HandGCNFormer: A Novel Topology-Aware Transformer Network for 3D Hand Pose Estimation

Yintong Wang[1,2], LiLi Chen[1,2,*], Jiamao Li[1,2,3], and Xiaolin Zhang[1,2,3,4,5]

[1]Bionic Vision System Laboratory, State Key Laboratory of Transducer Technology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
[3]Xiongan Institute of Innovation, Xiongan, 071700, China
[4]University of Science and Technology of China, Hefei, Anhui, 230027, China
[5]ShanghaiTech University, Shanghai 201210, China
{wytong, lilichen, jmli, xlzhang}@mail.sim.ac.cn

Abstract

Despite the substantial progress in 3D hand pose estimation, inferring plausible and accurate poses in the presence of severe self-occlusion and high self-similarity remains an inherent challenge. To mitigate the ambiguity arising from invisible and similar joints, we propose a novel Topology-aware Transformer network named HandGCNFormer, incorporating the prior knowledge of hand kinematic topology into the network while modeling long-range context information. Specifically, we present a novel Graphformer decoder with an additional node-offset graph convolutional layer (NoffGConv) that optimizes the synergy of Transformer and GCN, capturing long-range dependencies as well as local topology connection between joints. Furthermore, we replace the standard MLP prediction head with a novel Topology-aware head to better utilize local topology constraints for more plausible and accurate poses. Our method achieves state-of-the-art performance on four challenging datasets including Hands2017, NYU, ICVL, and MSRA.

1. Introduction

Accurate and robust 3D hand pose estimation is a crucial component within a variety of human-machine applications, including augmented reality, virtual reality, and third-person imitation learning. Hand pose estimation aims to estimate the location of hand joints from a single depth image or RGB image. As commodity depth cameras get more affordable and accurate, impressive advancements have been made in depth-based methods [4, 14, 9, 37, 35, 19, 17, 46]. However, it remains extremely challenging under the situations of severe self-occlusion and high self-similarity between hand parts, as shown in Figures 1a and 1b.

Humans have the capability to predict the accurate hand pose in complex scenarios thanks to their deep under-

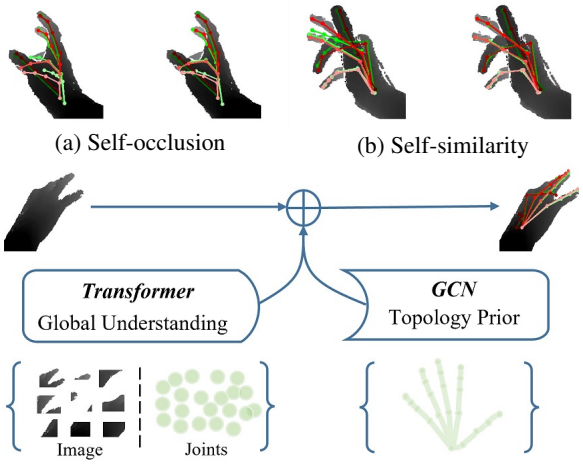


Figure 1: The illustration of the HandGCNFormer. (a) and (b) respectively indicate qualitative comparison between AWR (left) and our HandGCNFormer (right) under self-occlusion and self-similarity. Red pose is the ground truth. Green pose represents predicted result. (c) indicates the complementary feature representation of Transformer and GCN in HandGCNFormer.

*Corresponding author

standing of the scene and strong prior knowledge of hand kinematic structure, which provides sufficient context to mitigate the ambiguity generated by invisible and similar joints. Although CNN-based hand pose estimation approaches [4, 37, 19, 46] have been the dominant framework, they are incapable to model long-range dependencies due to operating on fixed-sized window. To break this limitation, recent methods [17, 18] leverage the superior global modeling capability of Transformer and yield better performance. Nevertheless, it only implicitly extracts the long-range dependencies underlying similarity of joint features, while ignoring the natural kinematic constraints of hand topology.

The kinematic topology of the hand reveals the inherent articulated connection between joints. Some previous works [1, 42, 50] have shown that the graph convolutional network (GCN) exhibits the powerful representation of topology. Recently, the pose-guided hierarchical graph convolution (PHG) method [35] attempts to model the long-range dependencies between hand parts through stacking multiple GCN layers. However, the cascaded GCN leads to error accumulation in long-term elements of the graph and the over-smoothing problem.

As illustrated in Figure 1c, we claim that the global attention of Transformer and local topology perception of GCN construct an effective and complementary feature representation. To maximize their synergy, we propose a novel Topology-aware Transformer network named **HandGCNFormer** unifying the non-autoregressive Transformer for modeling context information of depth image and long-range dependencies between joints, with the graph convolutional network (GCN) which naturally incorporates the hand topology prior into our network and explicitly learns the relative relationship between locally connected joints.

Specifically, we propose a **Graphformer decoder**. Each decoder block contains a novel node-offset graph convolutional layer (NoffGConv) in the front, followed by standard components including self-attention layer and cross-attention layer. Unlike vanilla GCN, NoffGConv decouples the node feature mapping and the offset feature mapping, enhancing the guidance of its own location information in the feature aggregation process.

In addition, most Transformer-based methods leverage a multiple layer perception (MLP) head consisting of fully connected layers to predict the coordinates of hand joints independently, ignoring the local connections among joints. We introduce a **Topology-aware head** based on semantic graph convolutional layer (SemGConv) [50] which incorporates the topology information without increasing model complexity. With a learned adjacency matrix, SemGConv is able to capture complex local spatial constraints between joints guided by hand topology, encouraging the Topology-aware head to obtain more plausible and accurate poses.

In summary, the contributions of this paper are four-fold:

- We propose a novel HandGCNFormer network for 3D hand pose estimation. Transformer and GCN layers are deeply integrated to model both global understanding of the scene and local topology connections of hand joints.
- A novel NoffGConv layer is proposed to decouple the node feature mapping and the offset feature mapping, which outperforms the popular GCNs for 3D hand pose estimation task.
- A Topology-aware head module is designed to adaptively establish the spatial topology constraints, which outperforms the standard MLP prediction head.
- Our method achieves state-of-the-art performance on four challenging datasets. In particular, it is superior to the top-performing approach by a margin of 3.2% with 7.6% fewer parameters for unseen subjects hand in Hands2017, revealing its excellent generalization ability.

2. Related work

2.1. 3D Hand Pose Estimation

3D hand pose estimation methods based on deep neural networks have exhibited high-quality prediction results, which can be divided into regression-based methods, detection-based methods, and hybrid methods according to the type of model output. Regression-based models [2, 4, 14, 13, 15, 26, 33, 32, 46] learn the mapping from the input image to output joint coordinates or angles directly. Oberweger *et al*. proposed DeepPrior [33] and DeepPrior++ [32] to learn the pose prior with a bottleneck layer and regress the pose with fully-connected layers. To better utilize fine-grained features, Pose-REN [4] applies multi-level cascade regression to iteratively refine the prediction, while other methods [13, 26, 14, 46] leverage the feature-level local ensemble. Despite the excellent performance, such methods suffer from a large model complexity.

Detection-based methods [30, 11, 37, 36, 31] generally predict a dense probability map for each joint from a depth image, point set, or voxel set. DenseReg [44] produces the 3D heatmap and unit vector field with an encoder-decoder module, maintaining the richer spatial context. However, since the post-processing of obtaining joint coordinates from heatmap is not differentiable, detection-based methods usually cannot be trained end-to-end. Later on, the hybrid methods [35, 27, 19, 39] are further proposed via combining the advantages of both methods. AWR [19] transforms the 3D hand joint coordinates as a 3D heatmap and unit vector field in a differentiable manner, implements direct supervision of joint position. However, the pure CNN-based methods fall short in understanding of global context due to their

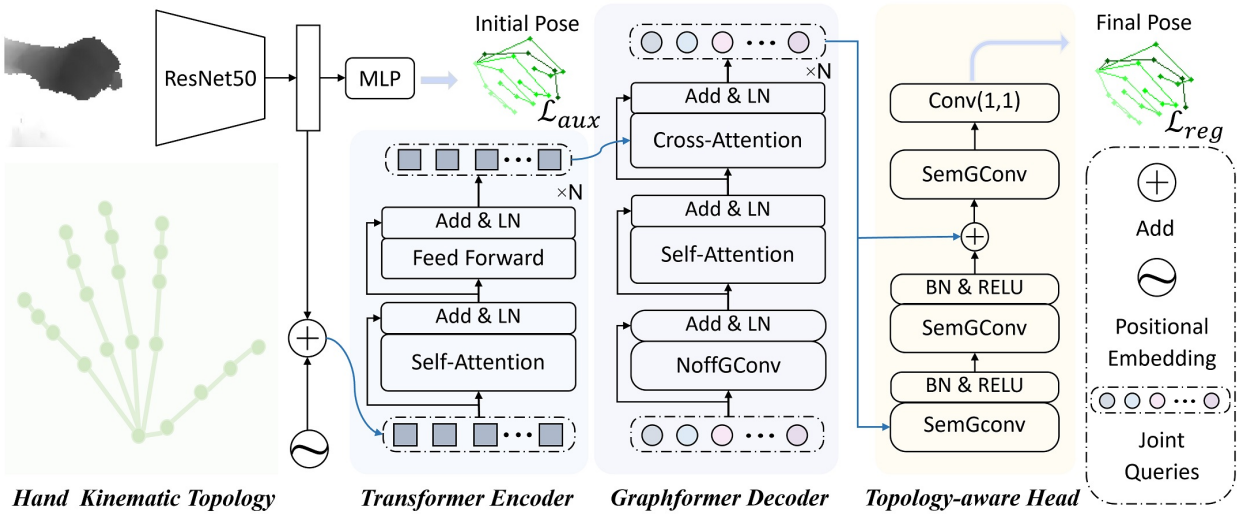


Figure 2: The overview of the HandGCNFormer. Our method introduces the prior knowledge of hand kinematic topology by NoffGConv and SemGConv layers, as well as models global understanding with self-attention mechanism, providing rich disambiguation evidence. ResNet and Transformer encoder form the image encoder module, capturing the global-local context of the image (Section 3.1). Graphformer decoder incorporates NoffGConv and attention modules capturing joint interaction globally without ignoring topological connections of joints (Section 3.2). Finally, the final pose is regressed by Topology-aware head, which constructs effective topology constraints during regressing (Section 3.3).

limited receptive field, making it difficult to handle severe self-occlusion and self-similarity cases which are common in 3D hand pose estimation.

2.2. Transformer in Computer Vision

Lately, the Transformer architecture [43] has been applied to image classification [7, 24], object detection [3, 51], and pose estimation [48, 28, 22, 23, 15]. In particular, PRTR [21] and TFPose [28] visualize the dynamic decoding process in Transformer decoder and demonstrate the applicability of Transformer to human pose modeling. In a closely related work, Hand-Transformer [17] applies a non-autoregressive Transformer decoding mechanism to localize each joint in parallel. Compared with the autoregressive approach, non-autoregressive decoding frees the restriction of sequence dependence and fulfills the real-time speed. However, detecting joints independently ignores the inherent adjacency relation among joints, leading to inferior performance, especially on invisible and similar joints.

2.3. Graph Convolutional Network

The GCN increasingly gains popularity for skeleton-based action recognition [29, 42, 47] and 2D-to-3D pose estimation tasks [34, 20], since it can effectively represent arbitrary topological data. SemGCN [50] is proposed to capture complex semantic relationships between neighbor joints of the human body. HOPE-Net [6] proposes an adaptive Graph U-Net inferring joint locations in 3D space from 2D keypoints. These works for 2D-to-3D lifting demonstrate that the topology information is essential to mitigate depth ambiguity. PHG [35] attempts to construct long-range dependencies of hand joints by exploiting a cascaded GCN module, achieving state-of-the-art performance. However, the cascaded GCN module exponentially introduces noisy information from extended neighbor nodes while constructing global relationships, leading to over-smoothing of the model. In this paper, we leverage Transformer to directly model global contextual information free from the limitations of the receptive field while incorporating GCN to capture the hand kinematic topology, which highly improves the representation of spatial structural features.

3. Methodology

The overview of our proposed HandGCNFormer is illustrated in Figure 2. It takes a depth image as input and predicts a set of 3D joint coordinates. The entire framework consists of an image encoder formed by a ResNet and a Transformer encoder, a Graphformer decoder, and a Topology-aware head.

3.1. Image Encoder

The image encoder extracts both local and global features from the input depth image. Our image encoder is inspired by DETR [3] which consists of a ResNet [16] and a Transformer encoder. Given a cropped hand depth image $\mathbf{I} \in \mathbb{R}^{H \times W}$, where H and W represent the image

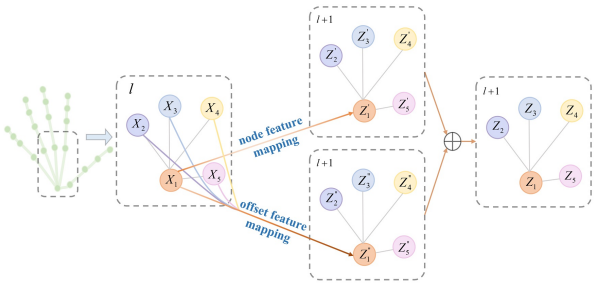


Figure 3: Illustration of NoffGConv. NoffGConv decouples the node feature mapping and the offset feature mapping during aggregating information. The gray lines indicate the connections between nodes, and the colored lines represent feature transfer.

height and width respectively, a ResNet is exploited to extract downsampled features $\mathbf{F} \in \mathbb{R}^{\frac{H}{32} \times \frac{W}{32} \times 2048}$. The feature map is then reduced in channels via a 1×1 convolutional layer and flattened spatially to obtain the sequence feature $\mathbf{T} \in \mathbb{R}^{\frac{HW}{1024} \times 256}$ that will be fed into the standard Transformer encoder. For retaining spatial positional information, sinusoidal positional embedding is added to the input sequence. Finally, the context features of input sequence are captured through a series of self-attentions and feed-forward networks (FFN).

3.2. Graphformer Decoder

The vanilla Transformer decoder consists of self-attention layers, cross-attention layers, and feed-forward networks, which are not aware of the inherent connections among joints that can be described by the hand kinematic topology (see lower left of Figure 2). To overcome this limitation, we design a Graphformer decoder that emphasizes the fusion of attention mechanism and GCN technique, benefiting from both the long-range dependencies and local topology connection of joints. Specifically, We build a graph $\mathcal{G} = \{\mathbf{V}, \mathbf{E}\}$ which consists of a set of nodes $\mathbf{V}$ and edges $\mathbf{E}$. Each node in the graph represents a hand joint. We incorporate the prior knowledge of hand kinematic topology into the model through the adjacency matrix of $\mathcal{G}$. There exists an edge between node i and j if and only if the two corresponding joints are connected in the hand kinematic topology.

In 3D hand pose estimation task, node features contain rich location information. On the other hand, the neighboring nodes also provide useful features to estimate the relative offsets which can play a critical role for invisible and similar joints. Inspired by this observation, we propose a node-offset graph convolutional layer (NoffGConv). As shown in Figure 3, NoffGConv decouples node feature mapping and offset feature mapping. The former depends on node feature alone, while the latter converges the refinement information flowing to central node from neighboring nodes and itself. To better complement with the following self-attention layer and to accelerate model convergence speed, NoffGConv applies a fixed adjacency matrix. Formally, let the input of the l-th layers in NoffGConv is $\mathbf{X}^{(l)} \in \mathbb{R}^{J \times D_l}$, J represents the number of nodes and D_l denotes input dimensions. The NoffGConv at the l-th layers can be formulated as the following:

$$\mathbf{X}^{(l+1)} = \sigma\left(\mathbf{W}_1\mathbf{X}^{(l)} + \mathbf{W}_2\mathbf{X}^{(l)}\tilde{\mathbf{A}}\right) \tag{1}$$

where σ is the activation function and $\tilde{\mathbf{A}}$ is the normalized adjacency matrix which is computed by $\tilde{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\tilde{\mathbf{D}}^{-\frac{1}{2}}$. $\tilde{\mathbf{D}}$ is a diagonal degree matrix. $\mathbf{A}$ is an adjacency matrix covering internal connections of $\mathcal{G}$. $\mathbf{I}$ is the identity matrix. With different weights $\mathbf{W}_1$ and $\mathbf{W}_2$, NoffGConv decouples the mapping of the node features and the offset features. Note that the vanilla GCN only has the second term in Equation 1, which assigns attention to the current node and its neighbors based on the degree matrix, weakening the guidance of its location information.

The Graphformer decoder contains N decoder blocks. Each block is composed of a NoffGConv layer followed by the standard self-attention layer and cross-attention layer. Our decoder takes learned joint queries as input, which represents positional embedding of joints. There is one-to-one matching between joint queries and hand joints, thus the Hungarian matching [3] is unnecessary. Additionally, since NoffGConv implements the nonlinear mapping of joint queries, we are able to remove the feed-forward network that usually follows the attention module.

3.3. Topology-Aware Head

The topological hand joints structure plays an essential role in predicting accurate hand pose especially in heavily self-occlusion and self-similarity cases. To overcome the lack of spatial structure cues in the existing an MLP head, we propose a Topology-aware head with the GCN technique. As mentioned above, GCN naturally provides a way to introduce the prior of hand kinematic topology. Then, the GCN aggregates information about the nodes and their corresponding neighbor nodes under the guidance of topology. However, the vanilla GCN has fixed attention to the connection between joints, which ignores the complex semantic relationship of neighboring nodes. Thus, we configure our Topology-aware head based on three semantic graph convolution layers (SemGConv) and a 1×1 convolution projection layer. Compared with vanilla GCN, SemGConv adds a learned weighting matrix $\mathbf{M} \in \mathbb{R}^{J \times J}$ to adaptively model connection strength between joints, which is written as:

$$\mathbf{X}^{(l+1)} = \sigma\left(\mathbf{W}\mathbf{X}^{(l)}\rho_i(\mathbf{M} \odot (\mathbf{A} + \mathbf{I}))\right) \tag{2}$$

where $\mathbf{W}$ is a transformation matrix; ρ_i is the Softmax nonlinearity which normalizes the weight of connections be-

Method	NYU	ICVL	MSRA	Hands2017			FPS
				AVG	SEEN	UNSEEN	
DenseReg [44]	10.21	7.24	7.23*	-	-	-	27.8
Pose-REN [4]	11.81	6.79	8.65	-	-	-	-
HandPointNet [10]	10.54	6.94	8.51	-	-	-	48
Point-to-Point [11]	9.05	6.33	7.71	-	-	-	41.8
V2V-PoseNet [30]	8.41	6.28	7.59	9.95	6.97	12.43	3.5
CrossInfoNet [8]	10.08	6.73	7.86	9.68	7.30	11.67	124.5
A2J [46]	8.61	6.46	-	8.57	6.92	9.95	105.6
SRN [37]	7.79	6.27	7.17	8.39	6.06	10.33	**263.1**
AWR [19]	7.48	5.98	7.20	7.48	5.21	9.36	-
PHG [35]	**7.39**	5.97	6.94	7.14	5.06	8.87	58.8
HandGCNFormer	7.43	**5.48**	**6.73**	**6.80**	**4.64**	**8.59**	72.8
PHG* [35]	6.75	5.94	5.82	-	-	-	58.8
HandGCNFormer*	**6.74**	**4.72**	**5.57**	**5.53**	**3.74**	**7.02**	72.8

Table 1: Comparisons with state-of-the-art methods on NYU, ICVL, MSRA, and Hands2017 using the mean of 3D distance error in millimeter. The "$*$" represents that the method adopts the average of the ground truth joints as hand region center for cropping images. "SEEN" and "UNSEEN" indicate the cases whether the test subjects are involved in training set. "AVG" denotes the mean of 3D distance error over all test frames. Best in **bold**.

tween a node i and the neighboring nodes $j \in \mathcal{N}(i)$; $\odot$ denotes elementwise multiplication.

Following the previous work [50], we leverage the residual connection to alleviate the over-smoothing problem during stacking multiple SemGConv layers. Furthermore, we stack the output embedding of all Graphformer decoder layers and fed them into our head module together, encouraging the network to implicitly extract the semantic information contained in different decoder layers. With excellent properties of SemGConv, our regression head constrains the pose to a more precise space guided by hand topology.

3.4. Overall Loss Function

For the pose estimation task, the distribution of prediction results is relatively sparse. Since the Laplace distribution is a more appropriate assumption for sparse data, the model is trained with a smooth L1 [19] loss to minimize the error between the estimated and ground truth poses. Both 2D and 3D poses are considered. Let $y_{2D} \in \mathbb{R}^{J\times 2}$ and $y_{3D} \in \mathbb{R}^{J\times 3}$ be the ground truth poses. The regression loss can be formulated as:

$$\mathcal{L}_{reg} = \sum_{n=1}^{N} \text{smooth}_{L1}\left(\hat{y}_{2D}^{n}, y_{2D}\right) + \text{smooth}_{L1}\left(\hat{y}_{3D}^{n}, y_{3\text{D}}\right) \tag{3}$$

where $\hat{y}_{3D}^{n}$ denotes the predicted 3D pose from the output of the n-th decoder layer. $\hat{y}_{2D}^{n}$ is calculated by projecting $\hat{y}_{3D}^{n}$ with camera intrinsics.

In addition, we apply an MLP on top of the ResNet backbone to predict a 3D initial pose, where the MLP is made of three fully connected layers. An auxiliary loss is employed to guide the backbone to learn stronger features and improve the overall performance, which is calculated as:

$$\mathcal{L}_{aux} = \text{smooth}_{L1}\left(\hat{p}_{2D}, y_{2\text{D}}\right) + \text{smooth}_{L1}\left(\hat{p}_{3D}, y_{3D}\right) \tag{4}$$

where $\hat{p}_{2D}$ and $\hat{p}_{3D}$ represent the 2D/3D coordinates corresponding to the initial pose, respectively.

Finally, the overall loss is the summation of the regression loss and auxiliary loss:

$$\mathcal{L}_{overall} = \mathcal{L}_{reg} + \mathcal{L}_{aux} \tag{5}$$

4. Experiments

4.1. Datasets

Hands2017 dataset [49] contains 957K training and 295K testing images. 21 hand joints are annotated.

NYU dataset [41] contains 72K training and 8.2K testing images labeled with 36 joint locations. Following the common convention [35, 30], we pick a subset of 14 joints from the frontal view for evaluation.

ICVL dataset [40] contains 22K training images and 1.6K testing images. The training data is augmented to 330K samples by leveraging in-plane rotation operations. The annotation of the pose contains 16 joints.

MSRA dataset [38] contains 76.5K images with 17 gestures. The ground truth pose annotates 21 joints. We evaluate this dataset with the common leave-one-subject-out cross-validation strategy [4, 19].

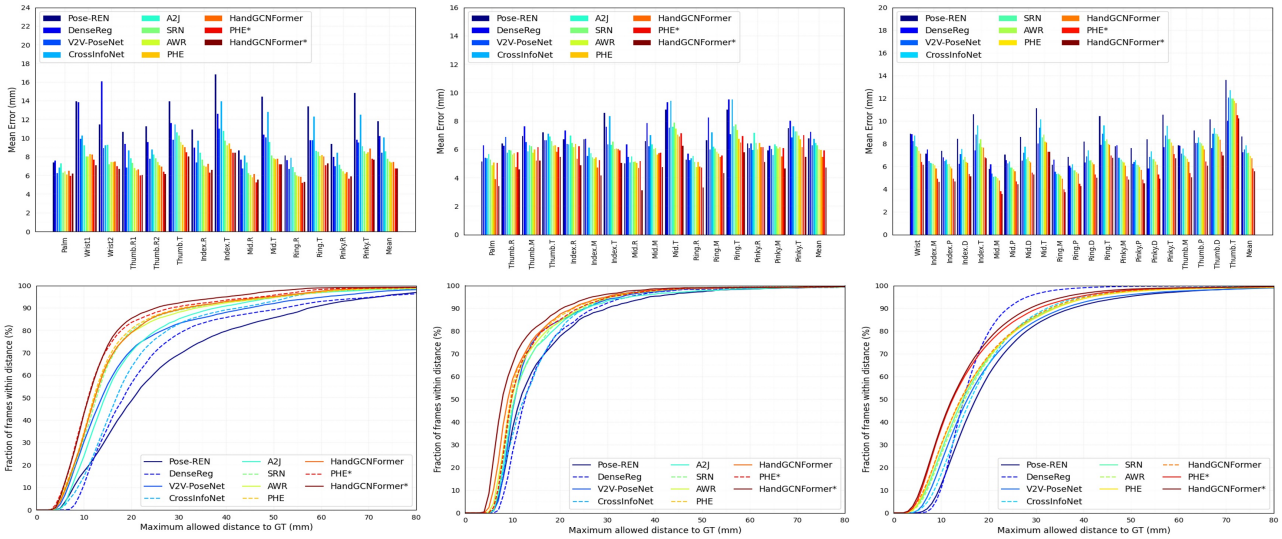


Figure 4: Comparison of our framework with the state-of-the-art works on NYU (left column), ICVL (middle column), and MSRA (right column) datasets. Top: The mean of 3D distance error for each joint. Bottom: The percentage of successful frames over different thresholds.

4.2. Experimental Settings

Implementation Details: We implement our model in an end-to-end fashion on one NVIDIA A100 Tensor Core GPU. Our method is trained with PyTorch framework using AdamW optimizer [25] with an initial learning rate of 0.0001. The training process covers 40 epochs. We leverage the multi-step learning rate schedule, which decays the learning rate by 0.1 at the 30th and 37th epoch respectively. The ResNet-50 is adopted as our backbone, which is pretrained on ImageNet and the rest of weights are initialized with Xavier init [12]. We adopt 8 heads for self-attention and four layers of Transformer encoder and Graphformer decoder. During inference, we utilize the prediction from the last decoder layer as the final result. Following the former works [37, 35, 19], we leverage the localization network proposed in V2V-poseNet [30] to get the center coordinates of the hand region in 3D space. The cropped images are resized to 256×256 and the depth value is normalized to $[-1, 1]$. We employ random scaling, random rotation and random translation for data augmentation in the world coordinate system. According to standard practice, we train one model for each benchmark with its own training set.

Evaluation Metrics: We evaluate our model with the same metrics adopted in former works: 1) the mean of 3D distance error and 2) the percentage of successful frames. The former is the average Euclidean distance error of per-joint between ground truth and predictions computed on the overall test set. The later represents the ratio of the number of successful frames in which all joint errors are below the threshold to the number of all test frames.

Baseline: Our baseline follows the DETR [3] framework without the Hungarian matching algorithm. The input queries of decoder correspond one by one to the hand joints. In addition, the baseline applies the same loss function as our method. Its architecture is detailed in supplementary material.

4.3. Comparison with the State-of-the-Art

We compare our HandGCNFormer with various existing methods [44, 4, 11, 10, 30, 8, 46, 37, 19, 35] on standard NYU, ICVL, MSRA and Hands2017 benchmarks. Table 1 shows the comparison results with the mean of 3D distance error as the metric. For a fair comparison, the results of previous work can be divided into two groups. The top group results adopt center coordinates provided by V2V-PoseNet as hand region center to crop images. The bottom group reports the results utilizing the average of the ground truth joints as hand region center, which is indicated by "$*$". Moreover, Figure 4 reports the per-joint mean error and the percentage of successful frames over different thresholds on NYU, ICVL, and MSRA datasets. The experimental results show that HandGCNFormer obtains comparable or superior performance to other methods achieving a real-time speed on a single GPU with 72.8 FPS. Note that the number of parameters in our model are also reduced by 7.6% compared to PHG which has 35.71M parameters.

Specifically, on **Hands2017 dataset**, our method outperforms other methods with the mean joint error of 6.80 mm. For unseen subjects hand, our method achieves the minimum mean joint error of 8.59 mm, essentially demonstrating the excellent generalization ability of our method. In addition, HandGCNFormer* improves 1.27mm compared

Method	AVG	SEEN	UNSEEN	Params(Flops)
Baseline	7.35	5.09	9.24	37.37M(5.81G)
+ Graphformer Decoder	6.94	4.77	8.74	33.18M(5.72G)
+ Topology-aware Head	6.90	4.67	8.77	37.24M(5.80G)
HandGCNFormer (+ both)	**6.80**	**4.64**	**8.59**	**33.04M(5.71G)**

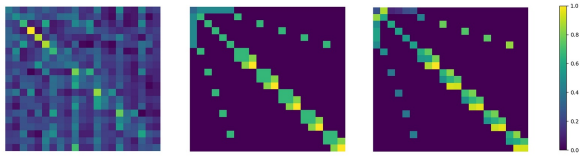
Table 2: Ablation study for the effectiveness of different modules in HandGCNFormer.

Method	AVG	SEEN	UNSEEN
Vanilla GCN [45]	6.95	4.83	8.73
ChebGConv(K=1) [5]	6.96	4.77	8.87
ChebGConv(K=2) [5]	6.94	4.83	8.69
SemGConv [50]	6.93	4.78	8.72
NoffGConv(Ours)	**6.80**	**4.64**	**8.59**

Table 3: Ablation study for the effectiveness of different GCN mothods in Graphformer decoder. K represents the order of convolution kernel in ChebGConv.

Method	AVG	SEEN	UNSEEN
N-S-C	**6.80**	**4.64**	**8.59**
S-N-C	6.86	4.67	8.69
S-C-N	6.86	4.68	8.68

Table 4: Ablation study for the effectiveness of different connection orders between three components in the Graphformer decoder. N, S, and C denote NoffGConv, self-attention and cross-attention, respectively.



(a) Self-attention (b) NofGConv (c) SemGConv

Figure 5: (a): The attention map of self-attention in decoder, dynamically models the global dependencies of joints. (b): Normalized adjacency matrix of NofGConv, focus on local topology perception with fixed connection strength between joints. (c): Learned weight matrix of SemGConv, adaptively models complex dependencies among neighboring joints.

with HandGCNFormer in the "AVG" test case, reflecting the fact that the accuracy of hand region center coordinates limits the performance of model. On **NYU dataset**, the results of our method are comparable to PHG. This is mainly because the annotations of the NYU dataset are noisy, which limits the performance of our method in terms of all-joint mean error. Even though, our method still obtains the best performance in terms of the percentage of successful frames as shown in lower left of Figure 4. On **ICVL dataset**, HandGCNFormer and HandGCNFormer* outperform the previous best results by a margin of 8.2% and 20.5%. In fact, HandGCNFormer achieves better accuracy than PHG*. For the per-joint error and the percentage of successful frames, our method significantly surpasses other methods under all the joints and thresholds. On **MSRA dataset**, our method is superior to PHG and PHG* by a margin of 3.0% and 4.3%, respectively. Our method reduces the per-joint error and achieves the optimal percentage of successful frames under 15mm threshold. Overall, HandGCNFormer is inherently superior to state-of-the-art methods, with a suitable trade-off between effectiveness and efficiency.

4.4. Ablation Study

In this section, we carry out extensive ablations to evaluate HandGCNFormer on Hands2017.

HandGCNFormer Modules: As shown in Table 2, we carry out experiments to quantify the contribution of our proposed modules (Graphformer decoder and Topology-aware head). Our baseline achieves a mean error of 7.35 mm on the "AVG" test item that denotes the mean of 3D distance error over all test frames, which is only slightly worse than PHG, reflecting that the Transformer framework can better capture long-range context information for hand pose estimation. Then, we replace the decoder of baseline with our Graphformer decoder. Benefiting from the synergy of NoffGConv and self-attention mechanism, the model with Graphformer decoder reduces the mean joint error by 0.41 mm and improves by 5.4% in terms of unseen subjects hand. Next, we only incorporate the Topology-aware head into baseline. The performance has significant gains, showing that spatial structure perception is essential to regress an accurate and robust pose. Additionally, it can be seen that our head achieves excellent performance without increasing the model parameters. Finally, HandGCNFormer, combining our decoder and regression head, achieves the best performance with the smallest model size. Particularly, HandGCNFormer outperforms baseline with a margin of 0.69 mm for unseen subjects hand, which shows that our method has advantages in generalization.

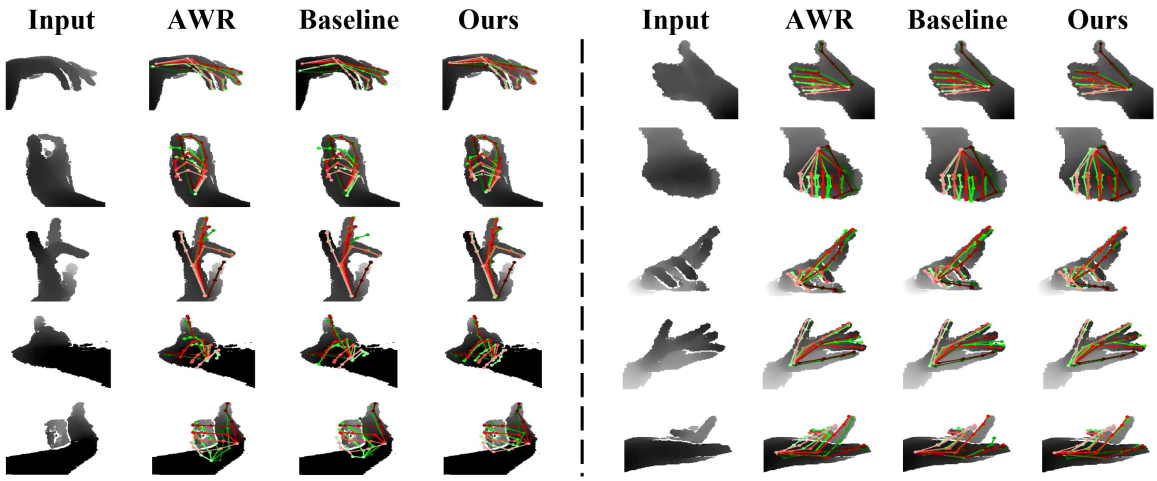


Figure 6: Qualitative comparison among AWR, our baseline, and our HandGCNFormer on Hands2017 dataset. Left: Qualitative results of images with self-occlusion. Right: Qualitative results of images with self-similarity. Red pose represents the ground truth. Green pose is predicted result.

NoffGConv: We compare NoffGConv with other GCN variations, including vanilla GCN [45], ChebGConv [5] and SemGConv [50]. Table 3 reports the comparison results, where K represents the order of convolution kernel in ChebGConv. Our method achieves superior performance than other methods, demonstrating the effectiveness of our NoffGConv in collaborating with self-attention. In addition, we compare three different connection orders between three components in the Graphformer decoder and report results in Table 4, where N, S, and C denote NoffGConv, self-attention, and cross-attention, respectively. "N-S-C" represents the structure of our decoder shown in Figure 2. "S-N-C" means NoffGConv is in the middle, and "S-C-N" means NoffGConv is following cross-attention. The experimental results show that "N-S-C" is the optimal order for fusing NoffGConv and attention modules.

4.5. Visualization

We visualize the weight matrices of self-attention in decoder, NoffGConv, and SemGConv during information aggregation. As illustrated in Figure 5, self-attention mechanism dynamically captures long-range dependencies between joints, but ignore inherent topology information of hand. NoffGConv and SemGConv focus on local connection relation of hand kinematic topology. Since self-attention learns the degree of dependencies between joints dynamically and flexibly, our NoffGConv assigns fixed attention to neighboring joints through a normalized adjacent matrix. In contrast, SemGConv exploits a learned weight matrix to adaptively extract complex relationships among neighboring joints, which provides richer spatial constraints for pose regression.

Figure 6 exhibits some qualitative results of self-occlusion and self-similarity samples on Hands2017. For a fair comparison, the results of AWR are reported at the same input size and the same hand region center as our method. It can be seen that HandGCNFormer achieves more accurate and plausible poses compared to AWR [19] and our strong baseline. Particularly, AWR will fail for the extreme self-occlusion case, whereas HandGCNFormer can successfully identify the location of joints and obtain a more plausible pose guided by global understanding of input data and the prior knowledge of hand topology.

5. Conclusion

In this paper, we propose a novel Topology-aware Transformer network named HandGCNFormer to infer plausible and accurate 3D hand poses. In HandGCNFormer, we design a Graphformer decoder and a Topology-aware head to maximize the synergy of Transformer and GCN. Our method comprehensively models the global understanding of image and joints as well as the intrinsic hand kinematic topology, effectively reducing ambiguities caused by invisible and similar joints. Extensive experimental results demonstrate that HandGCNFormer achieves state-of-the-art performances on four public datasets and significantly reduces the prediction error in complex scenarios. Benefiting GCN incorporated Transformer network, our method can also be easily generalized to other regression tasks of structured data.

Acknowledgement. This research was supported by Strategic Priority Research Program of Chinese Academy of Sciences (XDC08050100), Shanghai Municipal Science and Technology Major Project (ZHANGJIANG LAB) under Grant 2018SHZDZX01 and Shanghai Academic Research Leader (22XD1424500).

References

[1] Ruwen Bai, Min Li, Bo Meng, Fengfa Li, Junxing Ren, Miao Jiang, and Degang Sun. Gcst: Graph convolutional skeleton transformer for action recognition. *arXiv preprint arXiv:2109.02860*, 2021.

[2] Razvan Caramalau, Binod Bhattarai, and Tae-Kyun Kim. Active learning for bayesian 3d hand pose estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3419–3428, 2021.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[4] Xinghao Chen, Guijin Wang, Hengkai Guo, and Cairong Zhang. Pose guided structured region ensemble network for cascaded hand pose estimation. *Neurocomputing*, 395:138–149, 2020.

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

[6] Bardia Doosti, Shujon Naha, Majid Mirbagheri, and David J Crandall. Hope-net: A graph-based model for hand-object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6608–6617, 2020.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[8] Kuo Du, Xiangbo Lin, Yi Sun, and Xiaohong Ma. Crossinfonet: Multi-task information sharing based hand pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9896–9905, 2019.

[9] Linpu Fang, Xingyan Liu, Li Liu, Hang Xu, and Wenxiong Kang. Jgr-p2o: Joint graph reasoning based pixel-to-offset prediction network for 3d hand pose estimation from a single depth image. In *European Conference on Computer Vision*, pages 120–137. Springer, 2020.

[10] Liuhao Ge, Yujun Cai, Junwu Weng, and Junsong Yuan. Hand pointnet: 3d hand pose estimation using point sets. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8417–8426. IEEE, 2018.

[11] Liuhao Ge, Zhou Ren, and Junsong Yuan. Point-to-point regression pointnet for 3d hand pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 475–491, 2018.

[12] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[13] Hengkai Guo, Guijin Wang, Xinghao Chen, and Cairong Zhang. Towards good practices for deep 3d hand pose estimation. *arXiv preprint arXiv:1707.07248*, 2017.

[14] Hengkai Guo, Guijin Wang, Xinghao Chen, Cairong Zhang, Fei Qiao, and Huazhong Yang. Region ensemble network: Improving convolutional network for hand pose estimation. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 4512–4516. IEEE, 2017.

[15] Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent Lepetit. Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11090–11100, 2022.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[17] Lin Huang, Jianchao Tan, Ji Liu, and Junsong Yuan. Hand-transformer: Non-autoregressive structured modeling for 3d hand pose estimation. In *European Conference on Computer Vision*, pages 17–33. Springer, 2020.

[18] Lin Huang, Jianchao Tan, Jingjing Meng, Ji Liu, and Junsong Yuan. Hot-net: Non-autoregressive transformer for 3d hand-object pose estimation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3136–3145, 2020.

[19] Weiting Huang, Pengfei Ren, Jingyu Wang, Qi Qi, and Haifeng Sun. Awr: Adaptive weighting regression for 3d hand pose estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11061–11068, 2020.

[20] Deying Kong, Haoyu Ma, and Xiaohui Xie. Sia-gcn: A spatial information aware graph neural network with 2d convolutions for hand pose estimation. *arXiv preprint arXiv:2009.12473*, 2020.

[21] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1944–1953, 2021.

[22] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13147–13156, 2022.

[23] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1954–1963, 2021.

[24] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[26] Meysam Madadi, Sergio Escalera, Xavier Baró, and Jordi Gonzàlez. End-to-end global to local convolutional neural

network learning for hand pose recovery in depth data. *IET Computer Vision*, 16(1):50–66, 2022.

[27] Jameel Malik, Ibrahim Abdelaziz, Ahmed Elhayek, Soshi Shimada, Sk Aziz Ali, Vladislav Golyanik, Christian Theobalt, and Didier Stricker. Handvoxnet: Deep voxel-based network for 3d hand shape and pose estimation from a single depth map. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7113–7122, 2020.

[28] Weian Mao, Yongtao Ge, Chunhua Shen, Zhi Tian, Xinlong Wang, and Zhibin Wang. Tfpose: Direct human pose estimation with transformers. *arXiv preprint arXiv:2103.15320*, 2021.

[29] Angel Martínez-González, Michael Villamizar, and Jean-Marc Odobez. Pose transformers (potr): Human motion prediction with non-autoregressive transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2276–2284, 2021.

[30] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *Proceedings of the IEEE conference on computer vision and pattern Recognition*, pages 5079–5088, 2018.

[31] Gyeongsik Moon, Ju Yong Chang, Yumin Suh, and Kyoung Mu Lee. Holistic planimetric prediction to local volumetric prediction for 3d human pose estimation. *arXiv preprint arXiv:1706.04758*, 2017.

[32] Markus Oberweger and Vincent Lepetit. Deepprior++: Improving fast and accurate 3d hand pose estimation. In *Proceedings of the IEEE international conference on computer vision Workshops*, pages 585–594, 2017.

[33] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015.

[34] Lingteng Qiu, Xuanye Zhang, Yanran Li, Guanbin Li, Xiaojun Wu, Zixiang Xiong, Xiaoguang Han, and Shuguang Cui. Peeking into occluded joints: A novel framework for crowd pose estimation. In *European Conference on Computer Vision*, pages 488–504. Springer, 2020.

[35] Pengfei Ren, Haifeng Sun, Jiachang Hao, Qi Qi, Jingyu Wang, and Jianxin Liao. Pose-guided hierarchical graph reasoning for 3-d hand pose estimation from a single depth image. *IEEE Transactions on Cybernetics*, 2021.

[36] Pengfei Ren, Haifeng Sun, Weiting Huang, Jiachang Hao, Daixuan Cheng, Qi Qi, Jingyu Wang, and Jianxin Liao. Spatial-aware stacked regression network for real-time 3d hand pose estimation. *Neurocomputing*, 437:42–57, 2021.

[37] Pengfei Ren, Haifeng Sun, Qi Qi, Jingyu Wang, and Weiting Huang. Srn: Stacked regression network for real-time 3d hand pose estimation. In *BMVC*, page 112, 2019.

[38] Xiao Sun, Yichen Wei, Shuang Liang, Xiaoou Tang, and Jian Sun. Cascaded hand pose regression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 824–832, 2015.

[39] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018.

[40] Danhang Tang, Hyung Jin Chang, Alykhan Tejani, and Tae-Kyun Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3786–3793, 2014.

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

[42] Anirudh Tunga, Sai Vidyaranya Nuthalapati, and Juan Wachs. Pose-based sign language recognition using gcn and bert. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 31–40, 2021.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[44] Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao. Dense 3d regression for hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5147–5156, 2018.

[45] Max Welling and Thomas N Kipf. Semi-supervised classification with graph convolutional networks. In *J. International Conference on Learning Representations (ICLR 2017)*, 2016.

[46] Fu Xiong, Boshen Zhang, Yang Xiao, Zhiguo Cao, Taidong Yu, Joey Tianyi Zhou, and Junsong Yuan. A2j: Anchor-to-joint regression network for 3d articulated pose estimation from a single depth image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 793–802, 2019.

[47] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[48] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Towards explainable human pose estimation by transformer. *arXiv preprint arXiv:2012.14214*, 2(6), 2020.

[49] Shanxin Yuan, Qi Ye, Guillermo Garcia-Hernando, and Tae-Kyun Kim. The 2017 hands in the million challenge on 3d hand pose estimation. *arXiv preprint arXiv:1707.02237*, 2017.

[50] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3425–3435, 2019.

[51] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.