3D Hand Pose Estimation from RGB image

Chih-Tien Kuo
University of Minnesota-Twin Cities
kuo00013@umn.edu

Eng Hock Lee
University of Minnesota-Twin Cities
leex7132@umn.edu

Abstract

The use of hand gesture for human-computer interaction is an attractive alternative method to the conventional interactive peripheral device such as computer mouse and keyboard. Furthermore, the advancement of the virtual environment has encouraged the development of a hand tracking system to deliver an accurate and real-time hand motion capture. Fortunately, recent development in computer vision has made it possible to capture 3D hand poses from 2D RGB images. In this project, we aimed to improve upon the existing method of 3D hand pose estimation by introducing a biologically inspired loss function to further enhance the machine learning model generalization. Besides, we intended to resolve the issue of image occlusion by utilizing a new public available hand dataset that contain images with hand occlusion. Our results showed that not only were we able to overcome the issue of hand occlusion, we also achieved lower test hand pose error than that of the baseline methods. We also showed that the outcome of our proposed method provides a more solid and natural hand pose output compared to the baseline methods.

1. Introduction

Out of all human body parts, the hand is the most important input device for natural human-computer interaction (HCI) [1]. This is because it serves as an effective and general purpose tool to allow wide-range of commands and control interfaces to the computer. The use of hand as HCI can also allow the deployment of much broader applications in sophisticated computing systems such as virtual reality [2] and augmented reality [3], making the human interaction much more intuitive and natural. More importantly, it serves as a high degree-of-freedom control device to many demanding applications such as musical performance and surgical simulations, both which require highly sensitive and precise control to function.

A pivotal part of using the hand as a tool for HCI is the ability to accurately capture the hand motion. Currently, the most successful hand motion capturing tools are magnetic [4] or inertial [5] tracking devices. These hardware devices, also called ”data glove”, are able to accurately capture the global position of the hand(s) in real time. A more advanced data glove can also induce so-called haptic feedback to simulate a sense of touch to further enhance immersion experience. However, these devices tend to be bulky, expensive, and would require complex calibration and setup procedure in order to obtain precise measurements.

An alternative method to capture hand motion is the computer vision based method. This method is promising because it is free of cumbersome hardware devices and can allow a more natural and non-contact interactive experience. However, accurately estimating the hand motion from images is a very challenging and computationally intensive problem, especially when dealing with the fundamental issue of image occlusion. Image occlusion usually occurs in two different ways: when external object partially obstructs the view of the hand pose in the image, and part of the hand is self-occluded due to the position and angle of the camera [6, 7, 8]. Such issues limit the information available in an image, making it difficult to accurately estimate the hand pose. Fortunately, recent advancements in computer vision and deep learning techniques have made it possible to capture hand motion and estimate hand pose accurately in unprecedented speed. One such example is the work done by Ge *et al.* [9], where they achieved estimating 3D hand pose and generating 3D hand mesh from a single RGB image via a series of neural network models. In this project, we proposed to improve Ge's 3D hand pose estimation by introducing a new set of loss function inspired from the biological bone structure of the human hand. We postulated that such loss function can further improve the model generalization and can result in better and more natural estimated hand pose. Besides, we aimed to overcome the image occlusion issue by utilizing FreiHAND dataset [10]: an RGB hand pose image dataset that contains images with object- and self-occlusion.

2. Related Works

Many works on 3D hand pose estimation revolve around using two different types of optical sensors, namely, the

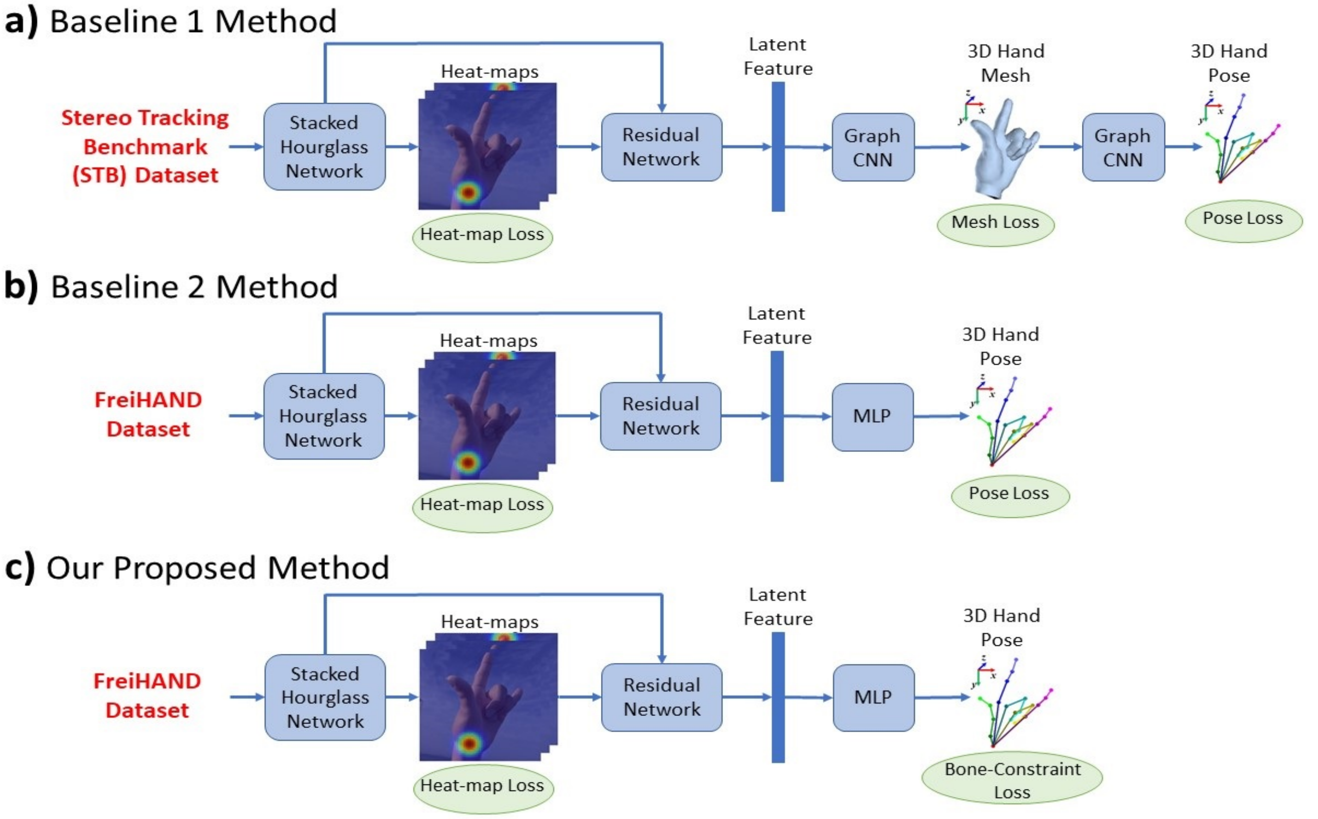

Figure 1: a) The model structure used by Ge *et al.* [9] as the baseline 1 method. b) The model structure of the baseline 2 method. Here, the FreiHAND dataset was used during the model training and testing. Note that the two graph CNNs were simplified to an MLP network for estimating the 3D hand pose. c) The model structure of the proposed method. The bone constraint loss function was used as the loss function to further generalize the model by imposing bone structural constraint onto the hand pose.

RGB and the RGB-Depth sensor. While these two types of sensors only differ with additional depth level information, such information can prove useful in providing an additional layer of geometrical description for a more robust pose estimation [11, 12, 13, 14, 15, 16]. In addition, many successful implementations of hand pose estimation utilized multidimensional inputs including the use of depth and pose [17, 18].

Alternatively, information on depth can also be extracted from a single RGB image to estimate 3D hand pose. Panteleris *et al.* [19] used predicted 2D hand joint location from the convolutional neural network (CNN) to estimate a 3D hand pose by solving the inverse kinematics problem. Similar to Panteleris, Zimmermann *et al.* [20] utilized CNN to predict 2D hand joint location, but used another CNN to predict the 3D hand joint location. He *et al*. [21] estimated 3D hand joint locations and then used Adverserial Learning model to fine-tune the joint locations. Recently, Ge *et al.* [9], Baek *et al.* [22], and Mueller *et al.* [23] designed a new neural network structure that allows for the prediction of both 2D and 3D hand joint locations concurrently. The idea was to simultaneously compensate the depth ambiguities in 2D joint location by using the 3D joint information as well as resolve 3D global hand rotation from the 2D joint information. Such implementation had shown to provide excellent 3D hand pose estimation even without the use of a depth sensor.

3. Baseline Method

We aimed to replicate Ge's 3D hand pose estimation model as the baseline 1 method [9]. In their work, the stereo tracking benchmark dataset (STB) [24] was used as the training and test data. The learning model structure was described next. First, a single input RGB image was passed through a two-stacked "hourglass" network [25] to extract 2D heat-maps of 21 different hand joint positions. Then, these heat-maps were combined with the image feature maps and fitted into a residual network [26] to output an array of latent feature vector. From here, the latent feature vector was passed through a Graph CNN [27] to generate 3D hand mesh, which was then regressed to obtain the 3D hand joint locations by using another Graph CNN. The details of the model structure can be found in the original paper [9]. The described procedure was pictorially illustrated in Figure 1a. We used the pre-trained model provided by [9] to evaluate their result.

However, using the baseline 1 method as the comparison to our proposed method was unfair for one reason. Instead of using STB dataset, we planned to use FreiHAND dataset for our proposed method to overcome the image occlusion issue. Therefore, comparing the results between these two models could lead to bias and false conclusion due to the different datasets being used and evaluated. For this reason, we devised another baseline 2 method that follows the similar model structure as baseline 1 method, but uses FreiHAND as the dataset. The model structure of the baseline 2 method is shown in Figure 1b. Note that the two graph CNNs used to estimate the 3D hand mesh and 3D hand pose in the baseline 1 method was simplified to a multilayer perceptron (MLP) network in the baseline 2. The reason was because our goal in this project was not to generate the 3D hand mesh, but rather to estimate the 3D hand pose. Therefore, it was redundant to have two graph CNNs in this case. Furthermore, in the original paper by Ge [9], they substituted the two graph CNNs with an MLP network and was able to achieve a similar estimation error. For this reason, we decided to implement the MLP to estimate the 3D hand pose.

For the baseline 2, the networks were supervised by heat-map loss L_{hm} for 2D hand pose estimation and pose loss L_{pose} for 3D hand pose estimation. The heat-map loss was formulated as:

$$L_{hm} = \sum_{j=1}^{21} \left\| \boldsymbol{h}_j - \hat{\boldsymbol{h}}_j \right\|_2^2 \tag{1}$$

where $\boldsymbol{h}_j$ and $\hat{\boldsymbol{h}}_j$ were ground truth and estimated heat-maps respectively, and the mean squared error was summed over 21 hand joints. The heat-map resolution was set as 64 × 64 px, and the ground truth heat-map was defined as a 2D Gaussian with a standard deviation of 2 px centered on the ground truth 2D joint location. The 2D joint locations were projected from the ground truth 3D joint location using the camera intrinsic matrices provided in the FreiHAND dataset.

Besides, the loss function for 3D hand pose estimation used in the baseline 2 was formulated as:

$$L_{pose} = \sum_{j=1}^{21} \left\| \boldsymbol{\phi}_j - \hat{\boldsymbol{\phi}}_j \right\|_2^2 \tag{2}$$

where $\boldsymbol{\phi}$ and $\hat{\boldsymbol{\phi}}$ were the ground-truth and estimated 3D joint locations in the 3-dimensional space respectively, and the mean squared error was summed over 21 hand joints. However, such loss function assumed every joints in the hand comprised of full translational degree-of-freedom, without considering the kinematics of human hand [28, 29, 30, 31]. The total loss function for the baseline 2 method L_{base} can be formulated as:

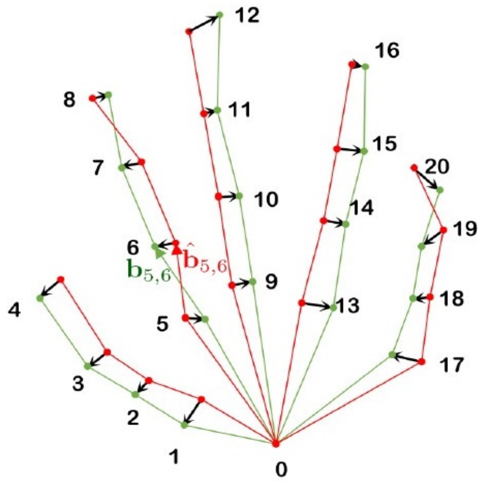

Figure 2: Visualization of the differences between the ground truth (shown as green) and the estimated hand pose (shown as red). It can be seen that the pose loss L_{pose} is small but the bone length L_{len} and bone direction L_{dir} loss can be large.

$$L_{base} = L_{hm} + L_{pose} \tag{3}$$

4. Proposed Method

For our proposed method, we used the same model structure as the baseline 2 method except for introducing a new bone-constraint loss function [21]. This bone-constraint loss function was biologically inspired and it imposed structural constraint based on the kinematics of human hand bone [21]. Specifically, the proposed bone-constrained loss function L_{prop} can be formulated as:

$$L_{prop} = \lambda_{hm} L_{hm} + \lambda_{pose} L_{pose} + \lambda_{len} L_{len} + \lambda_{dir} L_{dir} \tag{4}$$

where λ_{hm}, λ_{pose}, λ_{len}, and λ_{dir} are the hyperparameters for the trade-off between these losses. L_{len} quantifies the distance in bone length between the ground truth and its estimates, which was defined as:

$$L_{len} = \sum_{i,j} \left| ||\boldsymbol{b}_{i,j}||_2 - ||\hat{\boldsymbol{b}}_{i,j}||_2 \right| \tag{5}$$

where $\boldsymbol{b}_{i,j} = \boldsymbol{\phi}_i - \boldsymbol{\phi}_j$ was the ground truth bone vector between joint i and j, and $\hat{\boldsymbol{b}}_{i,j}$ was its predicted counterpart.

This loss function imposed the translational constraint on the joints to provide a more rigid and natural hand skeleton structure. Besides, L_{dir} measured the deviation in the direction of bones:

$$L_{dir} = \sum_{i,j} \left\| \frac{\boldsymbol{b}_{i,j}}{||\boldsymbol{b}_{i,j}||_2} - \frac{\hat{\boldsymbol{b}}_{i,j}}{||\hat{\boldsymbol{b}}_{i,j}||_2} \right\| \tag{6}$$

This loss function imposed the rotational constraint on the joints such that the estimated hand pose do not look distorted. Figure 2 illustrated the residual between the ground truth hand pose (shown as green) and the estimated hand pose (shown as red). In the figure, the pose loss L_{pose} can be small since the distance between the estimated and the ground truth of each hand joint was small. However, the bone length L_{len} and the bone direction L_{dir} loss can be large due to the fact that the estimated and true bone vector connecting two hand joints differ to a certain degree. Figure 1c summarized our proposed modeling procedure.

5. Experiments

5.1. Dataset Description

We trained the baseline 2 and the proposed model using the FreiHAND dataset [10]. It contained 130,240 and 3,960 training and test samples respectively. Within the training samples, it consisted of 32,560 different hand poses combined with 4 different sets of synthetic background. Each training sample was provided with an RGB image of 224 × 224 px, hand segmentation mask, intrinsic camera matrix, hand scale, 3D shape annotation, and 3D keypoint annotation for 21 hand joints. All of these information were used for 3D pose reconstruction as part of the pre-processing step. On the other hand, each evaluation sample was provided an RGB image with the dimension of 224 × 224 px, intrinsic camera matrix, and hand scale.

5.2. Implementation Details

We implemented the experiment using the PyTorch framework. The baseline 2 and proposed models were trained using 20,000 images from the training set. During the model training, we divided the model training into three stages, as shown in Table 1. First, we trained the stacked hourglass network using the heat-map loss. Then, the residual network and MLP was trained using the 3D reconstruction loss (pose loss and bone-constraint loss for the baseline 2 and proposed method, respectively). Finally, we trained the stacked hourglass network, residual network, and MLP all together using the heat-map loss and 3D reconstruction loss.

All networks were trained for 100 epochs using RMSprop optimizer [32] and a mini-batches of size 8. In stage I, the stacked hourglass network is trained with the learning rate of 10^{-3}. In stage II, the residual network and MLP are trained with the learning rate of 10^{-3}. When training the full model in stage III, the initial learning rate is set as 10^{-4} and decreased to 10^{-5} after 50 epochs. As for the hyperparameters of the bone-constraint loss function, we set $\lambda_{hm} = 0.1$, $\lambda_{pose} = 1$, $\lambda_{len} = 0.001$, and $\lambda_{dir} = 0.1$ for all three stages of the proposed methods.

Stage	Stacked Hourglas Network	ResNet & MLP	Learning Rate
I	✓		10^{-3}
II		✓	10^{-3}
III	✓	✓	$10^{-4}/10^{-5}$

Table 1: Implementation details in model training.

6. Quantitative Result

Since the FreiHAND dataset [10] did not provide the 3D annotations in the evaluation samples, we decided to use 2,000 images in the training samples for the evaluation (we only used 20,000 training samples to train our models).

We used the following two metrics to evaluate the performance of 3D hand pose estimation: (i) 3D pose error, which was the average joint location error in Euclidean distance between the estimated 3D joints and the ground truth; (ii) 3D PCK, which was the percentage of correct keypoints whose error in Euclidean distance is under a threshold. We performed the evaluation on baseline 1, baseline 2, and the proposed model and the results were shown in Table 2 and Figure 3.

For quantitative evaluation, we focused on the comparison between the baseline 2 and the proposed model. This was because, as previously mentioned in Section 3, the baseline 1 was trained using the STB dataset [24] and therefore it is not fair to compare baseline 1 with our proposed model, which was trained using the FreiHAND dataset [10].

Table 2 shows the 3D pose error for all three models. The baseline 2 method, which was trained using only the 3D pose loss, achieved the performance of 22.84 mm in 3D pose error. However, the proposed model, which was trained using bone-constraint loss, outperformed the baseline 1 and 2 methods by achieving the 3D pose error of 15.36 mm. Besides, the 3D PCK in Figure 3 also agreed the proposed model outperformed baseline 1 and 2 methods for all error thresholds and reached over 0.9 at the error threshold of 30 mm.

7. Qualitative Result

For qualitative comparison, we evaluated our proposed model in two folds: (i) Effectiveness of training using the

3D Pose Error		
Baseline 1	Baseline 2	Proposed model
94.10 mm	22.84 mm	15.36 mm

Table 2: Quantitative evaluation in 3D pose error for baseline 1, baseline2, and proposed model using 2000 test images from FreiHAND dataset.

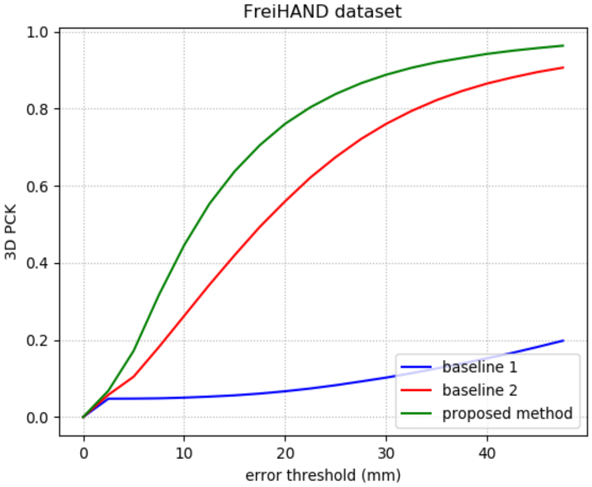

Figure 3: 3D PCK on the FreiHAND dataset for the comparison between the proposed method and the baselines.

FreiHAND dataset [10]; (ii) Effectiveness of training using the bone-constraint loss.

In the first comparison, our proposed model was compared with the baseline 1. We tested on a real-world image taken by ourselves, which included an occluded hand as shown in Figure 4. In the figure, the baseline 1 failed to reconstruct the occluded hand due to the fact that the baseline 1 was trained using the STB dataset [24], which does not contain images with occluded hands. On the other hand, our proposed method was able to successfully reconstruct the 3D hand pose with occlusion after training using the FreiHAND dataset [10].

In the second comparison, our proposed model was compared with the baseline 2. Figure 5 shows some of the zoom-in view of the hand structure. As shown in the figure, the reconstructed hand structure by our proposed model was more similar to the ground truth structure compared to the baseline 2.

Qualitative results of 3D hand pose reconstruction for the FreiHAND dataset [10] of the proposed model are shown in Figure 6. In the figure, multiple view points for 3D pose were provided to present the reconstruction quality.

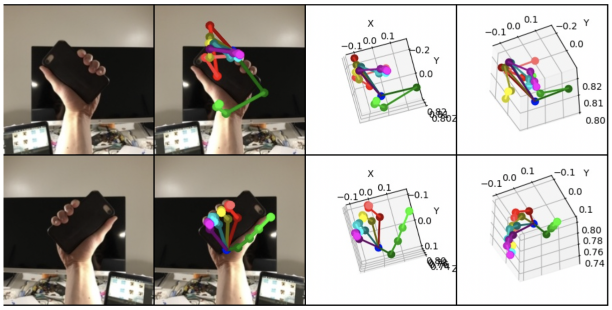

Figure 4: Qualitative results of baseline 1 (upper row) and the proposed model (lower row) with occluded hands. **From left to right:** input image, output 2D pose, output 3D pose from the camera viewpoint, output 3D pose from another viewpoint.

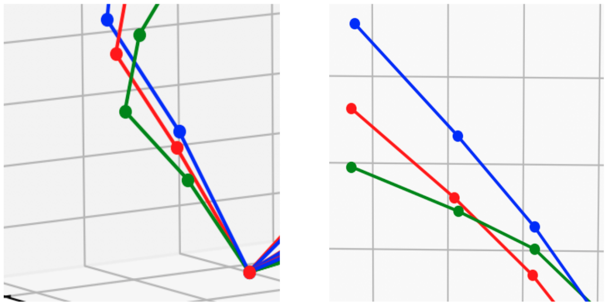

Figure 5: Comparison between the proposed model and baseline 2 with the zoom-in view of hand bones. **Red:** ground truth. **Green:** baseline 2. **Blue:** proposed model.

8. Conclusion

In summary, we performed accurate 3D hand pose estimation from a single RGB image. We proposed a bone-constraint loss function to further impose constraints on the hand bone structure for a more natural and solid hand pose estimation. Indeed, our result has shown that such loss function did improve the overall hand pose estimation (15.36mm for proposed method versus 22.84mm for baseline 2 method). Qualitative result has also shown that the estimated hand pose by proposed method had a closer hand bone structure to the ground truth than that of baseline 2 method (see Figure 5). On top of that, we resolved the issue of image occlusion by introducing the FreiHAND dataset as the training set for our proposed method, and the result can be visualized in Figure 4.

Throughout the project period, we have stumbled upon several points that are note-worthy. First, we only utilized around 15% of the total FreiHAND image sample for model training due to the limited memory and computing capacity of our computer system. Therefore, the model, in principle, could achieve much better model generalization if all

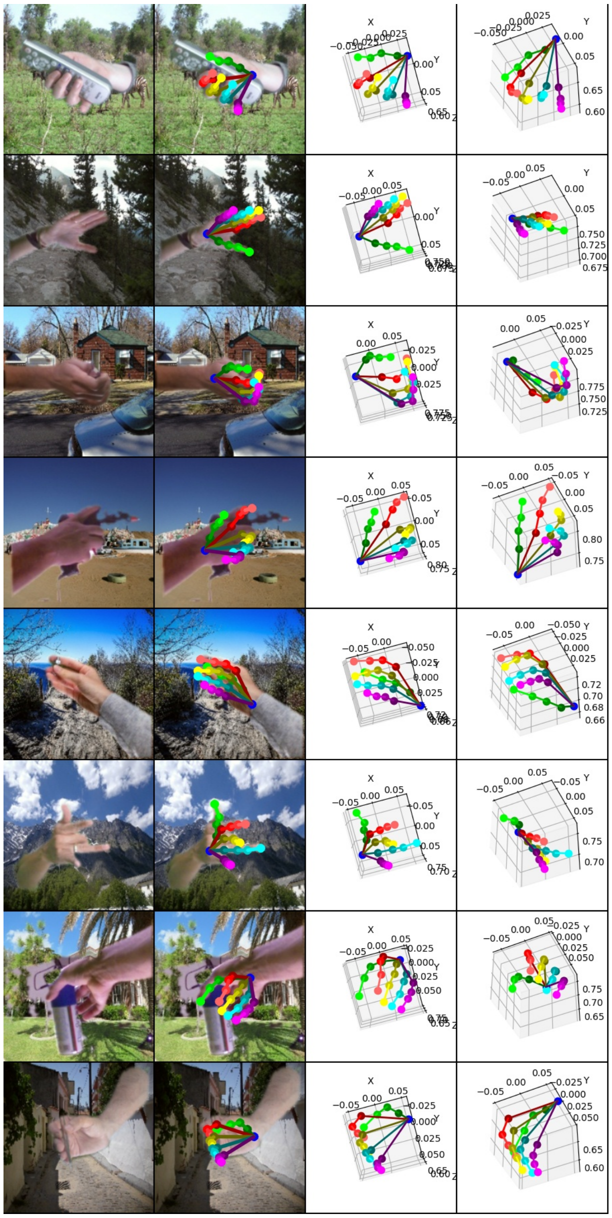

Figure 6: Qualitative results of our proposed model for the FreiHAND dataset [10]. **From left to right:** input image, output 2D pose, output 3D pose from the camera viewpoint, output 3D pose from another viewpoint

of the FreiHAND dataset was utilized. Second, we realized the bone direction loss L_{dir} decreases at a much faster rate than the bone length loss L_{len} during the model training, suggesting L_{dir} played a more critical role than L_{len}. This is further evident by the resulting bone structure shown in Figure 5, where the bone direction deviated much lower than the bone length when comparing the estimated hand pose of the proposed method and the ground truth.

Both of the students contributed equally when working on this project. Specifically, both Chih-Tien and Eng Hock designed the experiments. Chih Tien imported the FreiHAND dataset and implemented the baseline method. Eng Hock implemented the bone-constraint loss function and ran the experiments. Both Chih-Tien and Eng Hock wrote the proposal report, the final report, and the presentation slide.

References

[1] Erol, Ali, et al. “Vision-based hand pose estimation: A review.” *Computer Vision and Image Understanding* 108.1-2 (2007): 52-73.

[2] Cameron, Charles R., et al. “Hand tracking and visualization in a virtual reality simulation.” *2011 IEEE systems and information engineering design symposium*. IEEE, 2011.

[3] Malik, Shahzad, Chris McDonald, and Gerhard Roth. “Hand tracking for interactive pattern-based augmented reality.” *Proceedings. International Symposium on Mixed and Augmented Reality*. IEEE, 2002.

[4] Ma, Yinghong, et al. “Magnetic hand tracking for human-computer interface.” *IEEE Transactions on Magnetics* 47.5 (2011): 970-973.

[5] Gałka, Jakub, et al. “Inertial motion sensing glove for sign language gesture acquisition and recognition.” *IEEE Sensors Journal* 16.16 (2016): 6310-6316.

[6] Benenson R. (2014) Occlusion Detection. In: Ikeuchi K. (eds) Computer Vision. Springer, Boston, MA

[7] Rehg, James M., and Takeo Kanade. “Model-based tracking of self-occluding articulated objects.” *Proceedings of IEEE International Conference on Computer Vision*. IEEE, 1995.

[8] Utsumi, Akira, and Jun Ohya. “Multiple-hand-gesture tracking using multiple cameras.” Proceedings. *1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Cat. No PR00149). Vol. 1. IEEE, 1999.

[9] Ge, Liuhao, et al. “3d hand shape and pose estimation from a single rgb image.” *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2019.

[10] Zimmermann, Christian, et al. “FreiHAND: A Dataset for Markerless Capture of Hand Pose and Shape from Single RGB Images.” *Proceedings of the IEEE International Conference on Computer Vision*. 2019.

[11] Yang, Xiaodong, and YingLi Tian. "Super normal vector for human activity recognition with depth cameras." *IEEE transactions on pattern analysis and machine intelligence* 39.5 (2016): 1028-1039.

[12] Oreifej, Omar, and Zicheng Liu. "Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2013.

[13] Shotton, Jamie, et al. "Real-time human pose recognition in parts from single depth images." *CVPR 2011*. Ieee, 2011.

[14] Rogez, Grégory, et al. "3D hand pose detection in egocentric RGB-D images." *European Conference on Computer Vision*. Springer, Cham, 2014.

[15] Mueller, Franziska, et al. "Real-time hand tracking under occlusion from an egocentric rgb-d sensor." *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2017.

[16] Garcia-Hernando, Guillermo, et al. "First-person hand action benchmark with rgb-d videos and 3d hand pose annotations." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[17] Dibra, Endri, et al. "Monocular RGB hand pose inference from unsupervised refinable nets." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2018.

[18] Baek, Seungryul, et al. "Kinematic-layout-aware random forests for depth-based action recognition." *arXiv preprint arXiv:1607.06972* (2016).

[19] Panteleris, Paschalis, Iason Oikonomidis, and Antonis 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)*. IEEE, 2018.

[20] Zimmermann, Christian, and Thomas Brox. "Learning to estimate 3d hand pose from single rgb images." *Proceedings of the IEEE International Conference on Computer Vision.* 2017.

[21] He, Yiming, et al. "3D Hand Pose Estimation in the Wild via Graph Refinement under Adversarial Learning." *arXiv preprint arXiv:1912.01875* (2020).

[22] Baek, Seungryul, Kwang In Kim, and Tae-Kyun Kim. "Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019.

[23] Mueller, Franziska, et al. "Ganerated hands for real-time 3d hand tracking from monocular rgb." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

[24] Zhang, Jiawei, et al. "A hand pose tracking benchmark from stereo matching." *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017.

[25] Newell, Alejandro, Kaiyu Yang, and Jia Deng. "Stacked hourglass networks for human pose estimation." *European conference on computer vision*. Springer, Cham, 2016.

[26] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

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

[28] Chen Chen, Fai, et al. "Constraint study for a hand exoskeleton: human hand kinematics and dynamics." *Journal of Robotics* 2013 (2013).

[29] Ma'touq, Jumana, Tingli Hu, and Sami Haddadin. "Sub-millimetre accurate human hand kinematics: from surface to skeleton." *Computer methods in biomechanics and biomedical engineering* 21.2 (2018): 113-128.

[30] Cobos, Salvador, et al. "Efficient human hand kinematics for manipulation tasks." 2008 *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008.

[31] Gustus, Agneta, et al. "Human hand modelling: kinematics, dynamics, applications." *Biological cybernetics* 106.11-12 (2012): 741-755.

[32] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop, coursera: Neural networks for machine learning. *University of Toronto, Technical Report*, 2012.