844

beginning, and then the learning rate is reduced by a factor of 10 at the 30th epoch. The final model generates a 2D heatmap of the hand, a hand mask, a hand mesh, and a 3D skeleton point of the hand.

C. Experimental results

As shown in Fig. 5., we tested the output of our model in the well-known public dataset FreiHAND, visualized it, and compared it with Ground truth, and the results show that our model can predict the 3D hand mesh very well. we can restore the hand shape and hand pose effectively. While the results on HO3D-V2 are shown in Table V and visualized in Fig. 6.

We conducted a comparative analysis of our results with those of other studies using the FreiHAND dataset. We conducted a comparative analysis of our results with those from other papers using the FreiHAND dataset. Despite our accuracy being lower compared to [48]-[49], it's crucial to highlight that their adoption of a transformer architecture led to a substantial increase in the number of parameters. To operate within the constraints of limited resources, we opted for an approach with a reduced parameter count.

When compared to other model-based papers employing parameters lower than those associated with transformer architectures, our findings showcase commendable performance and higher accuracy. Our ability to test the 3D pose directly generated and the hand mesh generated by the comparison model is shown in Table I, where Pose error is the Euclidean distance between the 3D skeleton point coordinates generated by our model and the 3D skeleton point coordinates of Ground truth, and Pose AUC is the area under the PCK curve within different error thresholds, and PCK is the percentage of correctly detected key points. PCK is the percentage of correctly detected critical points. F-score is the harmonic mean of recall and precision, we set the thresholds to 5 mm and 15 mm respectively.

D. Ablation Evaluation

To better demonstrate how the ideas mentioned in our paper help the network model and to show the magnitude of the impact of each module on the predictive power of the network model. In this subchapter, we test the network model proposed in the paper on the well-known public dataset FreiHAND. Table II shows the results of the ablation experiments on whether to use Deltamap and Locationmap in the intermediate representations part of the paper. This is good proof of the importance of using the intermediate representations to participate in the regression, which can effectively help the network to simplify the input and exclude the background interference, and the hand skeleton information can also effectively guide the network model to better understand the input image. Table III shows the results of the ablation experiments of the paper network with or without the Adjust part to assist in correcting the 3D hand mesh coordinates, from the results, we can see that the accuracy of the network model prediction of 3D hand mesh has increased significantly by adding the Adjust part to the network model. In the model-based approach of 3D hand mesh prediction based on monocular RGB image as input, the general tone of most approaches is to regress the MANO parameters and find ways to better regress the MANO parameters, some approaches use the GAN method for training, and some use different loss functions to enhance the training. However, my idea is to generate the hand model and then correct it afterward, because the model-base approach to 3D hand mesh prediction has the inherent drawback of having a strong base template constraint so that some special hand postures cannot be reconstructed well. At the same time, the approach we used in the Adjust part [55] has a very important result, where we add all the graph features, shallow features, deep features, skeleton point features, and 3D hand mesh features from the previous network to the crude hand mesh as an input to the GIN (Graph Isomorphism Network) network input. Here, two modules are used to transform the features in front of the network model: the local module and the global module. It is worth mentioning that the global module, unlike the local module which provides local features, provides overall features and uses the conv1D process

Table VI：The number of parameters of the neural network proposed in the paper

Method	Parameters
A+B+C	56.33M
A+B	43.45M
C	**12.88M**

to generate global features. We also tested if the graphical neural convolution network directly regressed the hand mesh instead of the offset, and we found that the regression of the offset was more accurate. Table IV shows the results of the ablation experiments of the paper network on whether to use L_{M2J} to help train the network model. Our paper includes ten loss functions, all of which are supervising the output of our network model during the runtime, we except the mesh to joint (L_{M2J}), where the supervised 3D hand skeleton point coordinates are generated by the 3D hand mesh generated by our network model. The mesh is defined by a MANO skeleton point regression matrix to obtain our 3D skeleton points here, and the results in Table IV show that the addition of this loss function contributes to the accuracy of our network model predictions.

Table VI shows the parameter scales for each part of the neural network proposed in the paper and the computational scales for each part of the neural network proposed in the paper, respectively. The number of parameters in the overall structure of our paper is 56.33 M, and the total computational volume is 21.8 G. The number of parameters in the Adjust part is 12.88 M, and the computational volume is 12.8 G. We find that the addition of the Adjust part makes the original network much larger, but we think it is worthwhile to improve the accuracy, and the focus of our network model architecture is to improve the accuracy of 3D hand mesh prediction, without considering the model size and computation speed as the most important measurement. The size of the model and the speed of computation are not considered the most important criteria.

V. Conclusions

In this paper, we propose a neural network that can accurately and efficiently predict 3D hand meshes from monocular RGB images. Our model consists of three main parts. The first Extract features part is to extract intermediate representations of the hand to facilitate the next parameter regression, playing a simplifying and guiding role. The second part, the MANO part, regresses the parameters of the parametric hand model MANO to produce a rough hand model first. The final part is the Adjust part, which generates the offset of the vertex coordinates on the hand mesh to correct our previous crude hand mesh and finally obtain the final hand mesh. We also demonstrate in Table II that the feature maps of the mesomorphs are important in the network. Table III demonstrates that our modified network is effective in improving the accuracy of our model. Compared to previous tasks, we have achieved better results. In the future, we will explore methods to reduce the model size and computational complexity without significantly affecting accuracy. W believe 3D hand mesh prediction will continue to be a very important field, and 3D hand mesh prediction will be indispensable for future games, VR and AR. These means more multimedia systems will cope with this key module in the near future.

Declarations

Competing Interests The authors declare that they have no conflict of interest.

Ethics Approval This is not relevant because this study didn't use any human or animal participants.

Consent to participate This is not relevant because no human or animal participants were used.

Consent to publication Publication didn't require consent because there were no human participants.

Funding
This research did not receive funding.

References

[1] C. Zimmermann and T. Brox, "Learning to Estimate 3D Hand Pose from Single RGB Images," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 4913-4921, doi: 10.1109/ICCV.2017.525.

[2] A. Spurr, J. Song, S. Park and O. Hilliges, "Cross-Modal Deep Variational Hand Pose Estimation," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 89-98, doi: 10.1109/CVPR.2018.00017.

[3] F. Mueller et al., "GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 49-59, doi: 10.1109/CVPR.2018.00013.

[4] P. Panteleris, I. Oikonomidis and A. Argyros, "Using a Single RGB Frame for Real Time 3D Hand Pose Estimation in the Wild," 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018, pp. 436-445, doi: 10.1109/WACV.2018.00054.

[5] Liuhao Ge, Hui Liang, Junsong Yuan, Daniel Thalmann, "Robust 3D hand pose estimation in single depth images: from single-view CNN to multi-view CNNs," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3593-3601

[6] L. Ge, H. Liang, J. Yuan and D. Thalmann, "Real-Time 3D Hand Pose Estimation with 3D Convolutional Neural Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 4, pp. 956-970, 1 April 2019, doi: 10.1109/TPAMI.2018.2827052.

[7] L. Ge, Y. Cai, J. Weng and J. Yuan, "Hand PointNet: 3D Hand Pose Estimation Using Point Sets," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 8417-8426, doi: 10.1109/CVPR.2018.00878.

[8] L. Ge, H. Liang, J. Yuan and D. Thalmann, "Robust 3D Hand Pose Estimation From Single Depth Images Using Multi-View CNNs," in IEEE Transactions on Image Processing, vol. 27, no. 9, pp. 4422-4436, Sept. 2018, doi: 10.1109/TIP.2018.2834824.

[9] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. J. Argus and T. Brox, "FreiHAND: A Dataset for Markerless Capture of Hand Pose and Shape From Single RGB Images," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 813-822, doi: 10.1109/ICCV.2019.00090.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep Residual Learning for Image Recognition," In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[11] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. J. Argus and T. Brox, "FreiHAND: A Dataset for Markerless Capture of Hand Pose and Shape From Single RGB Images," 2019 IEEE/CVF International Conference on

Computer Vision (ICCV), 2019, pp. 813-822, doi: 10.1109/ICCV.2019.00090.

[12] J. Y. Chang, G. Moon and K. M. Lee, "V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 5079-5088, doi: 10.1109/CVPR.2018.00533.

[13] P. Panteleris, I. Oikonomidis and A. Argyros, "Using a Single RGB Frame for Real Time 3D Hand Pose Estimation in the Wild," 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018, pp. 436-445, doi: 10.1109/WACV.2018.00054.

[14] L. Yang and A. Yao, "Disentangling Latent Hands for Image Synthesis and Pose Estimation," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 9869-9878, doi: 10.1109/CVPR.2019.01011.

[15] S. Li and D. Lee, "Point-To-Pose Voting Based Hand Pose Estimation Using Residual Permutation Equivariant Layer," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 11919-11928, doi: 10.1109/CVPR.2019.01220.

[16] Y. Cai et al., "Exploiting Spatial-Temporal Relationships for 3D Pose Estimation via Graph Convolutional Networks," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 2272-2281, doi: 10.1109/ICCV.2019.00236.

[17] Y. Chen, Z. Tu, L. Ge, D. Zhang, R. Chen and J. Yuan, "SO-HandNet: Self-Organizing Network for 3D Hand Pose Estimation With Semi-Supervised Learning," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 6960-6969, doi: 10.1109/ICCV.2019.00706.

[18] C. Wan, T. Probst, L. Van Gool and A. Yao, "Self-Supervised 3D Hand Pose Estimation Through Training by Fitting," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 10845-10854, doi: 10.1109/CVPR.2019.01111.

[19] S. Baek, K. I. Kim and T. -K. Kim, "Weakly-Supervised Domain Adaptation via GAN and Mesh Model for Estimating 3D Hand Poses Interacting Objects," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 6120-6130, doi: 10.1109/CVPR42600.2020.00616.

[20] L. Zhao, X. Peng, Y. Chen, M. Kapadia and D. N. Metaxas, "Knowledge As Priors: Cross-Modal Knowledge Generalization for Datasets Without Superior Knowledge," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 6527-6536, doi: 10.1109/CVPR42600.2020.00656.

[21] A. Spurr, J. Song, S. Park and O. Hilliges, "Cross-Modal Deep Variational Hand Pose Estimation," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 89-98, doi: 10.1109/CVPR.2018.00017.

[22] B. Doosti, S. Naha, M. Mirbagheri and D. J. Crandall, "HOPE-Net: A Graph-Based Model for Hand-Object Pose Estimation," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 6607-6616, doi: 10.1109/CVPR42600.2020.00664.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. "Attention is all you need," In NIPS, pages 5998– 6008, 2017

[24] Y. He, R. Yan, K. Fragkiadaki and S. -I. Yu, "Epipolar Transformers," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 7776-7785, doi: 10.1109/CVPR42600.2020.00780.

[25] Lin Huang, Jianchao Tan, Ji Liu, and Junsong Yuan. "HandTransformer: Non-autoregressive structured modeling for 3D hand pose estimation," In ECCV, pages 17–33, 2020

[26] Kan, Zhehan, et al. "Cross-Constrained Progressive Inference for 3D Hand Pose Estimation with Dynamic Observer-Decision-Adjuster Networks." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. No. 3. 2024.

[27] M. Schröder, J. Maycock, H. Ritter and M. Botsch, "Real-time hand tracking using synergistic inverse kinematics," 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 5447-5454, doi: 10.1109/ICRA.2014.6907660

[28] S. Fleishman, M. Kliger, A. Lerner and G. Kutliroff, "ICPIK: Inverse Kinematics based articulated-ICP," 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2015, pp. 28-35, doi: 10.1109/CVPRW.2015.7301345

[29] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin , "Real-time continuous pose recovery of human hands using convolutional networks," ACM Transactions on Graphics, 33(5):169:1–169:10, Sept. 2014

[30] M. Oberweger, P. Wohlhart and V. Lepetit, "Training a Feedback Loop for Hand Pose Estimation," 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 3316-3324, doi: 10.1109/ICCV.2015.379

[31] M. Oberweger, P. Wohlhart and V. Lepetit, "Generalized Feedback Loop for Joint Hand-Object Pose Estimation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 8, pp. 1898-1912, 1 Aug. 2020, doi: 10.1109/TPAMI.2019.2907951.

[32] K. Du, X. Lin, Y. Sun and X. Ma, "CrossInfoNet: Multi-Task Information Sharing Based Hand Pose Estimation," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 9888-9897, doi: 10.1109/CVPR.2019.01013.

[33] J. Malik et al., "DeepHPS: End-to-end Estimation of 3D Hand Pose and Shape by Learning from Synthetic Depth," 2018 International Conference on 3D Vision (3DV), 2018, pp. 110-119, doi: 10.1109/3DV.2018.00023.

[34] L. Ge, Y. Cai, J. Weng and J. Yuan, "Hand PointNet: 3D Hand Pose Estimation Using Point Sets," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 8417-8426, doi: 10.1109/CVPR.2018.00878

[35] J. Y. Chang, G. Moon and K. M. Lee, "V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 5079-5088, doi: 10.1109/CVPR.2018.00533

[36] Franziska Mueller, Micah Davis, Florian Bernard, Oleksandr Sotnychenko, Mickeal Verschoor, Miguel A Otaduy, Dan Casas, and Christian Theobalt, " Real-time pose and shape reconstruction of two interacting hands with a single depth camera," ACM Transactions on Graphics (SIGGRAPH), 38(4):49:1–49:13, 2019

[37] Jingjing Shen, Thomas J Cashman, Qi Ye, Tim Hutton, Toby Sharp, Federica Bogo, Andrew Fitzgibbon, and Jamie Shotton, "The phong surface: Efficient 3D model fitting using lifted optimization," In ECCV, pages 687–703, 2020

[38] Lixin Yang, Jiasen Li, Wenqiang Xu, Yiqun Diao, and Cewu Lu, "BiHand: Recovering hand mesh with multi-stage bisected hourglass networks," In BMVC, 2020

[39] A. Boukhayma, R. de Bem and P. H. S. Torr, "3D Hand Shape and Pose From Images in the Wild," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 10835-10844, doi: 10.1109/CVPR.2019.01110

[40] A. Kanazawa, M. J. Black, D. W. Jacobs and J. Malik, "End-to-End Recovery of Human Shape and Pose," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7122-7131, doi: 10.1109/CVPR.2018.00744.

[41] X. Zhang, Q. Li, H. Mo, W. Zhang and W. Zheng, "End-to-End Hand Mesh Recovery From a Monocular RGB Image," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 2354-2364, doi: 10.1109/ICCV.2019.00244.

[42] S. Baek, K. I. Kim and T. -K. Kim, "Pushing the Envelope for RGB-Based Dense 3D Hand Pose Estimation via Neural Rendering," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 1067-1076, doi: 10.1109/CVPR.2019.00116.

[43] Yuxiao Zhou, Marc Habermann, Weipeng Xu, Ikhsanul Habibie, Christian Theobalt, and Feng Xu ,"Monocular realtime hand shape and motion capture using multi-modal data," In CVPR, pages 5346–5355, 2020.

[44] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee,"Pose2Mesh: Graph convolutional network for 3D human pose and mesh recovery from a 2D human pose," In ECCV, pages 769–787, 2020.

[45] Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Otmar Hilliges, and Jan Kautz," Weakly supervised 3D hand pose estimation via biomechanical constraints," In ECCV, pages 211–228, 2020

[46] K. Lin, L. Wang and Z. Liu, "End-to-End Human Pose and Mesh Reconstruction with Transformers," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 1954-1963, doi: 10.1109/CVPR46437.2021.00199.

[47] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[48] LIN, Kevin; WANG, Lijuan; LIU, Zicheng. End-to-end human pose and mesh reconstruction with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021. p. 1954-1963.

[49] CHO, Junhyeong; YOUWANG, Kim; OH, Tae-Hyun. Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In: European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2022. p. 342-359.

[50] G. Pavlakos, L. Zhu, X. Zhou and K. Daniilidis, "Learning to Estimate 3D Human Pose and Shape from a Single Color Image," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 459-468, doi: 10.1109/CVPR.2018.00055

[51] A. Kanazawa, M. J. Black, D. W. Jacobs and J. Malik, "End-to-End Recovery of Human Shape and Pose," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7122-7131, doi: 10.1109/CVPR.2018.00744.

[52] J. Carreira, P. Agrawal, K. Fragkiadaki and J. Malik, "Human Pose Estimation with Iterative Error Feedback," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4733-4742, doi: 10.1109/CVPR.2016.512.

[53] X. Tang, T. Wang and C. -W. Fu, "Towards Accurate Alignment in Real-time 3D Hand-Mesh Reconstruction," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 11678-11687, doi: 10.1109/ICCV48922.2021.01149

[54] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka , "How powerful are graph neural networks? ," In ICLR, 2019

[55] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alch´e-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 32, pages 8026–8037. Curran Associates, Inc., 2019

[56] Tieleman, T., & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning, 4(2), 26-31.

[57] Y. Hasson *et al*., "Learning Joint Reconstruction of Hands and Manipulated Objects," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 11799-11808, doi: 10.1109/CVPR.2019.01208.

[58] WANG, Jiye; LUAN, Xiaoming. 3D Hand Pose Estimation and Shape Reconstruction based on Multi-task Learning. In: *2023 IEEE International Conference on Mechatronics and Automation (ICMA)*. IEEE, 2023. p. 7-12.

[59] Lim, G.M., Jatesiktat, P., Ang, W.T. (2020). "MobileHand: Real-Time 3D Hand Shape and Pose Estimation from Color Image," In: Yang, H., Pasupa, K., Leung, A.CS., Kwok, J.T., Chan, J.H., King, I. (eds) Neural Information Processing. ICONIP 2020. Communications in Computer and Information Science, vol 1332. Springer, Cham. https://doi.org/10.1007/978-3-030-63820-7_52