Journal of Computers Vol. 32 No. 5, 2021, pp. 1-14
doi:10.53106/199115992021103205001

3D Gesture Estimation from RGB Images Based on DB-InterNet

Ji-kai Zhang[1*], Jun Zhao[1], Qi Li[1], Xiao-qi Lv[2], Jun-lan Nie[3]

[1] School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou, Inner Mongolia, 014010, China
jkzhang0314@imust.edu.cn

[2] Institute of Information Engineering, Inner Mongolia University of Technology, Hohhot, Inner Mongolia, 010051, China
522507320@qq.com

[3] Institute of Information Science and Engineering, Yanshan University, Qinhuangdao Hebei, 066004, China
Niejl3@163.com

Received 29 January 2021; Revised 30 May 2021; Accepted 2 July 2021

Abstract. The booming of deep learning has made it possible to estimate 3D gestures from ordinary color images. However, the high accuracy of inferring 3D hand postures from RGB images is still not available due to the high flexibility of the gestures themselves. This paper aims to address the problem of low accuracy of keypoint coordinates position in InterNet gesture estimation network, by means of improving the confidence coordinate function, and selecting different suppression factor β according to different keypoints to fit t interacting hand estimation. By doing so, the maximum position coordinates are more precise and the keypoint prediction is more accurate. With respect to good measures, enhancing the representation capability of the model and employing dynamic activation function at different locations of the network to learn features in a dynamic way so as to boost the learning of hidden joints. In this way, different layers dynamically adjusted the segmental activation function according to the input to improve the performance of the model by dynamically learning features in a more flexible way. The experimental results indicate that compared with the baseline algorithm, the MPJPE and MRRPE of this algorithm are reduced by 0.73% and 2.04%, respectively, and the accuracy of single hand estimation is higher while the accuracy of interacting hand estimation is also effectively improved.

Keywords: gesture estimation, InterNet network, dynamic activation function, confidence coordinate function

1 Introduction

3D gesture estimation of the hand has been a long-standing research content in computer vision field and plays an important role in many applications, including human-computer interaction and virtual reality. Due to the complex structure and dexterous motion of the hand, the accurate estimation of its gesture has been a challenge. In recent years, as hardware and artificial intelligence techniques develop considerably, a variety of data glove prototypes and computer vision-based methods have been proposed for more accurate and faster gesture estimation. As deep learning techniques prosper rapidly, they were applied to gesture estimation field by researches and yielded fruitful breakthroughs.

The practical significance of hand pose estimation is that it opens the door for gesture-based human computer interactions (HCIs). The hand skeleton information provided by gesture estimation can effectively reflect the hand motion in the domain of space and time, thus cultivating a good condition for

* Corresponding Author

motion interaction. Hence, it is of great practical importance for applications such as immersive virtual reality (VR) and augmented reality (AR). Currently, data gloves [1-2] that can accurately capture hand motion in real time which are not highly practical as they are expensive and the measurement results require complex calibration and setup procedures. Another approach to capture hand motion is a vision-based one that locates hand joints in 3D space directly from the input hand image and provides a non-contact, naturalistic interaction experience in the absence of expensive and complex hardware devices. Among vision-based gesture estimation methods, a slew of algorithms utilize depth cameras for gesture 3D pose estimation [3-5] with some results, but the limitations of the depth sensor principle lead to its inapplicability to all scenarios. Given the wide availability of RGB cameras, RGB gesture-based 3D pose estimation solutions are more favorable over depth-based solutions in many vision applications.

Since the RGB image contains less information than depth disparity maps, the RGB-based network is harder to train and requires a larger dataset. The current RGB image-based 3D gesture estimation is perplexed by many problems in which the most important issue in image-based method is the occlusion case that when the hand itself occludes some parts of the hand. Additionally, the similarity between fingers contributes to the difficulties in distinguishing them. Therefore, this paper concentrates on the inaccurate keypoint location estimation caused by the hand similarity and invisibility of hidden keypoints due to occlusion.

This paper proposes to estimate 3D hand joint locations directly from single RGB image based on the network architecture of InterNet. It avails of dynamic activation function [6] and confidence coordinate function to overcome self-similarity and occlusion. Owing to the method based on dynamic activation function [6] and confidence coordinate function, this approach is named DB-InterNet.

The main contributions of this paper are as follows:

1. The pipeline of the proposed system consists of three modules. First, a hand detection network (DetectNet) detects the bounding boxes of hand in an input image. Second, the proposed 3D hand root localization network (RootNet) estimates the camera-centered coordinates of the detected hands' roots. Third, a root-relative 3D hand pose estimation network (PoseNet) estimates the root-relative 3D pose for each detected hand. The DB-InterNet network predicted both 2D and 3D hand joint positions, employing 3D joint information in compensation for the depth blur in 2D joint localization, in a bid to surmount the inaccurate estimation of hidden keypoints of gestures caused by occlusion.

2. To address the problem of inaccurate keypoint location estimation caused by the similarity of hand appearance, the root joint and other joints are distinguished by the soft-max function, and the β-Soft-Argmax function [7] is used to obtain the relative position coordinates of the root joint and other keypoints to enhance the accuracy of gesture estimation and curtail the joint position error.

3. To cope with the problem of invisibility of hidden keypoints caused by occlusion, the dynamic activation function is used in the network to determine the corresponding activation function according to the input, and the image features are dynamically learned to promote the model's representation capability.

This paper introduces related works in the second section; the introduction of 3D gesture estimation network in the third section; the experimental results and analyses in the fourth section; the conclusion and prospect in the fifth section.

2 Related Work

2.1 3D Gesture Estimation Based on Depth Image

Depth-based methods are employed for gesture estimation with a depth camera providing synchronized RGB video and depth map as data. Some depth-based methods [8-10], where RGB video is only auxiliary data and hand segmentation is completed by skin detection, mainly use the depth map to extract gesture data. Wan et al. [11] proposed a self-supervised system for 3D gesture estimation based on the depth map, initializing the neural network with synthetic data and fine-tuning the unlabeled depth map to improve the accuracy of gesture estimation. The JGR-P2O algorithm [12] models the complex dependencies between joints on the basis of the joint graph inference module, estimating the offsets of joint pixels in the image plane and depth space and calculating joint positions with the help of the weighted averaging of all the predicted pixels. Ge et al. [13] used 3DCNNs [4] and PointNet [14-15] to directly estimate 3D hand positions, using 3D volume representation of hand depth image and 3D point

cloud as input to regress the joint heatmap, respectively, but it required complex post-processing. Sinha et al. [16] trained a separate network for each finger using a depth map-based approach to regress the three joints on each finger, but removed the other pixels contained in the hand shape cropping frame. Moon et al. [17] designed a detection-based voxel-to-voxel 3D network that takes the voxel representation of one hand as input and outputs a 3D heatmap of each joint. Although depth image-based gesture estimation methods are favored for its good preservation of shape information and insensitivity to cluttered backgrounds, they are not widely used on account of its high energy consumption, poor near coverage accuracy, and poor outdoor performances.

2.2 3D Gesture Estimation Based on RGB Images

RGB-based 3D gesture estimation uses RGB images as input to train the model. Although this approach endows the model with favorable generalization, the dimension input sees a reduction from 2.5D to 2D, which makes tasks more difficult. Therefore, RGB-based networks are more difficult to train and require a larger dataset for support. Zimmermann et al. [18] input a single RGB image into a segmentation network to segment the hand mask and crop the gesture image, and then pass it to a gesture network in view to locate gesture nodes, and finally convert 2D joint prediction to 3D gesture estimation. Simon et al. [19] employed a multi-camera setup to estimate hand pose by shooting from different angles to tackle self-occlusion and viewpoint occlusion of joints, annotating the newly generated dataset with multiple views, and iteratively improving the accuracy of the gesture estimation through detector. Cai et al. [20] argued that it should employ RGB images as input for 3D gesture prediction and implicitly reconstruct the depth map, and use the depth map as a weak supervision for 3D pose regression to estimate 3D hand joint coordinates. Panteleris et al. [21] detected gestures in images and input to open pose network [19] to locate 2D key points using detection algorithm, and finally adopted model fitting method to calculate 3D gestures. Yang et al. [22] proposed the depth generative model to learn potential space of 2D gestures and estimate 3D poses of gestures by decoding samples of potential space.

The above methods have certain limitations in 3D hand pose estimation from a single RGB image, such as requiring additional depth map or multi-view images or taking a single depth map and not a single RGB. Compared to above approaches, this paper uses InterHand2.6M dataset including a variety of real-captured RGB images. This method can perform 3D single and interacting hand pose estimation simultaneously from a single RGB image. In contrast with [27], within the InterNet framework, this paper adds dynamic activation function and confidence coordinate function. In order to predict coordinates more accurately, it applies a soft-max function to distinguish the root joint from other joints, and then obtained other keypoints relative depth value to the root joint.

This paper conducts comprehensive experiments on the three publicly available hand pose datasets. Experimental results reflect that this proposed method can achieve superior accuracy performance on 3D hand pose estimation, compared with state-of-the-art methods.

3 3D Gesture Estimation Based on DB-InterNet Network

In DB-InterNet network, a single RGB image served as input, and the cropped image adjusted to a uniform resolution for input to the sub-network for gesture detection and root joints coordinate prediction. This paper uses the β-Soft-Argmax function to improve the accuracy of gesture estimation and the dynamic activation function to enhance the model's representation capability. The final 2D and 3D hand gestures are estimated based on the 2D and 3D positions of the keypoints of gestures.

3.1 Gesture Estimation Network

In the gesture estimation network stated in this paper, after inputting an RGB image, the gesture category could be predicted, and the gesture's 3D pose is reconstructed using a 2.5D heatmap [23]. 2.5D heatmap is composed of 2D heatmap of keypoints and depth map of keypoints, extracting the 2D coordinates of keypoints on the 2D heatmap using β-Soft-Argmax function, and fusing the depth map and 2D pose to recover the 3D pose.

The DB-InterNet network consisted of three sub-networks, DetectNet, RootNet and PoseNet. The DetectNet sub-network detected the bounding box of each hand in the input image using a target detection network. Then the RootNet network obtained the cropped and resized hand gesture image from the DetectNet network, and the hand gesture depth value and 2D image coordinates are back-projected to the camera-centered coordinate space to locate the 3D coordinates of the root joint; among that the correction factor γ of the image and the absolute depth k from the camera to the object are calculated in the same way as in the literature [24]. Finally, the PoseNet network adopted the model of Sun et al. [25] to calculate the 2D and 3D positions of other keypoints using the cropped images of DetectNet and the coordinates of the RootNet root node to estimate the final 2D and 3D hand pose. The overall pipeline of DB-InterNet is shown in Fig. 1.

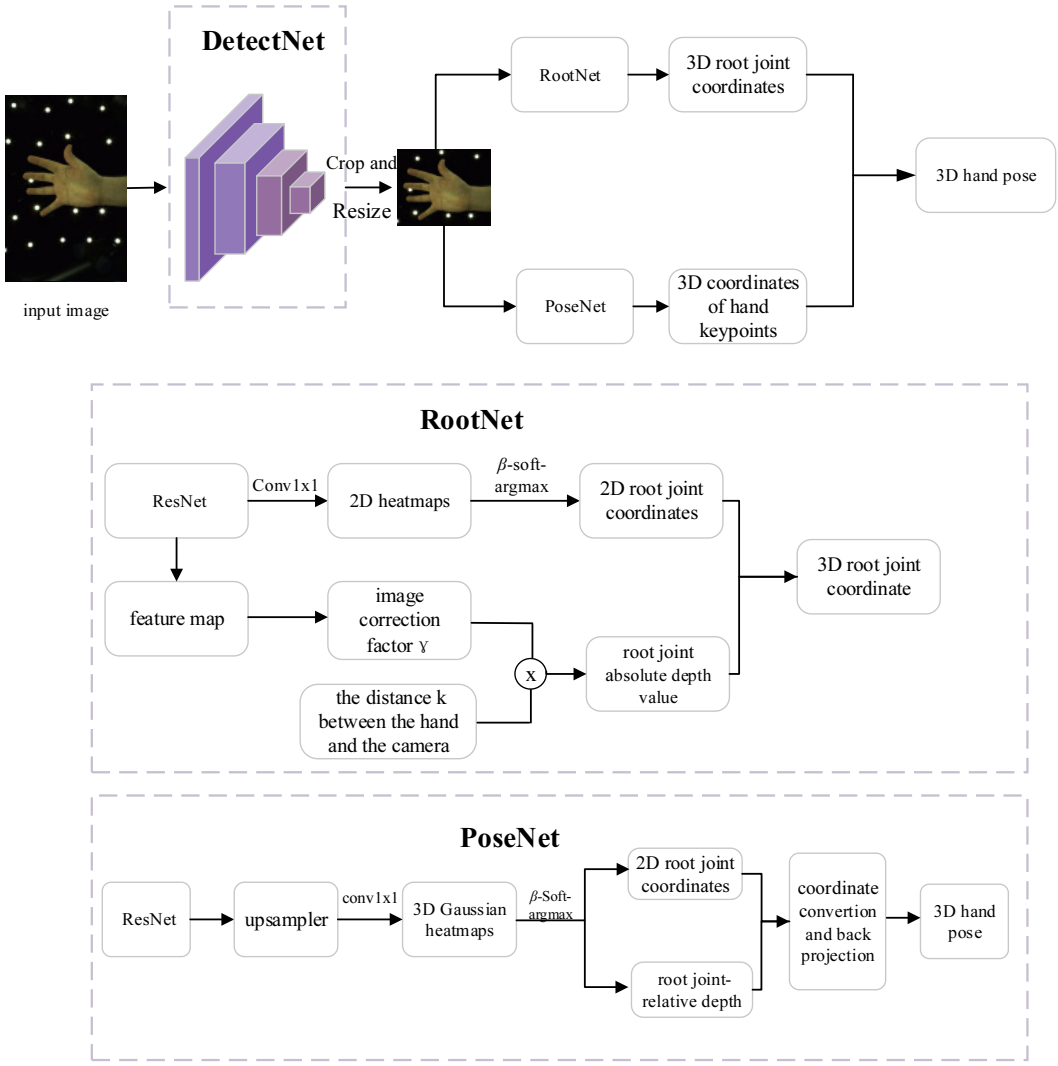


Fig. 1. The overall pipeline of the DB-Internet

DB-InterNet takes a single RGB image as an input and extracts the image feature using ResNet with its fully-connected layers trimmed. It prepares I by cropping the hand region from an image and resizing it to uniform resolution. From image feature, DB-InterNet predicts the gesture class, 2.5D left and right hand poses, and relative depth of right hand to left hand according to the extracted image features. The 2.5D hand pose comprises 2D pose in *x*-axis and *y*-axis and root joint (i.e., wrist)-relative depth in *z*-axis, widely used in state-of-the-art 3D human body and hand pose estimation from a single RGB image. The output of its network structure consists of three parts. The first part is gesture classification, constructing two fully connected layers to extract image features, each of which is followed by dynamic ReLU activation functions except the last one; after that, the probabilities of left hand, right hand, and interactive hand are estimated using sigmoid functions. The second part is 2.5D left and right hand pose

consisting of a 2D pose in *x*- and *y*-axis and root joint-relative depth in *z*-axis; the estimation of the 2.5D left and right hand pose was based on extracting image features through the upsampler containing a deconvolutional layer, a batch normalization layer, a dynamic ReLU activation function and a convolutional layer to generate a 3D Gaussian heatmap of the joints, where the structure of the upsampler module is shown in Fig. 2. Each voxel of the 3D Gaussian heatmap of the joint j represents the likelihood of the existence of a hand joint j in that position. The third part is the right hand-relative left hand depth; the estimation of the relative depth of the root joint of left is based on the absolute depth of the root joint of right, and the estimation of the relative depth of other keypoints is based on the right and left root joint. Finally, the 2D image coordinates (x_j, y_j) and the relative depth values of the left wrist are extracted using the β-Soft-Argmax function, and the results of gesture estimation are obtained by coordinate transformation.

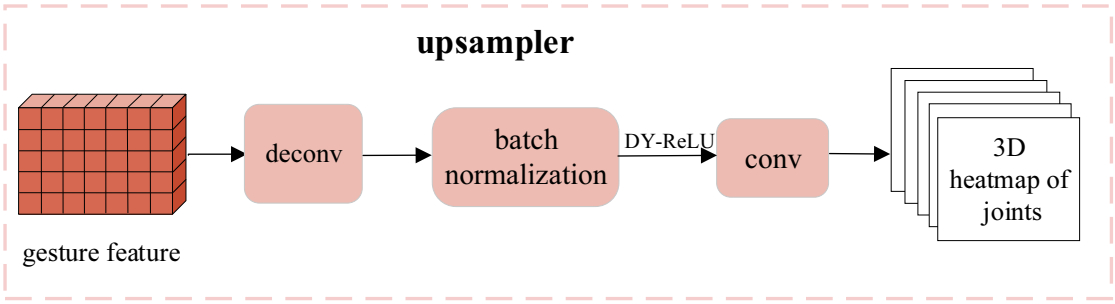


Fig. 2. The structure of the upsampler module

In this paper, the DB-InterNet gesture estimation network is trained with a multi-task loss function, and the loss function L is the sum of the classification loss function L_h, the 2.5D left-and-right-hand gesture loss function L_{pose} and the relative depth loss function L_{rel}, defined as presented in equation (1).

$$L = L_h + L_{pose} + L_{rel}. \tag{1}$$

The gesture classification loss function L_h uses a binary cross-entropy loss function to boost the probability of the gesture category predicted by the network, as defined in equation (2), where Q is the gesture, δ^Q the probability of the gesture classification predicted by the network, and h^Q the probability of the labeled gesture classification.

$$L_h = -\frac{1}{2}\Sigma_{Q\in(R,L)}(\delta^Q \log h^Q + (1-\delta^Q)\log(1-h^Q)). \tag{2}$$

The 2.5D left and right gesture loss function L_{pose} trained the joint heatmap by minimizing the L_2 distance loss function between the estimated heatmap and groundtruth heatmap, defined as in Eq. (3), where $H^Q_{2.5D}$ is the predicted 3D Gaussian heatmap of the hand joint and $H^{Q_*}_{2.5D}$ the labeled 3D Gaussian heatmap. If the input image do not contain a hand, its loss would be set to zero.

$$L_{pose} = \Sigma_{Q\in(R,L)} \| H^2_{2.5D} - H^{Q_*}_{2.5D} \|. \tag{3}$$

L_{rel} trained the relative root depth by minimizing the estimated L_1 distance loss function between the depth of the left hand relative to the right hand and labeled depth to locate left hand joint position. L_{rel} is shown in equation (4), where $Z^{R\to L}$ is the relative depth value and $Z^{R\to L_*}$ the labeled depth value. When there is only one hand in the input image, the loss is set to zero.

$$L_{rel} = | Z^{R\to L_*} - Z^{R\to L} |. \tag{4}$$

3.2 Dynamic ReLU Activation Function

Currently, numerous neural network models adopt the ReLU activation function to introduce nonlinearity to make the performance of forward networks more superior. To address the problem that the static activation function ReLU performs exactly the same way for different inputs, in this paper, dynamic

ReLU [6] is adopted to substitute for the traditional ReLU function embedded in the model stated in this paper and appropriately tackle the invisibility of hidden keypoints owing to occlusion for the purpose of a better representation model. This function dynamically adjusted the segmentation function to determine the appropriate activation function by encoding the global context of the input, giving it greater representational power compared with a static model. In short, dynamic ReLU adaptively learned the corresponding activation function for the sample, which significantly improved the performance of the model. Dynamic ReLU refers to as DY-ReLU, and the function image is shown in Fig. 3. It adapts slopes and intercepts of two linear functions in ReLU through the learned function of inputs. Therefore, this paper inputs some challenge images to boost the representation capability of the network.

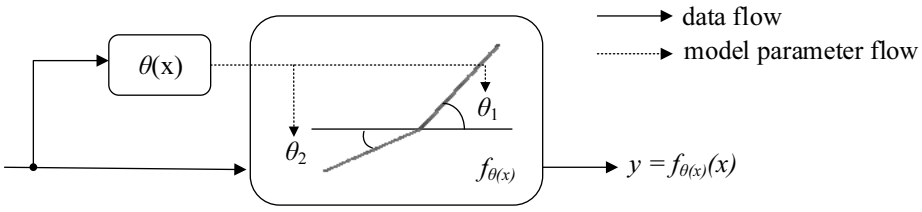


Fig. 3. Function image of DY-ReLU

DY-ReLU is a parametric segmentation function consisting of $\theta(x)$ that calculates the parameters of the activation function and activation function $f_{\theta(x)}(x)$ that calculates the output. Parameter $\theta(x)$ that can be learned can adapt to and input x through a dynamic linear transformation. The definition of $\theta(x)$ is shown in equation (5).

$$\theta(x) = [a_1^1, ..., a_c^1, ..., a_1^K, ..., a_c^K, ..., b_1^1, ..., b_c^1, ..., b_1^K, ..., b_c^K]^T. \tag{5}$$

Where the coefficients a_c^k and b_c^k are the outputs, the slope $a_c^k(x)$ and intercept $b_c^k(x)$ are calculated as is shown in equations (6) and (7).

$$a_c^k(x) = a^k + \lambda_a \Delta a_c^k(x), \tag{6}$$

$$b_c^k(x) = \gamma^k + \lambda_b \Delta b_c^k(x). \tag{7}$$

Where α_c^k and γ_c^k are the initialized values of slope $a_c^k(x)$ and intercept $b_c^k(x)$ intercepted, respectively, and λ_a and λ_b are the hyperparameters regulating the slope and intercepted to stay within a certain range so that the function is neither too wide nor too narrow. K is the number of functions, and C the number of channels.

For a given input vector x_c of the c_{th} channel, $f_{\theta(x)}(x)$ used the coefficients a_c^k and b_c^k output by function $\theta(x)$ to adjust segmented linear function to fit the input element $x = \{x_c\}$, as defined in equation (8).

$$y_c = f_{\theta(x)}(x) = \max_{1 \le k \le K} \{a_c^k(x) + b_c^k(x)\}. \tag{8}$$

In this paper, the DY-ReLU function is used in the feature extraction network ResNet, after the fully connected layer in gesture classification, the convolutional layer of upsampler in 2.5D left and right gesture estimation. The number of linear functions is set to K=2, the initialized values of slope and intercept are set to $\alpha^1 = 1, \alpha^2 = \gamma^1 = \gamma^2 = 0$, and the ranges of slope and intercept are set to $\lambda_a = 1$ and $\lambda_b = 0.5$ in the experiment, respectively. The coefficients of the segmented linear function generated by function $\theta(x)$ are employed to dynamically adjust the activation function to the corresponding input, and strengthen the representation capability of the network, and thus obtaining better performance.

3.3 Confidence Coordinate Function

Currently, related methods principally use Argmax function and Soft-Argmax function [25-26] to generate heatmap of each keypoint by Gaussian function, and the heatmap to obtain the confidence level of the maximum position and then calculate the final coordinates. However, the results of Argmax are discrete and integer-only, inhabiting the accuracy of the final coordinates. Differentiable as the Soft-Argmax function is, there will be a large number of values close to 0 in the predicted heatmap with the normalization of the real heatmap. This will reduce the probability of the maximum value occurring, thus affecting the accuracy of keypoint prediction.

To address the inaccurate estimation of keypoint locations caused by the difficulty in distinguishing due to similarities, in this paper, the soft-max function was employed to distinguish the root joint from other joints, and then the locations of the root joints are obtained and applying other keypoints relative to the root joint by the β-Soft-Argmax function to make the prediction of other keypoints more accurate. β-Soft-Argmax optimizes the formula of Soft-Argmax by adding a coefficient β before the heatmap $H_k(x, y)$ to suppress the effect of the values close to zero. By adding a factor β before the heatmap, the β-Soft-Argmax function suppressed the influence of a large number of zero values on the accuracy of keypoint prediction, and increased the relative maximum value while attenuated the influence of other values to obtain more accurate prediction of the maximum position coordinates. The confidence coordinate function is shown in equation (9), where (x_k, y_k) and $H_k(x, y)$ are the 2D coordinates and the labeled heatmap of the kth keypoint, respectively. Same as Soft-Argmax, the final coordinates are obtained through $\widehat{x_k} = \Sigma S_k \circ W_x$, $\widehat{y_k} = \Sigma S_k \circ W_y$, where W_x and W_y are constant weight matrixs and $\circ$ means element-wise multiplication.

$$S_k(x, y) = \frac{e^{\beta H_K(x, y)}}{\Sigma_x \Sigma_y e^{\beta H_K(x, y)}}, (\beta > 1). \tag{9}$$

To improve the accuracy of the predicted keypoint coordinates and fit the prediction of the root joint of the left hand relative to the right hand in this paper, the best effect in the experiment is found when $\beta = 160$. When β is greater than 160, the improvement of the network tends to be saturated. Consequently, in this paper, the heatmap is classified and identified by softmax function, and β is set to 1 when the keypoint is a point other than the relative root keypoint. Although accurate gesture estimation could be achieved using InterNet network [27], to pursue more accurate keypoint location prediction, in this paper, β-Soft-Argmax post-processing function is added to DB-InterNet network, so as to suppress the situation of a large number of zero values in the heatmap and augment the relative maximum value. On top of that, corresponding values are set according to different types of keypoints to make them continuous and get more accurate coordinate prediction results, in a bid to make the model more accurate.

4 Experimental Results and Analysis

4.1 Dataset and Evaluation Metrics

This method is evaluated on three publicly available datasets: the InterHand2.6M dataset, the RHD dataset [18] and the STB dataset [28]. InterHand2.6M dataset with 512 × 334 resolution is annotated with the rotation centers of fingertips and three joints. This dataset are annotated 20 keypoints of each hand and the wrist rotation centers. InterHand2.6M consists of large-scale real-captured RGB images and a variety of sequences. This paper divides the dataset into a training set with 23082 images, a validation set with 826 images and a test set of 2500 images.

RHD is a synthesized color image dataset by software under different lightings, backgrounds and camera views with 320×320 resolution, composed of 41238 samples for training and 2728 samples for testing. This dataset is highly challenging due to the low resolution and the large variations of viewpoints.

STB [28] includes 6 pairs of stereo sequences of diverse poses with different backgrounds from a single person. It was taken in an indoor environment with 18,000 images in total, each of which included 21 labeled 3D hand joint coordinates. The dataset with 640 × 480 resolution contains 15,000 training images and 3,000 test images.

It evaluates the performance of 3D hand pose estimation with four metrics:(i) MPJPE [29]: the average errors of each joint position is to calculate the Euclidean distance between the predicted keypoint and the labeled 3D joint position by centering on the coordinates of the root joint in the camera coordinate system. It measures the accuracy of 3D joint pose estimation.(ii) MRRPE: the average of the relative root position coordinate errors predicts the Euclidean distance between the predicted right hand root position and the labeled right hand relative to the left hand root position, and this metric measures the accuracy of the prediction of the right hand relative to the left hand joint position.(iii) 3D PCK: the percentage of the predicted keypoint is falling within a given radius sphere relative to the labeled keypoint, and this metric measures the accuracy of the predicted keypoint.(iv) AUC: area under the curve (AUC) on the percentage of correct keypoints (PCK) curve. AUC represents the percentage of predicted keypoints that fall within certain errors thresholds compared with ground-truth poses.

4.2 Experiment Details

The method is implemented within the Pytorch framework. The network are trained using the Adam optimizer to update the weights with a batch size of 16. For training on the InterHand2.6M dataset, the input image resolution is adjusted to 256 × 256, the initial learning rate is set to 10^{-4}, and the training is performed for 20 epochs, with a 10-fold drop in epochs 15 and 17, respectively. When training on the RHD dataset, the initial learning rate is also set to 10^{-4} for 50 epochs, with a 10-fold dip at epochs 45 and 47, respectively. This paper runs the all experiments on a dual GPU workstation with Nvidia Tesla V100.

4.3 Gesture Estimation Results and Analysis

4.3.1 Comparison of MPJPE and MRRPE

In this paper, the impact of activation function and confidence coordinate function on the network performance were first evaluated, both using resnet-50 as the feature extraction network. The experiments made comparisons in the following three aspects: (1) comparison of different activation functions; (2) setting the value of the confidence coordinate function β, β is set to 160 for InterNet-β_1, to 1 for InterNet-β_2 to adjust the coordinates of the root joint location of the gesture and to 160 to adjust the coordinates other than the root joint; (3) using both the DY-ReLU activation function and the confidence coordinate function. The four experimental parameters are consistent and evaluate on the InterHand2.6M dataset, and the comparison results are shown in Table 1.

Table 1. The results of gesture estimation between the benchmark experimental model and our method

Algorithm	Activation Function	Confidence Coordinate Function	MPJPE	MRRPE
InterNet	ReLU	Soft-Argmax	14.22	32.57
InterNet-Dy	DY-ReLU	Soft-Argmax	14.04	30.66
InterNet-β_1	ReLU	β-Soft-Argmax	14.00	32.61
InterNet-β_2	ReLU	β-Soft-Argmax	13.98	30.65
DB-InterNet	DY-ReLU	β-Soft-Argmax	13.49	30.53

The approach was applied to the InterHand2.6M dataset, currently the largest dataset for hand pose estimation from single color images. As indicated in Table 1, it finds that using either the activation function or the confidence coordinate function alone can improve the accuracy of the network and reduces the error, but using these two simultaneously works best. The reduction of both MPJPE and MRRPE using the DY-ReLU function over the static ReLU method is due to the dynamic use of segmented linear activation functions for each input algorithm, boosting the representation and reinforcing the learning of hidden keypoints.

Based on the comparison of the impact of β-Soft-Argmax, it inferred that the InterNet-β_1 algorithm falls MPJPE when β =160 because the predicted keypoint location coordinates are more accurate, while MRRPE is higher than the baseline algorithm because the relative position of the relative root increased after adding a factor β to the coordinates of the relative root, leading to an increase in the relative coordinate error of the left hand. As a consequence, InterNet-β_2 set different β to adjust the relative root and other joint points, the coordinates of the right hand relative to the left hand remained unchanged, and the heatmaps of other keypoints is multiplied by the factor β = 160 to enhance the accuracy of the

position coordinate prediction. As can be seen from Table 2, the error of InterNet-β_2 is lower than that of InterNet-β_1. The main reason is that setting the value of the suppression factor β separately is beneficial to the accuracy of the joints location estimation. Since β =160 and β =1 are the most accurate for the location coordinates estimation of keypoints, β =1 and β =160 are used for the experiments in the later sections.

Table 2. Single and interacting hand MPJPE comparison from models

Algorithm	Activation Function	Confidence Coordinate Function	Single MPJPE	Interacting MPJPE
InterNet	ReLU	Soft-Argmax	12.16	16.02
InterNet-Dy	DY-ReLU	Soft-Argmax	12.08	15.76
InterNet- β	ReLU	β-Soft-Argmax	11.72	15.11
DB-InterNet	DY-ReLU	β-Soft-Argmax	11.63	14.99

As shown in the last one row of Table 1, when the DY-ReLU function and β-Soft-Argmax function are added to the network at the same time, both MPJPE and MRRPE are reduced so as to further improve the network representation capability while outputting more accurate predictions. The experiments indicate our proposed DY-ReLU function and the set value of β can further improve the estimation accuracy of joint locations.

To investigate the benefits of DY-ReLU function and β-Soft-Argmax function for 3D interacting hand pose estimation, this paper makes comparisons of single and interacting hand MPJPE of our method trained with and without interacting hand data in Table 2. It uses resnet-50 as the feature extraction network, and the experimental comparison results are shown in Table 2.

It observes that DY-ReLU function and β-Soft-Argmax function improved not only interacting hand pose estimation performance, but also single hand pose estimation. These comparisons clarified the benefits of the newly introduced functions for 3D single and interacting hand pose estimation.

After the careful observation, it is apparent that the accuracy of gesture estimation of interacting hand is lower than single hand, and interacting hand impeded the correct prediction of keypoints to some extent. The estimation performance of the network for single and interacting hand is further improved using the DY-ReLU function and the β-Soft-Argmax function. It is also found that the improved network offered the estimation of interacting hand in a more precise way, indicating that the DY-ReLU function and β-Soft-Argmax enhanced both the estimation accuracy of the single hand and occluded keypoints of interacting hands.

In this work, DY-ReLU dynamic learning feature is thought to strengthen the representation ability of the model and enhance the learning ability of the network for the occluded keypoints. β-Soft-Argmax makes the keypoint coordinates prediction more accurate, both of which promote the learning of the network at the same time.

To further evaluate our model, analyses are conducted to the functions of our model on different deep network. Table 3 shows the effects on the deep network models. By comparison, both β-Soft-Argmax and DY-ReLU improved the network models of different depths to some certain extent. The accuracy of choosing resnet-101 as the backbone model outstripped that of using resnet-50, and the accuracy achieved by the DB-InterNet algorithm with resnet-101 as the backbone is the highest, indicating that both DY-ReLU and β-Soft-Argmax jointly and separately improved the accuracy of the network models at different depths. Considering our DB-InterNet achieves state-of-the-art performance on publicly available dataset [27], it is concluded that activation function and β-Soft-Argmax are critically important for better performance on different deep network models.

Table 3. Gesture estimation accuracy of models with different depths

Algorithm	Backbone	Activation Function	Confidence Coordinate Function	MPJPE	MRRPE
InterNet	R-50	ReLU	Soft-Argmax	14.22	32.57
InterNet	R-50	DY-ReLU	Soft-Argmax	14.04	30.66
InterNet	R-50	ReLU	β-Soft-Argmax	13.98	30.65
DB-InterNet	R-50	DY-ReLU	β-Soft-Argmax	13.49	30.53
InterNet	R-101	ReLU	Soft-Argmax	13.46	30.95
InterNet	R-101	DY-ReLU	Soft-Argmax	13.27	30.63
InterNet	R-101	ReLU	β-Soft-Argmax	13.13	30.33
DB-InterNet	R-101	DY-ReLU	β-Soft-Argmax	12.92	30.28

It makes comparisons of the performance of our DB-InterNet with Internet 3D hand pose estimation method on the STB and RHP in Table 4. The backbone network use resnet-50. As the RHD and STB datasets are annotated with only one hand, the value of β is set to 160. When both hands are absent, there is no error MRRPE of the relative root position coordinate. As a result, the metric of MRRPE has not been used. The Table 4 shows the proposed DB-InterNet outperforms previous method. In contrast, the accuracy is improved using the β-Soft-Argmax function alone and is higher than that of DY-ReLU alone. And the effect of β-Soft-Argmax on coordinate accuracy is somewhat greater. Using DY-ReLU and β-Soft-Argmax simultaneously is conducive to the reduction of MPJPE, which indicates that our DB-InterNet is applicable to different data sets.

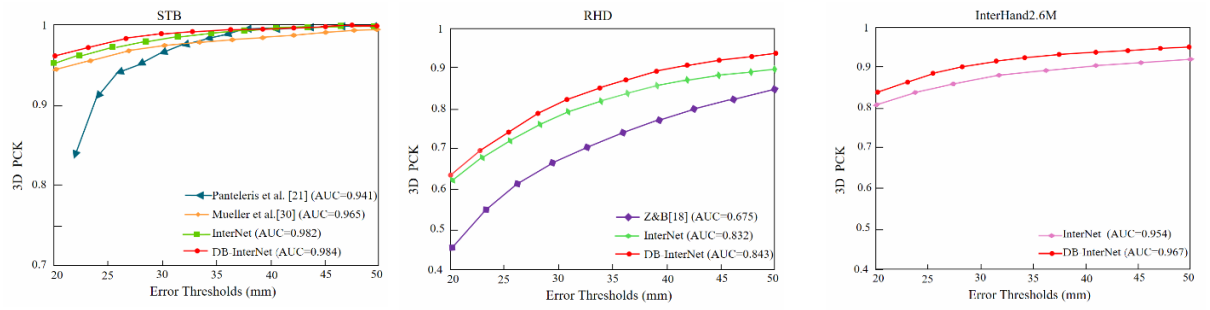
Table 4. Comparison of MPJPE on the RHD and STB dataset

Algorithm	Activation Function	Confidence Coordinate Function	MPJPE	
			RHD	STB
InterNet	ReLU	Soft-Argmax	20.89	7.95
InterNet-Dy	DY-ReLU	Soft-Argmax	20.84	7.92
InterNet-β	ReLU	β-Soft-Argmax	20.80	7.91
DB-InterNet	DY-ReLU	β-Soft-Argmax	20.78	7.90

On account of the simplicity of the background content and gestures of the STB dataset, the performance tests of many advanced methods on this dataset have tended to be saturated. Although the accuracy of our algorithm on the STB dataset experiences slight improvement over the baseline algorithm, the benefits of our method are more visible on the STB.

4.3.2 Comparison of 3D PCK

It also compares the PCK curve of the approach with other state-of-the-art methods in Fig. 4. The area AUC under the percentage PCK curve of correct keypoints outperforms the state-of-the-art [21, 27, 30] on the STB dataset, whereas on the RHD dataset, It surpasses state-of-the-art [18] with a significant gap. Compared with the baseline, this method achieves the highest AUC value on the 3D PCK. This means that the incorporated dynamic activation function and position coordinate function can improve the results of 3D gesture estimation and reduce the error and improve the performance.



(a) 3D PCK on STB dataset (b) 3D PCK on RHD dataset (c) 3D PCK on InterHand2.6M dataset

Fig. 4. PCK curve of our model on datasets for RGB to 3D

4.3.3 Qualitative Results

This paper trains the networks on different datasets and test the model. Some qualitative results of 2D and 3D single hand pose estimation for InterHand2.6M, RHD, and STB datasets are shown in Fig. 5. The proposed approach can handle occlusions, complex hand articulations and achieve good performance for joint pose estimation.

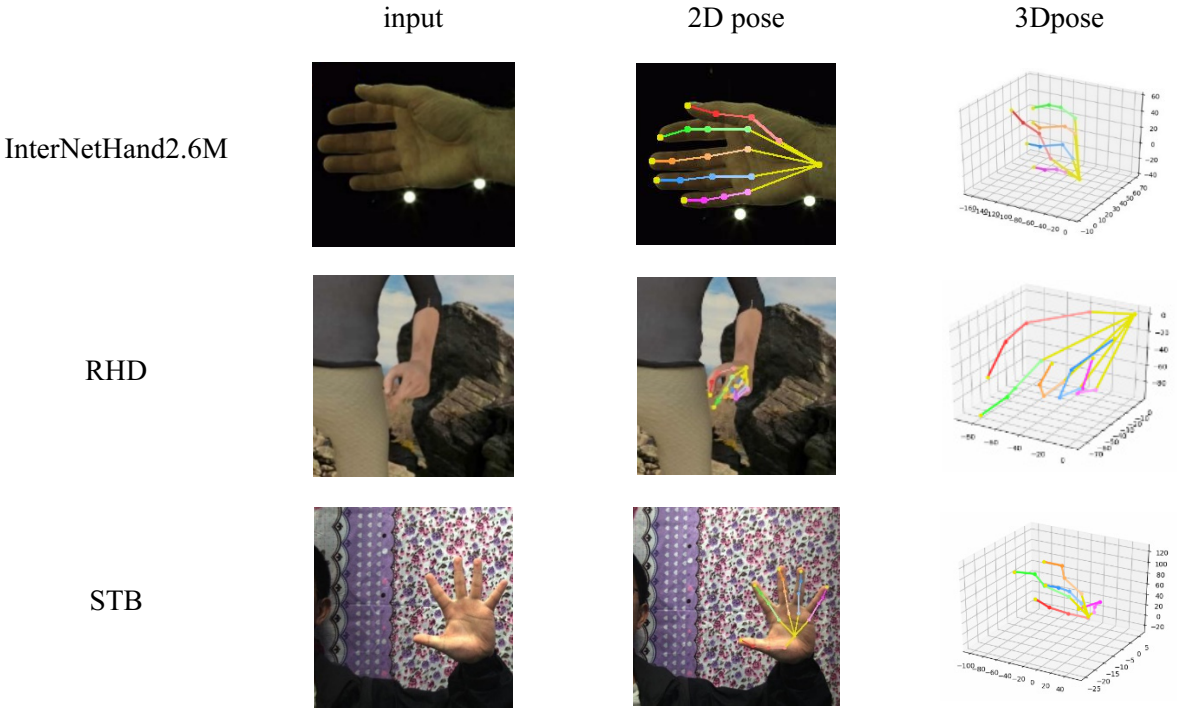


Fig. 5. 2D and 3D single hand Qualitative results

It also present interacting hand qualitative results in Fig. 6. The proposed approach can deal with severe occlusions caused by interacting hand, complex hand articulations. In the last one row of Fig. 6, 2D and 3D interacting hand pose estimation error surges. Considering our DB-InterNet achieves state-of-the-art performance on publicly available single hand datasets, it is concluded that 3D interacting hand pose estimation from a single RGB image is far from solved.

5 Conclusion

In this paper, it presents a highly accurate method for 3D hand pose estimation from a single color image. By introducing a β-Soft-Argmax function and DY-ReLU function in InterNet, the approach achieve the accurate joint localization and dynamic piecewise activation function. Confidence coordinate function boosts the position accuracy of the root joint and other keypoints in a targeted manner to tackle poor keypoint prediction caused by difficulties in distinguishing due to similarities. DY-ReLU function dynamically adjusts the piecewise function according to different inputs to determine the appropriate activation function to enhance the model capability and the learning of hidden keypoints in the network.

The experiments demonstrate that the proposed method can accurately estimate 2D and 3D joint positions with better estimation accuracy compared with the baseline algorithm, MPJPE and MRRPE of hand pose estimation are 13.49% and 30.53%, but the number of frames processed per second is slightly slower than the baseline algorithm. Experimental results on three public hand pose datasets indicate that this method achieves superior performance for 3D hand pose estimation. In future research, the lightweight network will be investigated to maintain the accuracy while further improving the real-time estimation speed of this algorithm to accommodate dynamic gesture estimation.

Acknowledgements

This paper is partially supported by the National Natural Science Foundation of China under Grant No.61771266, by the Natural Science Foundation of Inner Mongolia Autonomous Region under Grant No. 2019BS06005, by the the Research Program of Science and Technology at Universities of Inner Mongolia Autonomous Region under Grant No. NJZY20095, and by the Science and Technology Program of Inner Mongolia Autonomous Region Grant No. 2019GG138.

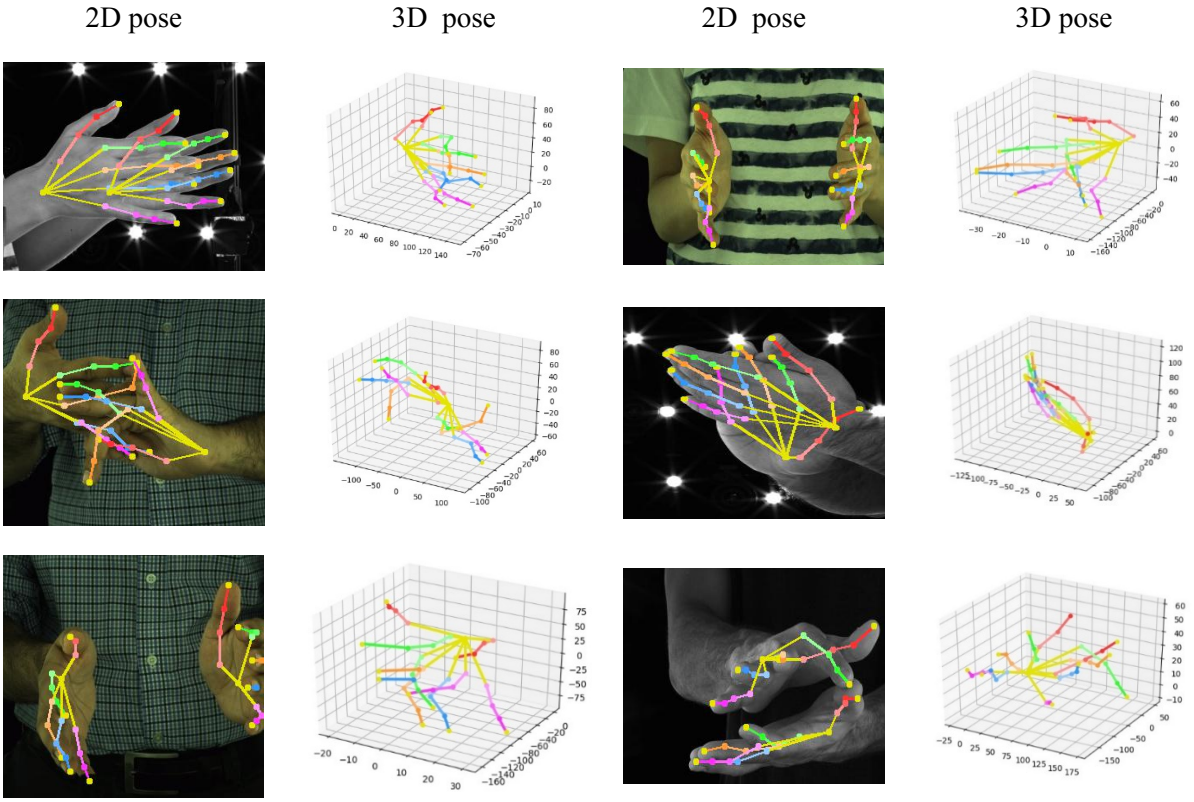


Fig. 6. 2D and 3D interacting hand qualitative results

References

[1] Y. Ma, Z. H. Mao, W. Jia, C. Li, J. Yang, M. Sun, Magnetic hand tracking for human-computer interface, IEEE Transactions on Magnetics 47(5)(2011) 970-973.

[2] J. Gałka, M. Mąsior, M. Zaborski, K. Barczewska, Inertial motion sensing glove for sign language gesture acquisition and recognition, IEEE Sensors Journal 16(16)(2016) 6310-6316.

[3] C. Keskin, F. Kıraç, Y.E. Kara, L. Akarun, Hand pose estimation and hand shape classification using multi-layered randomized decision forests, in: Proc. European Conference on Computer Vision (ECCV), 2012.

[4] L. Ge, H. Liang, J. Yuan, D. Thalmann, 3D convolutional neural networks for efficient and robust hand pose estimation from single depth images, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[5] L. Ge, H. Liang, J. Yuan, D. Thalmann, Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[6] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, Z. Liu, Dynamic relu, in: Proc. European Conference on Computer Vision (ECCV), 2020.

[7] Z. Zhang, J. Tang, G. Wu, Simple and lightweight human pose estimation. <https://arxiv.org/abs/1911.10346>, 2019.

[8] D. Tzionas, A. Srikantha, P. Aponte, J. Gall, Capturing hand motion with an RGB-D sensor, fusing a generative model with salient points, in: Proc. German Conference on Pattern Recognition, 2014.

[9] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, J. Gall, Capturing hands in action using discrim inative salient points and physics simulation, International Journal of Computer Vision 118(2)(2016) 172-193.

[10] A. Makris, N. Kyriazis, A. A. Argyros, Hierarchical particle filtering for 3d hand tracking, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2015.

[11] C. Wan, T. Probst, L. V. Gool, A. Yao, Self-supervised 3d hand pose estimation through training by fitting, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[12] L. Fang, X. Liu, L. Liu, H. Xu, W. Kang, JGR-P2O: Joint Graph Reasoning based Pixel-to-Offset Prediction Network for 3D Hand Pose Estimation from a Single Depth Image, in: Proc. European Conference on Computer Vision (ECCV), 2020.

[13] L. Ge, Y. Cai, J. Weng, J. Yuan, Hand pointnet: 3d hand pose estimation using point sets, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[14] C.R. Qi, H. Su, K. Mo, L.J. Guibas, Pointnet: Deep learning on point sets for 3d classification and segmentation, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[15] C.R. Qi, L. Yi, H. Su, L.J. Guibas, Pointnet++: Deep hierarchical feature learning on point sets in a metric space. <https://arxiv.org/abs/1706.02413>, 2017.

[16] A. Sinha, C. Choi, K. Ramani, Deephand: Robust hand pose estimation by completing a matrix imputed with deep features, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[17] G. Moon, J.Y. Chang, K.M. Lee, V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[18] C. Zimmermann, T. Brox, Learnning to estimate 3d hand pose from single rgb images, in: Proc. IEEE International Conference on Computer Vision (ICCV), 2017.

[19] T. Simon, H. Joo, I. Matthews, Y. Sheikh, Hand keypoint detection in single images using multiview bootstrapping, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[20] Y. Cai, L. Ge, J. Cai, J. Yuan, Weakly-supervised 3d hand pose estimation from monocular rgb images, in: Proc. European Conference on Computer Vision (ECCV), 2018.

[21] P. Panteleris, I. Oikonomidis, A. Argyros, Using a single rgb frame for real time 3d hand pose estimation in the wild, in: Proc. 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018.

[22] L. Yang, A. Yao, Disentangling latent hands for image synthesis and pose estimation, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[23] U. Iqbal, P. Molchanov, T.B.J. Gall, J. Kautz, Hand pose estimation via latent 2.5 d heatmap regression, in: Proc. European Conference on Computer Vision (ECCV), 2018.

[24] G. Moon, J.Y. Chang, K.M. Lee, Camera distance-aware top-down approach for 3d multi-person pose estimation from a single rgb image, in: Proc. IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[25] X. Sun, B. Xiao, F. Wei, S. Liang, Y. Wei, Integral human pose regression, in: Proc. European Conference on Computer Vision (ECCV), 2018.

[26] D.C. Luvizon, H. Tabia, D. Picard, Human pose regression by combining indirect part detection and contextual information, Computers & Graphics (2019) 15-22.

[27] G. Moon, S.I. Yu, H. Wen, T. Shiratori, K.M. Lee, InterHand2. 6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image. <https://arxiv.org/abs/2008.09309>, 2020.

[28] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, Q. Yang, 3D hand pose tracking and estimation using stereo matching. <https://arxiv.org/abs/1610.07214>, 2016.

[29] C. Ionescu, D. Papava, V. Olaru, C. Sminchisescu, Human3. 6M: Large scale datasets and predictive methods for 3d human sensing in natural environments, IEEE transactions on pattern analysis and machine intelligence 36(7)(2013) 1325-1339.

[30] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta1, S. Sridhar, D. Casas, C. Theobalt, Ganerated hands for real-time 3d hand tracking from monocular rgb, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition(CVPR), 2018.