

3D Hand Pose and Shape Estimation from Monocular RGB via Efficient 2D Cues —Supplementary Material—

Fenghao Zhang¹, Lin Zhao², Shengling Li¹, Wanjuan Su², Liman Liu¹(✉), and Wenbing Tao²

© The Author(s)

Abstract Estimating 3D hand shape from a single-view RGB image is important for many applications. However, the diversity of hand shapes or postures, depth ambiguity, and occlusion of interactive objects from a monocular image result in the wrong pose and noisy hand mesh. Making full use of 2D cues (e.g., 2D pose) can effectively improve the quality of 3D human hand shape estimation. In this paper, we use 2D joint heatmaps to obtain spatial details for robust pose estimation and introduce a depth-independent 2D mesh to avoid the depth ambiguity in mesh regression and achieve efficient hand-image alignment. Our method consists of three cascaded stages: 2D cues extraction, pose encoding, and 3D mesh generation. Specifically, we first encode the image to get semantic features in the step of 2D cues extraction which is also used to predict hand joints and segmentation. Then, during the pose encoding stage, we propose a Hand Joints Encoder to learn spatial information from the joint heatmaps. Finally, the 3D hand mesh can be obtained in the step of 3D mesh generation where a Mesh Squeeze-and-Excitation Block and a Global Mesh Refiner are introduced. The Mesh Squeeze-and-Excitation Block aims to fuse different hand features to enhance the perception of 3D hand structures, while the Global Mesh Refiner is proposed to learn the non-local relation between vertexes of hand mesh from predicted 2D mesh and predict an offset hand mesh to further fine-tune the reconstruction results and facilitate the shape prediction. Extensive quantitative and qualitative results on benchmark dataset FreiHAND demonstrate that our approach achieves state-of-the-art performance.

Keywords 3D human hand reconstruction; Monocular RGB image; Deep learning; Image features; 3D Mesh

In the following sections we provide additional details of our proposed method.

1 Pose Feature Encoding

The proposed HJE extracts the pose features f_{pose} from the predicted 2D joint heatmaps. This process can be calculated as follows:

$$p'_{ij} = \begin{cases} p_{ij} & \text{if } p_{ij} = \text{MAX}(H_j) \\ 0 & \text{otherwise} \end{cases},$$
$$H_{hand} = \sum_{j=1}^{21} H'_j, \quad (1)$$
$$f_{pose} = \text{RoIAlign}(H_{hand}, B),$$

where p_{ij} and p'_{ij} are elements in heatmap H_j and H'_j . B indicates the region of interest, determined by joint position. See Section 3.1 in the manuscript for more details.

2 Global Dependencies

In Section 3.3 in the manuscript, we introduced a Global Mesh Refiner. Here we show the specific details of GMR. As shown in Fig. 1, we use three linear layers to introduce input features V'_f into *Query*, *Key*, and *Value* branches, respectively. The non-local attention can be calculated as follows:

- 1 School of Biomedical Engineering, South-Central Minzu University, Wuhan 430074, China. E-mail: F. Zhang, 2020110502@mail.scuec.edu.cn; S. Li, 2021110529@mail.scuec.edu.cn; L. Liu, liman-liu@mail.scuec.edu.cn(✉)
- 2 National Key Laboratory of Science and Technology on Multi-spectral Information Processing, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China. E-mail: L. Zhao, linzhao@hust.edu.cn; W. Su, suwanjuan@hust.edu.cn; W. Tao, wenbingtao@hust.edu.cn

Manuscript received: 2022-01-01; accepted: 2022-01-01

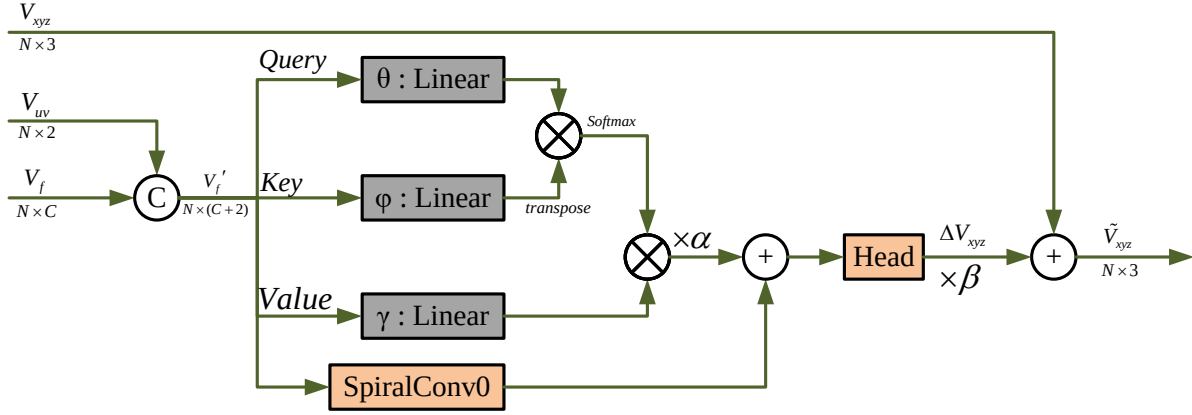


Fig. 1 The global mesh refiner. We fine-tune the input rough 3D hand mesh with predicted offset mesh in GMR.

$$\begin{aligned} Q &= f_{\theta}(V'_f), K = f_{\phi}(V'_f), V = f_{\gamma}(V'_f), \\ Y &= \text{Softmax}\left(\frac{Q \otimes K^T}{\sqrt{d_k}}\right) \otimes V, \end{aligned} \quad (2)$$

where $[\cdot]$ is concatenate operator, f_{θ}, f_{ϕ} and f_{γ} are linear layer. d_k indicates the dimension, $\otimes$ is matrix multiplication and Y is a mesh feature map containing global information.

3 Spiral Operator

In this section, we compare different spiral operators. As shown in Fig.2, the SpiralConv [1] defines a spiral selection of neighbors for a center vertex v :

$$\begin{aligned} 0\text{-ring}(v) &= \{v\}, \\ (k+1)\text{-ring}(v) &= \mathbb{N}(k\text{-ring}(v)) \setminus k\text{-disk}(v), \\ k\text{-disk}(v) &= \cup_{i=0 \dots k} i\text{-ring}(v), \end{aligned} \quad (3)$$

where v is the center vertex, $\mathbb{N}(V)$ is the set of all neighborhood vertices in the set V .

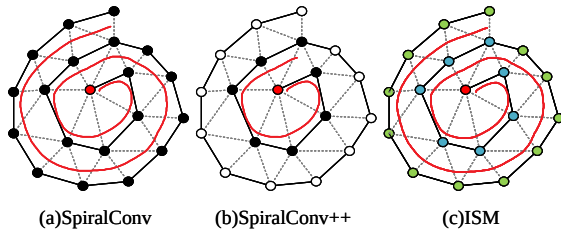


Fig. 2 Spiral selection of neighbors for a center vertex in clockwise order. (a): SpiralConv [1].(b): SpiralConv++ [2].(c): ISM [3].

SpiralConv++[2] uses a fixed-length sequence $S(v, l)$ as vertex neighbors (black dots), and uses the fully connected layer to collect neighbors information. ISM [3] uses hierarchy neighbors (red, blue and green dots):

$$S(v, l) = (v, 1\text{-ring}(v), \dots, k\text{-ring}(v)),$$

$$\begin{aligned} \text{SpiralConv++}(v) &= W(S(v, l)) + b, \\ o_i(v) &= W_i f(i\text{-disk}(v)) + b_i, \end{aligned} \quad (4)$$

$$\text{ISM}(v) = o_0(v) + [o_1(v), o_2(v), o_3(v)],$$

where the spiral length l is fixed for each vertex in the same layer of a neural network, W and b are shared weights and bias. ISM uses 4 different lengths of spiral sequence. And $[\cdot]$ is concatenate operator.

References

- [1] Lim I, Dielen A, Campen M, Kobbelt L. A simple approach to intrinsic correspondence learning on unstructured 3d meshes. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, 0–0.
- [2] Gong S, Chen L, Bronstein M, Zafeiriou S. Spiralnet++: A fast and highly efficient mesh convolution operator. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, 0–0.
- [3] Chen X, Liu Y, Ma C, Chang J, Wang H, Chen T, Guo X, Wan P, Zheng W. Camera-space hand mesh recovery via semantic aggregation and adaptive 2d-1d registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, 13274–13283.