This document is downloaded from DR-NTU (https://dr.ntu.edu.sg)
Nanyang Technological University, Singapore.

Exploiting spatial-temporal relationships for 3D pose estimation via graph convolutional networks

Cai, Yujun; Ge, Liuhao; Liu, Jun; Cai, Jianfei; Cham, Tat-Jen; Yuan, Junsong; Thalmann, Nadia Magnenat

2019

Cai, Y., Ge, L., Liu, J., Cai, J., Cham, T.-J., Yuan, J., & Thalmann, N. M. (2019). Exploiting spatial-temporal relationships for 3D pose estimation via graph convolutional networks. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2272-2281. doi:10.1109/ICCV.2019.00236

https://hdl.handle.net/10356/86102

https://doi.org/10.1109/ICCV.2019.00236

© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: https://doi.org/10.1109/ICCV.2019.00236

Downloaded on 19 Mar 2024 21:42:48 SGT

Exploiting Spatial-temporal Relationships for 3D Pose Estimation via Graph Convolutional Networks *

Yujun Cai[1], Liuhao Ge[1], Jun Liu[1], Jianfei Cai[1,2], Tat-Jen Cham[1], Junsong Yuan[3], Nadia Magnenat Thalmann[1]
[1]Nanyang Technological University, Singapore
[2]Monash University, Australia
[3]State University of New York at Buffalo University, Buffalo, NY, USA
{yujun001, ge0001ao, jliu029}@e.ntu.edu.sg
{asjfcai, astjcham}@ntu.edu.sg, jsyuan@buffalo.edu, nadiathalmann@ntu.edu.sg

Abstract

Despite great progress in 3D pose estimation from single-view images or videos, it remains a challenging task due to the substantial depth ambiguity and severe self-occlusions. Motivated by the effectiveness of incorporating spatial dependencies and temporal consistencies to alleviate these issues, we propose a novel graph-based method to tackle the problem of 3D human body and 3D hand pose estimation from a short sequence of 2D joint detections. Particularly, domain knowledge about the human hand (body) configurations is explicitly incorporated into the graph convolutional operations to meet the specific demand of the 3D pose estimation. Furthermore, we introduce a local-to-global network architecture, which is capable of learning multi-scale features for the graph-based representations. We evaluate the proposed method on challenging benchmark datasets for both 3D hand pose estimation and 3D body pose estimation. Experimental results show that our method achieves state-of-the-art performance on both tasks.

1. Introduction

3D pose estimation that involves estimating 3D joint locations of a human hand or body from single-view images or videos is a fast-growing research area and has aroused long-standing research attention in the past decades[11, 47, 48], since it plays a significant role in numerous applications such as gesture recognition, robotics and human-computer interactions. Despite the tremendous success achieved in recent years [8, 27, 28, 38, 44, 5, 16, 49, 25, 13], it remains a challenging problem due to the frequent self-occlusions and substantial depth ambiguity in 2D representations.

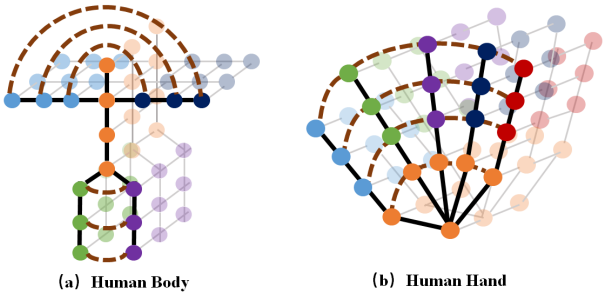


Figure 1. Graphical spatial-temporal dependencies between different joints of (a) full human body, and (b) human hand. The temporal edges connect the same joints between consecutive frames and the spatial edges represent the natural connections of each frame. For easy illustration, we only plot the whole spatial connections on the front frame of the spatial-temporal graph, including the direct physical connections (solid line) and the indirect "symmetrical" relations (dashed curve). We color-code the joints to show different parts of the human body (hand).

Many existing works [3, 12, 17, 29, 54, 53, 15, 14] rely on effective 2D pose estimation frameworks to first localize the 2D keypoints on the image plane, and then lift 3D poses from the estimated 2D joint positions. Additionally, recent works [12, 17, 29] have shown that well-designed deep networks can achieve competitive performance in 3D pose estimation using only 2D joint detections as input. However, it is worth noting that estimating 3D poses from 2D representations is inherently an ill-posed problem, since there may exist multiple valid 3D interpretations for a single 2D skele-

*This research is supported by the BeingTogether Centre, a collaboration between Nanyang Technological University (NTU) Singapore and University of North Carolina (UNC) at Chapel Hill. The BeingTogether Centre is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative. This research is also supported in part by Singapore MoE Tier-2 Grant (MOE2016-T2-2-065) and start-up funds from University at Buffalo.

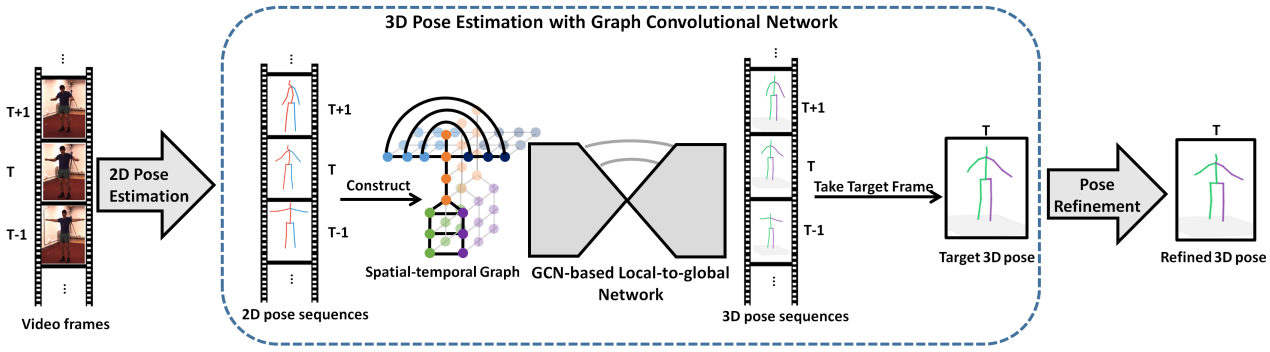


Figure 2. Schematic overview of our proposed network architecture for 3D pose estimation from consecutive 2D poses. The input is a small number of adjacent 2D poses estimated from RGB images and the output is the 3D joint locations of the target frame. We construct a spatial-temporal graph on skeleton sequences and design a hierarchical "local-to-global" architecture with graph convolutional operations to effectively process and consolidate features across scales. To further refine the estimation results, a pose-refinement process is applied which can be trained end-to-end with the graph convolutional network. Note that this pipeline is applicable for both 3D human body and hand pose estimation and here we simply take 3D human body pose estimation as a visualization example.

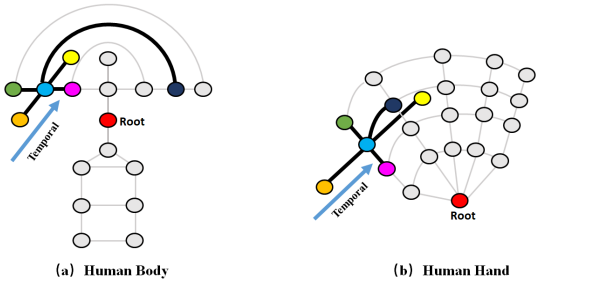
ton, which makes it difficult to infer a unique valid solution, especially for cases with severe occlusions. To overcome this ambiguity, several methods [4, 51, 12] attempted to embed kinematic correlations to ensure the spatial validity of the 3D structures. For instance, Fang *et al.* [12] explicitly incorporated geometric dependencies among different body parts by enforcing spatial consistency over the estimated 3D human poses. Moreover, to deal with the incoherent and jittery predictions, some work [17, 39, 30] turned to exploit the temporal information across sequences. For example, Hossain *et al.* [17] designed a sequence-to-sequence network to predict 3D joint locations and imposed temporal smoothness constraints during training to ensure the temporal consistency over a sequence.

Despite their promising results, we observe that most of the existing work only focus on incorporating either spatial configuration constraints or temporal correlations, while ignoring the complementarity between these two types of information. More precisely, we note that having priors on the spatial dependencies can reduce the possibility of generating physically impossible 3D structures and alleviate the problem of self-occlusions, while utilizing temporal inference helps resolve the challenging issues such as depth ambiguity and visible jitters. These observations encourage us to develop a method that can effectively embed both spatial and temporal relationships into a learning-based framework, and leverage it for 3D pose estimation.

Motivated by the natural graph-based representation for a series of skeletal forms and inspired by recent advances in graph convolution networks (GCNs) [9, 20, 41, 50], in this work, we propose to utilize GCNs to exploit spatial and temporal relationships for 3D pose estimation. Note that different from the two recent papers [15, 26] that either uses uniform GCN for dense hand mesh reconstruction or considers spatial graph-lstm, our work uses GCN for spatial-temporal graph with semantic grouping for sequential 3D pose estimation. Specifically, as depicted in Figure 1, we define the sequence of skeletal joints as a spatial-temporal graph. The graph topology is formed with joints as the graph nodes, linked by two types of connections: spatial edges that represent spatial dependencies among different joints, and temporal edges that connect the same joint across neighboring frames. To deal with sparse connections and functionally-variant graph edges for 3D pose estimation, we propose to learn different convolutional kernel weights for different neighborhood types, while the generic graph convolutional operatinos uniformly treat the neighboring nodes at the same degree with shared kernel weights. Moreover, inspired by the previous 2D pose estimation approach [32] that processed and consolidated information at multiple resolutions, we analogously propose a graph-convolutional "local-to-global" hierarchical network architecture that captures multi-scale features, where our graph pooling and upsampling layers are designed based on the interpretable human body (or hand) configurations. Finally, a pose refinement step is introduced to further improve the estimation accuracy (see Figure 2 for system overview).

The contributions of this work are threefold:

- By treating a sequence of skeletons as a spatial-temporal graph, we propose to use GCN to effectively exploit the spatial configurations and temporal consistencies for 3D pose estimation, both of which are significant for improving the 3D pose estimation accuracy.
- We design a local-to-global network architecture, which is capable of learning multi-scale graph features via successive graph pooling and upsampling layers. Experimental results demonstrate the benefits of such



(a) Human Body (b) Human Hand

Figure 3. Visualization of different neighboring nodes for (a) human body and (b) human hand. The neighboring nodes are divided into six classes according to their sematic meanings: center node (blue), physically-connected nodes including the one closer (purple) to and the one farther (green) from the skeleton root, indirect "symmetrically"-related node (dark blue), time forward node (yellow), and time backward node (orange).

hierarchical architecture that can effectively consolidate the local and global features in our network.

- We propose a non-uniform graph convolutional strategy based on the generic graph convolutional operations, which learns different convolutional kernel weights for different neighboring nodes according to their semantic meanings. Experiments show that the proposed graph convolutional strategy is crucial for performance improvement with the constructed sparse spatial-temporal graph for 3D pose estimation.

We conduct comprehensive experiments on two widely-used benchmarks: the Human3.6M dataset [18] for 3D human body pose estimation and the STB dataset [52] for 3D hand pose estimation. Experimental results show that our proposed method achieves state-of-the-art performance on both tasks.

2. Related Work

3D Pose Estimation. Different aspects of learning-based human hand (and body) pose estimation have been explored in the past few years, which can be roughly classified into two categories: i) directly regressing the 3D locations of each joint from 2D images; ii) decoupling 3d pose estimation into the 2D pose estimation and 3D pose estimation from 2D joint detections.

For the first category, Li and Chan [24] designed a multi-task framework that jointly learns pose regression and body part detectors. Park *et al.* [36] introduced an end-to-end framework with simultaneous training of both 2D joint classification and 3D joint regression. Pavlakos *et al.* [38] introduced a deep convolutional neural network based on the stacked hourglass architecture, with a fine discretization of the 3D space to predict per voxel likelihoods for each joint.

For the second category, Martinez *et al*. [28] directly regressed 3D keypoints from extracted 2D poses via a simple network composed of several fully-connected layers. Zimmermann *et al.* [54] adopted a PoseNet module to localize the 2D hand joint locations, from which the most likely 3D structure of the hand was then estimated. To incorporate spatial priors into the framework, Fang [12] developed a deep grammar network to explicitly encode the human body dependencies and relations. Moreover, to deal with the depth ambiguity and visual jitters in static image, Hossain *et al.* [17] utilized the temporal information by propagating joint position information across frames based on a sequence-to-sequence model. The performance gain achieved by these methods motivates us to take a follow-up exploration towards the incorporation of both spatial and temporal dependencies, instead of only focusing on one aspect. Specifically, our approach learns the spatial-temporal information implicitly by combining graph convolutional operations with the domain-specific knowledge for 3D pose estimation.

Graph Convolutional Neural Network (GCN). GCNs are deep learning based methods that perform convolutional operations on graphs. Compared with traditional CNN, GCN has its unique convolutional operators for irregular data structures. In general, GCNs can be divided into two categories: spectral based GCN [9, 20, 22, 23, 41] and non-spectral based GCN [1, 2, 10]. The latter attempts to expand the spatial definition of a convolution by rearranging graph vertices into a certain grid form so as to directly apply conventional convolutional operations, while the former performs the convolutional process with Fourier transformation. Usually spectral GCN is good for handling graphs with fixed topology while non-spectral GCN can handle topology-varied graphs.

3. Methodology

Overview. Figure 2 depicts an overview of our proposed network architecture. Given a small number of adjacent 2D joint locations of a hand (or body) estimated from video frames as input, we aim at predicting a target frame's 3D joint locations $\mathbf{\Phi} = \{\phi_i\}_{i=1}^{M} \in \mathbf{\Lambda}_{3D}$ in the camera coordinate system, where M is the number of joints, and $\mathbf{\Lambda}_{3D}$ is the $M \times 3$ dimensional hand joint space. In particular, we construct a spatial-temporal graph with the joints as graph nodes and the local connectivities in the spatial (skeleton structure) and temporal domains as graph edges. To effectively learn the multi-scale features of the graph-based representation, a hierarchical "local-to-global" scheme is introduced into the framework, which takes successive steps of pooling and upsampling before generating the 3D predictions. Lastly, a pose-refinement process is added to further refine the 3D pose estimation. The whole model is trained in an end-to-end manner with backpropagation. Next, we will describe the individual components in details.

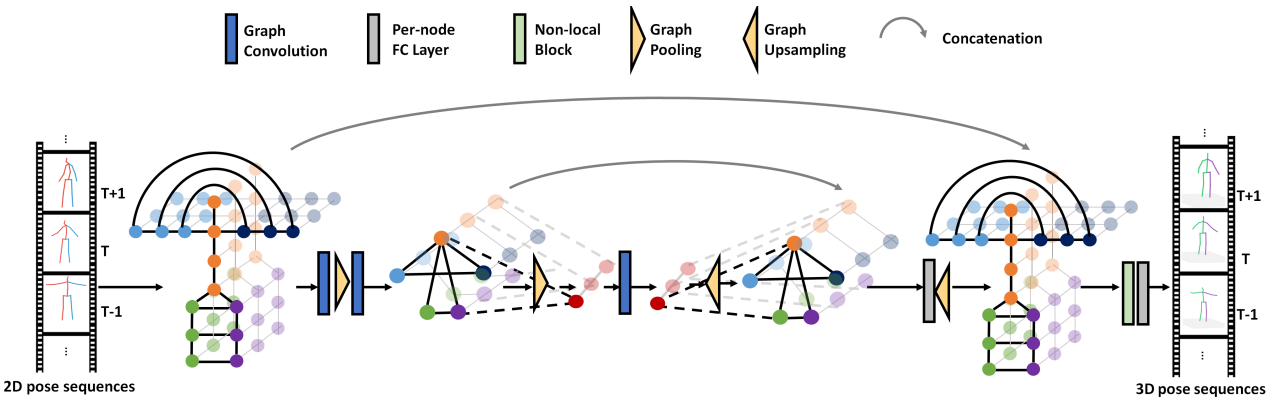


Figure 4. Illustration of the "local-to-global" network architecture, which is able to effectively process and consolidate features across scales. For convenience of illustration, we only plot the whole spatial connections on the front frame of the spatial-temporal graphs.

Spatial-temporal Graph Construction. A skeleton sequence can be naturally organized as a spatial-temporal graph representation. Specifically, we define a pose sequence as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$, where $\mathcal{V} = \{v_{ti} | t = 1, ...T; i = 1, ...M\}$ denotes a set of vertices, corresponding to T frames and M body joints per frame, $\mathcal{E} = \{e_{ij}\}$ is the set of edges, indicating the connections between nodes, and $W = (w_{ij})_{N \times N}$ with $N = MT$ is the adjacency matrix, with $w_{ij} = 0$ if $(i, j) \notin \mathcal{E}$, and $w_{ij} = 1$ if $(i, j) \in \mathcal{E}$. The normalized graph Laplacian [7] is computed as $L = I_N - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where $D^{ii} = \sum_j W^{ij}$. The edge set consists of two parts: temporal connections that link each joint with its counterpart in the neighboring frames, and spatial connections that include both direct and indirect kinematic dependencies in each frame (see Figure 1).

3.1. Revisiting Graph Convolutional NNs

In this work, we adopt a spectral-based GCN, since it works well with structured graphs with predefined topology. In particular, the spectral convolutions on graphs [41] can be considered as the multiplication of a signal $x \in \mathbb{R}^N$ with a filter $g_\theta = diag(\theta)$ in Fourier domain:

$$g_\theta * x = U g_\theta U^T x, \tag{1}$$

where graph Fourier basis U is the matrix of the eigenvectors of the normalized graph Laplacian L, and $U^T x$ denotes the graph Fourier transform of x.

To reduce the computational complexity, Kipf and Welling [20] introduced a layer-wise linear formulation defined by stacking multiple localized graph convolutional layers with the first-order approximation of graph Laplacian:

$$Z = \widetilde{D}^{-\frac{1}{2}} \widetilde{W} \widetilde{D}^{-\frac{1}{2}} X \Theta, \tag{2}$$

where the input signal $X \in \mathbb{R}^{N \times C}$ is a generalized one, representing the C-dim features of N vertices on the graph, $\Theta \in \mathbb{R}^{C \times F}$ is the matrix of filter parameters, $\widetilde{W}$ and $\widetilde{D}$ are the normalized versions with $\widetilde{W} = W + I_N$ and $\widetilde{D}^{ii} = \sum_j \widetilde{W}^{ij}$, and $Z \in \mathbb{R}^{N \times F}$ is the convolved signal matrix.

3.2. Graph Convolution for Pose Estimation

In the existing graph convolution (Eq. (2)), essentially each kernel Θ is shared by all the 1-hop neighboring nodes. This works fine for dense graph. However, our spatial-temporal graph for 3D pose estimation is sparse with functionally-variant graph edges (*e.g.*, spatial edges and temporal edges representing different correlations), for which a uniform treatment of neighboring nodes is not suitable.

To tackle this issue, inspired from the previous studies [33, 50] that take the convolutional operator with a larger kernel size, we made modifications to the generic graph convolutional operations. In particular, we classify neighboring nodes according to their semantic meanings and use different kernels for different neighboring nodes. As presented in Figure 3, the neighboring nodes are divided into six classes based on intuitive interpretations: 1) the center node itself; 2) a physically-connected neighboring node that is closer to the root node than the center node; 3) a physically-connected neighboring node that is farther from the root node than the center node; 4) an indirect "symmetrically-related" neighboring node; 5) a time-forward neighboring node; and 6) a time-backward neighboring node. Based on the classification, the graph convolution in (2) is updated to:

$$Z = \sum_k D_k^{-\frac{1}{2}} W_k D_k^{-\frac{1}{2}} X \Theta_k, \tag{3}$$

where k is the index of the neighbor types, and Θ_k is the filter matrix for the k-th type 1-hop neighboring nodes. Note that here $\widetilde{W}$ is dismantled into k sub-matrices with $\widetilde{W} = \sum_k W_k$, and $D_k^{ii} = \sum_j W_k^{ij}$.

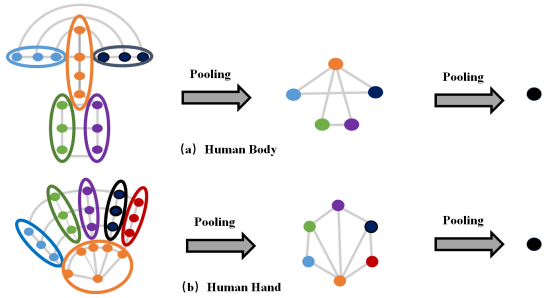


Figure 5. The defined hierarchical graph pooling strategy for (a) human body and (b) human hand. Given the original graph structure per-frame, we first divide the nodes into individual subsets based on the interpretable skeleton structure, which are represented with the same color, and then perform the max-pooling operations on each of the subsets. Next, the coarsened graph is max-pooled into one node which contains the global information of the whole skeleton. Note that in the subsequent top-down processing, upsampling is performed as the reverse operation of the proposed pooling, which allocates the features of a vertex in the coarser graph to its children vertices in the finer graph.

3.3. GCN-based Local-to-global Prediction

A design choice that has been particularly effective for pose estimation is capturing visual patterns or semantics at different resolutions in a feed-forward fashion. Bottom-up processing is first performed by subsampling the feature maps, and then top-down processing is conducted by upsampling the feature maps with the combination of higher resolution features from bottom layers, as proposed in the Stacked Hourglass network [32] for 2D pose estimation. Inspired by the success of such hierarchical architectures, we propose a conceptually similar "local-to-global" scheme, which aims at learning multi-scale features but from the graph-based representations.

Graph Pooling and Upsampling: For graph-based representations, the pooling operation requires meaningful neighborhoods on graphs, where similar vertices are clustered together. In this work, we propose to gradually cluster the whole skeleton per frame based on interpretable human body (or hand) configurations, as specified in Figure 5. For the top-down process, the upsampling procedure simply takes a reverse step of the graph pooling procedure, where the features of vertices in the coarser graph are duplicated to the corresponding child vertices in the finer scale. In addition, the temporal links remain the same throughout the different abstraction levels, connecting each node with its counterparts in neighboring frames.

Hierarchical Architecture: Figure 4 shows the proposed hierarchical "local-to-global" network, which can effectively process and consolidate features across scales. In the earlier stage, we gradually perform the graph convolution and pooling operations from the original scale to a very low resolution. Thereafter, the network conducts a top-down process with a sequence of upsampling and combining of features across scales. To utilize both bottom-up and top-down features, we perform an element-wise concatenation for features with the same scale, followed by a per-node FC layer to update the combined features. Furthermore, a non-local block [45] is introduced before generating the 3D pose sequences to facilitate a holistic processing of the full body.

3.4. Pose Refinement

For the 3D pose estimation task, there are two types of widely-used 3D pose representations. The first uses root-relative 3D coordinates of the joints in the camera coordinate system, while the second involves concatenating the predicted depths of each joint and the UV coordinates extracted from 2D detectors. These two representations can be easily converted from one to the other using the camera intrinsic matrix.

For relatively accurate 2D pose , the second representation is preferred since it guarantees the consistency between the predicted 3D pose and the 2D projections on the image plane. However, for poor 2D pose , maintaining the consistency between the projections and the 3D pose often leads to a physically invalid 3D pose structure; here the first representation is better as it is more capable of generating a valid 3D pose structure. To strike a balance between the two circumstances, we design a simple two-layer fully-connected network for pose refinement, which takes the 3D pose estimation results in both representations (where the depth values in the second representation are directly computed from the first) as the input, and output the confidence values for the two sets of results. Finally, the refined 3D joint locations are computed as the confidence-weighted sum of the two sets of estimation results.

3.5. Training

We use the following losses in training.

3D Pose Loss. $L_p = \sum_{t=1}^{T}\sum_{i=1}^{M}\left\|\hat{\phi}_{t,i} - \phi_{t,i}\right\|_2^2$, where $\hat{\phi}_{t,i}$ and $\phi_{t,i}$ represent the estimated and ground truth 3D joint locations of joint i at time t, respectively

Derivative Loss. Similar to [17], we adopt a derivative loss L_d to enforce temporal smoothness. Considering that joints located at limb terminals commonly move faster than other joints, we divide the joints of a human body into three sets: torso head, limb mid and limb terminal, while for the human hand we divide the 21 joints into: palm root, finger mid and finger terminal. Mathematically, the derivative loss L_d is defined as

$$L_d = \sum_{t=2}^{T}\sum_{i=1}^{M}\sum_{s\in S}\eta_s\left\|\hat{\phi}^s_{t,i} - \hat{\phi}^s_{t-1,i}\right\|_2^2, \tag{4}$$

where $\hat{\phi}^s_{t,i}$ denotes the the predicted 3D locations of joints belonging to the set s, and η_s is the scalar hyper-parameter controlling the significance of each set, where a higher value is assigned to the set of joints that are generally more stable than others.

Symmetry Loss L_s. It is defined for penalizing the differences in the lengths of left and right bone pairs, as is typically employed in 3D body pose estimation. Mathematically, L_s can be written as

$$L_s = \sum_{t=1}^{T} \sum_{b} \left\| \hat{B}_{t,b} - \hat{B}_{t,C(b)} \right\|_2^2 \tag{5}$$

where $\hat{B}_{t,b}$ is the estimated bone length for a right-side bone b and $C(b)$ is the corresponding left-side bone.

Training strategy. In our implementation, we first train the network prior to the pose refinement layers using the 3D pose loss L_p, which generates consecutive 3D joint locations from input 2D pose sequences. We then train the entire network in an end-to-end manner with the combined loss:

$$L = \lambda_p L_p + \lambda_d L_d + \lambda_s L_s \tag{6}$$

where $\lambda_p = 1, \lambda_d = 1$ and $\lambda_s = 0.01$. Note that the pose loss L_p and the symmetry loss L_s are applied on all of the 3D pose estimation results, including all intermediate 3D pose predictions and the final refined 3D joint locations. The derivative loss is only applied on the consecutive 3D joint estimation before pose refinement.

4. Experiments

4.1. Implementation Details

In our experiments, we first feed the input 2D skeletons into a batch normalization layer to keep the consistency of the input data, which are then passed to our proposed hierarchical "local-to-global" network. Specifically, we employ six graph convolutional layers during the bottom-up process, with 3, 2 and 2 layers for the three graph resolutions. For the top-down process, we deploy a per-node fully-connected operation for each stage of the feature concatenation to get the consecutive 3D joint locations and then choose the target frame 3D pose estimation. Finally, we feed the estimation results into a pose refinement network which is composed of two fully-connected layers with 1024 hidden units followed by a ReLU function. For better understanding, detailed diagrams of our network architecture can be found in our supplementary materials.

We implement our method within the PyTorch framework. For the first training stage described in Section 3.5, we train for 60 epochs with a mini-batch size of 256 using the Amsgrad optimizer. The learning rate starts from 0.001, with a shrink factor of 0.95 applied after each epoch and 0.5 after every 10 epochs. For the second stage, we set $\lambda_p = 1, \lambda_d = 1$ and $\lambda_s = 0.01$ and train for 20 epochs with the learning rate of 5×10^{-6}. All experiments were conducted on one GeForce GTX 1080 GPU with CUDA 8.0.

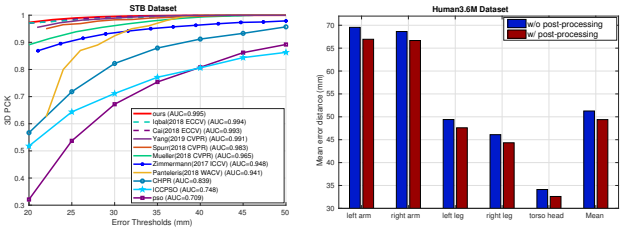


Figure 6. Left: Comparisons of the 3D PCK results with the state-of-the-art methods on STB for 3D hand pose estimation. Right: The impact of the pose refinement on the mean error distances of different body parts on Human3.6M.

4.2. Datasets

We evaluate our method on two publicly available datasets: the Human3.6M dataset [18] for 3D human body pose estimation, and STB [52] for 3D hand pose estimation.

Human3.6M. The Human3.6M dataset [18] is a large-scale and commonly used dataset for 3D human pose estimation, which consists of 3.6 million images captured from 4 different cameras, with 11 subjects performing a variety of actions, such as "Walking", "Sitting" and "Smoking". The 3D pose ground truth and all camera parameters (including intrinsic and extrinsic parameters) are provided in this dataset. In this research, we follow the evaluation protocols in prior work [17, 21, 26, 28, 37, 39], in which 5 subjects (S1, S5, S6, S7, S8) are used for training and 2 subjects (S9 and S11) are adopted for testing. All camera views are trained with a single model for all actions. We perform 2D pose detections using the cascaded pyramid network (CPN) [6] which is an extension of FPN, as proposed in [39].

STB Dataset. The STB (Stereo Hand Pose Tracking Benchmark) dataset [52] is a real world dataset captured under varying illumination conditions with 6 different backgrounds. Both 2D and 3D annotations of the total 21 hand keypoints are provided for each frame. We follow the same training and evaluation protocol used in [3, 54], training on 10 sequences and testing on the other two, with the Convolutional Pose Machine [46] used for detecting the 2D joint locations.

4.3. Evaluation Metrics

For Human3.6M, we report the mean per joint position error (MPJPE) as the evaluation metric, which calculates the average Euclidean distance of the estimated joints to ground truth after the alignment of the root joint (central hip). This protocol is referred to as protocol #1. In some work, an alternative metric is adopted, where the estimated

Protocol #1	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
Mehta, 3DV'17 [29] ($T = 1$)	57.5	68.6	59.6	67.3	78.1	82.4	56.9	69.1	100.0	117.5	69.4	68.0	55.2	76.5	61.4	72.9
Pavlakos, CVPR17 [38] ($T = 1$)	67.4	71.9	66.7	69.1	72.0	77.0	65.0	68.3	83.7	96.5	71.7	65.8	74.9	59.1	63.2	71.9
Zhou, ICCV'17 [53] ($T = 1$)	54.8	60.7	58.2	71.4	62.0	65.5	53.8	55.6	75.2	111.6	64.1	66.0	51.4	63.2	55.3	64.9
Martinez, ICCV'17[43] ($T = 1$)	51.8	56.2	58.1	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	52.4	62.9
Sun, ICCV17 [43] ($T = 1$)	52.8	54.8	54.2	54.3	61.8	67.2	53.1	53.6	71.7	86.7	61.5	53.4	61.6	47.1	53.4	59.1
Fang, AAAI'18 [12] ($T = 1$)	50.1	54.3	57.0	57.1	66.6	73.3	53.4	55.7	72.8	88.6	60.3	57.7	62.7	47.5	50.6	60.4
Pavlakos, CVPR18 [37] ($T = 1$)	48.5	54.4	54.4	52.0	59.4	65.3	49.9	52.9	65.8	71.1	56.6	52.9	60.9	44.7	47.8	56.2
Hossain, ECCV'18 [17] ($T = 5$)	48.4	50.7	57.2	55.2	63.1	72.6	53.0	51.7	66.1	80.9	59.0	57.3	62.4	46.6	49.6	58.3
Lee, ECCV18 [21] ($T = 3$)	**40.2**	49.2	47.8	52.6	**50.1**	75.0	50.2	**43.0**	**55.8**	73.9	54.1	55.6	58.2	43.3	43.3	52.8
Liu, TPAMI'19 [26] ($T = 1$)	50.7	60.0	51.1	63.6	59.7	69.3	48.8	52.0	72.7	105.3	58.6	61.0	62.2	45.9	48.7	61.1
Pavllo, arxiv'18 [39] ($T = 9$)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	49.8
Pavllo, arxiv'18 [39] ($T = 1$)	47.1	50.6	49.0	51.8	53.6	61.4	49.4	47.4	59.3	67.4	52.4	49.5	55.3	39.5	42.7	51.8
Ours, ($T = 1$)	46.5	48.8	47.6	50.9	52.9	61.3	48.3	45.8	59.2	64.4	51.2	48.4	53.5	39.2	41.2	50.6
Ours, ($T = 3$)	44.9	48.1	46.1	49.4	50.6	**58.4**	47.2	44.4	57.1	62.2	49.7	47.2	52.2	38.2	40.8	49.1
Ours, ($T = 7$)	44.6	**47.4**	**45.6**	**48.8**	50.8	59.0	**47.2**	43.9	57.9	**61.9**	**49.7**	**46.6**	**51.3**	**37.1**	**39.4**	**48.8**
Protocol #2	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
Martinez, ICCV'17 [28] ($T = 1$)	39.5	43.2	46.4	47.0	51.0	56.0	41.4	40.6	56.5	69.4	49.2	45.0	49.5	38.0	43.1	47.7
Sun, ICCV17 [43] ($T = 1$)	42.1	44.3	45.0	45.4	51.5	53.0	43.2	41.3	59.3	73.3	51.0	44.0	48.0	38.3	44.8	48.3
Fang, AAAI'18 [12] ($T = 1$)	38.2	41.7	43.7	44.9	48.5	55.3	40.2	38.2	54.5	64.4	47.2	44.3	47.3	36.7	41.7	45.7
Pavlakos, CVPR18 [37] ($T = 1$)	**34.7**	39.8	41.8	**38.6**	42.5	47.5	38.0	36.6	50.7	56.8	42.6	39.6	43.9	32.1	36.5	41.8
Hossain, ECCV'18 [17] ($T = 5$)	35.7	39.3	44.6	43.0	47.2	54.0	38.3	37.5	51.6	61.3	46.5	41.4	47.3	34.2	39.4	44.1
Lee, ECCV18 [21] ($T = 3$)	34.9	**35.2**	43.2	42.6	46.2	55.0	37.6	38.8	50.9	67.3	48.9	**35.2**	50.7	31.0	34.6	43.4
Pavllo,arxiv'18 [39] ($T = 1$)	36.0	38.7	38.0	41.7	40.1	45.9	37.1	35.4	46.8	53.4	41.4	36.9	43.1	30.3	34.8	40.0
Ours, ($T = 1$)	36.8	38.7	38.2	41.7	40.7	46.8	37.9	35.6	47.6	51.7	41.3	36.8	42.7	31.0	34.7	40.2
Ours, ($T = 3$)	36.0	38.4	37.6	40.8	39.9	45.2	**37.0**	35.0	**46.0**	50.5	40.6	36.5	42.2	30.6	34.5	39.4
Ours, ($T = 7$)	35.7	37.8	**36.9**	40.7	**39.6**	**45.2**	37.4	**34.5**	46.9	**50.1**	**40.5**	36.1	**41.0**	**29.6**	**33.2**	**39.0**

Table 1. Quantitative comparisons of Mean Per Joint Position Error (MPJPE) in millimeter between the estimated pose and the ground-truth on Human3.6M under Protocol #1 and Protocol #2, where T denotes the number of input frames used in each method. The best score is marked in **bold**.

3D pose is aligned to the ground truth via a rigid transformation, which is referred as protocol #2. For the STB dataset, we evaluate the 3D hand pose estimation performance with two metrics. The first metric is the area under the curve (AUC) on the percentage of correct keypoints (PCK) score, which is a popular criterion to evaluate the pose estimation accuracy with different thresholds, as proposed in [3, 54]. The second metric is MPJPE, identical to that for 3D body pose estimation. Following the same condition used in [3, 42, 54], we assume that the global hand scale and the absolute depth of the root joint are provided at test time for 3D hand pose estimation.

4.4. Comparison with the State-of-the-art

Results on Human3.6M. As shown in Table 1, we compare the performance of our approach with previously reported results on Human3.6M, where T represents the number of input frames. For fair comparison, previous methods with different input sequence lengths are listed in this table. Note that [39] reported better results for 3D pose estimation using 243 frames. However, this is not suitable for the online scenarios we focus on, wherein it is not viable to have long sequences of frames as input. From the table, we can see that compared with the state-of-the-art methods with a similar number of input frames, our approach achieves the best performance under all protocols.

Results on STB Dataset. Figure 6 (left) shows the comparison with the state-of-the-art methods [3, 19, 31, 34, 35, 40, 42, 54] on STB for 3D hand pose estimation. It can be seen that our approach outperforms the state-of-the-art methods over most error thresholds, improving the AUC value to 0.995 in the joint error range between 20mm and 50mm. Note that here we measure the 3D PCK curve of our proposed method with a single-frame model for fair comparison, since most of the previous works focus on estimating 3D pose from a single image.

	1-frame	3-frames	5-frames	7-frames
Human3.6M	50.62	49.08	48.86	**48.78**
STB	6.95	6.70	6.65	**6.61**

Table 2. MPJPE Results (in mm) of our method with different input sequence lengths on Human3.6M and STB.

Method	Error (mm)
Uniform GCN	69.8
Split Temporal Connect.	54.8
Split Temporal & Symmetrical Connect.	54.0
Split Temporal & Symmetrical & Physical Connect. (proposed)	**49.1**

Table 3. MPJPE Results (in mm) of our method with 3 input frames and different graph convolutional strategies on Human3.6M.

4.5. Ablation Studies

Impact of input sequence length. Table 2 shows the MPJPE results of our method with different input sequence

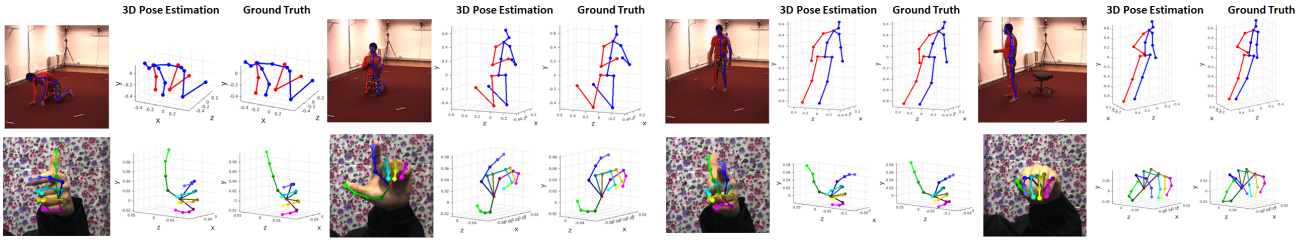


Figure 7. Visual results of our proposed method on Human3.6M and STB datasets. First row: Human3.6M [18]. Second row: STB [52]. Note that skeletons are shown at a novel viewpoint for easy comparison.

lengths on the Human3.6M and STB datasets. We can see that with more input frames used for predictions, our proposed method obtained larger gains in both 3D human and hand pose estimation. This is expected since temporal correlations help resolve issues such as depth ambiguity and self-occlusions, which are typically challenging for single-frame 3D pose estimation task. Noticing that the estimation error with $T = 3$ (49.08 mm) is only slightly higher than those with $T = 5$ (48.86 mm) and $T = 7$ (48.78 mm), we fix $T = 3$ in the following experiments to balance between the estimation accuracy and the computational complexity.

Effect of modified graph convolution. To assess the effectiveness of our modified graph convolution for 3D pose estimation, we carry out experiments on Human3.6M with three variants of our method. a) **Uniform GCN**: all nodes in a neighborhood are uniformly treated with a shared filter matrix. b) **Split Temporal Connect.**: neighboring nodes are divided into three classes: time-forward node, time-backward node and other nodes. c) **Split Temporal & Symmetrical Connect.**: neighboring nodes are divided into four classes: time-forward node, time-backward node, symmetrical node and other nodes. All the models are with 3 input frames and consistent graph topology for fair comparisons. The results are presented in Table 3. It can be seen that the strategy of separating neighboring nodes into three classes (the first variant) with individual kernel weights considerably improves the performance by a large margin (from 69.8 mm to 54.8 mm). Among the multiple ways of partitioning neighboring nodes, our proposed implementation (Split Temporal & Symmetrical & Physical Connect.) achieves the best result (49.1 mm), which indicates the effectiveness of our proposed non-uniform graph convolution that precisely classifies neighboring nodes based on the semantics of the sparse spatial-temporal graph for 3D pose estimation.

Effect of local-to-global prediction. We examine the advantage of using our proposed local-to-global architecture by successively removing the graph pooling and upsampling layers from our model. As presented in Table 4, removing the pooling and upsampling layers leads to 3 mm to 5 mm increase in error, which demonstrates the benefit of leveraging multi-scale features in our proposed framework.

Method	Error (mm)	Δ
Ours, proposed	49.1	-
w/o last pooling & 1st upsampling layers	52.3	3.2
w/o all pooling & upsampling layers	53.9	4.8

Table 4. Ablation studies on different components of our network architecture. The evaluation is performed on Human3.6M with the MPJPE metric under Protocol #1.

Impact of pose refinement. We also evaluate the impact of the proposed pose refinement. As presented in Figure 6 (right), with the pose refinement, the average estimation errors of different body parts as well as the overall mean errors consistently decrease on Human3.6M [18], which indicates that our proposed pose refinement can further improve the estimation accuracy of 3D joint locations.

4.6. Qualitative results.

Figure 7 shows some visual results of our method on Human3.6M [18] and STB [52] datasets. We exhibit samples captured from various viewpoints with serious self-occlusions. The results show that our proposed model can reliably handle the challenging poses with various orientations and complicated pose articulation.

5. Conclusion

In this paper, we have presented a novel graph-based method for 3D pose estimation from a short sequence of extracted 2D joint locations. To incorporate the domain-specific knowledge of the constructed spatial-temporal graph, we have introduced a non-uniform graph convolutional operation by learning individual kernel weights for functionally-variant neighbors. Moreover, a local-to-global network architecture has also been proposed to effectively capture the representative features at different scales. Experimental results on two benchmark datasets have demonstrated the superior performance of our method for both 3D hand pose estimation and 3D human body pose estimation tasks.

References

[1] James Atwood and Don Towsley. Diffusion-convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1993–2001, 2016.

[2] Xavier Bresson and Thomas Laurent. Residual gated graph convnets. *arXiv preprint arXiv:1711.07553*, 2017.

[3] Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 666–682, 2018.

[4] Xianjie Chen and Alan L Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *Advances in neural information processing systems*, pages 1736–1744, 2014.

[5] Yujin Chen, Zhigang Tu, Liuhao Ge, Dejun Zhang, Ruizhi Chen, and Junsong Yuan. So-handnet: Self-organizing network for 3d hand pose estimation with semi-supervised learning. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[6] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018.

[7] Fan RK Chung and Fan Chung Graham. *Spectral graph theory*. Number 92. American Mathematical Soc., 1997.

[8] Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. Learning 3d human pose from structure and motion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 668–683, 2018.

[9] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, pages 3844–3852, 2016.

[10] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems*, pages 2224–2232, 2015.

[11] Ali Erol, George Bebis, Mircea Nicolescu, Richard D Boyle, and Xander Twombly. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108(1-2):52–73, 2007.

[12] Hao-Shu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. Learning pose grammar to encode human body configuration for 3d pose estimation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

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

[14] Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3593–3601, 2016.

[15] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10833–10842, 2019.

[16] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11807–11816, 2019.

[17] Mir Rayat Imtiaz Hossain and James J Little. Exploiting temporal information for 3d human pose estimation. In *European Conference on Computer Vision*, pages 69–86. Springer, 2018.

[18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian 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):1325–1339, 2014.

[19] Umar Iqbal, Pavlo Molchanov, Thomas Breuel Juergen Gall, and Jan Kautz. Hand pose estimation via latent 2.5 d heatmap regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 118–134, 2018.

[20] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[21] Kyoungoh Lee, Inwoong Lee, and Sanghoon Lee. Propagating lstm: 3d pose estimation based on joint interdependency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 119–135, 2018.

[22] Ron Levie, Federico Monti, Xavier Bresson, and Michael M Bronstein. Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Transactions on Signal Processing*, 67(1):97–109, 2017.

[23] Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang. Adaptive graph convolutional neural networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[24] Sijin Li and Antoni B Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *Asian Conference on Computer Vision*, pages 332–347. Springer, 2014.

[25] Hui Liang, Junsong Yuan, and Daniel Thalman. Egocentric hand pose estimation and distance recovery in a single rgb image. In *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2015.

[26] Jun Liu, Henghui Ding, Amir Shahroudy, Ling-Yu Duan, Xudong Jiang, Gang Wang, and Alex Kot Chichung. Feature boosting network for 3d pose estimation. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[27] Diogo C Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5137–5146, 2018.

[28] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2640–2649, 2017.

[29] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 International Conference on 3D Vision (3DV)*, pages 506–516. IEEE, 2017.

[30] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):44, 2017.

[31] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[32] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016.

[33] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In *International conference on machine learning*, pages 2014–2023, 2016.

[34] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BmVC*, volume 1, page 3, 2011.

[35] Paschalis Panteleris, Iason Oikonomidis, and Antonis Argyros. Using a single rgb frame for real time 3d hand pose estimation in the wild. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 436–445. IEEE, 2018.

[36] Sungheon Park, Jihye Hwang, and Nojun Kwak. 3d human pose estimation using convolutional neural networks with 2d pose information. In *European Conference on Computer Vision*, pages 156–169. Springer, 2016.

[37] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3d human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7307–7316, 2018.

[38] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7025–7034, 2017.

[39] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. *arXiv preprint arXiv:1811.11742*, 2018.

[40] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. Realtime and robust hand tracking from depth. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1106–1113, 2014.

[41] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *arXiv preprint arXiv:1211.0053*, 2012.

[42] Adrian Spurr, Jie Song, Seonwook Park, and Otmar Hilliges. Cross-modal deep variational hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 89–98, 2018.

[43] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional human pose regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2602–2611, 2017.

[44] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Structured prediction of 3d human pose with deep neural networks.

[45] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018.

[46] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[47] Ying Wu and Thomas S Huang. Capturing articulated human hand motion: A divide-and-conquer approach. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 606–611. IEEE, 1999.

[48] Ying Wu and Thomas S Huang. View-independent recognition of hand postures. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 88–94. IEEE, 2000.

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

[50] 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.

[51] Wei Yang, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3073–3082, 2016.

[52] Jiawei Zhang, Jianbo Jiao, Mingliang Chen, Liangqiong Qu, Xiaobin Xu, and Qingxiong Yang. A hand pose tracking benchmark from stereo matching. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 982–986. IEEE, 2017.

[53] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Towards 3d human pose estimation in the wild: a weakly-supervised approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 398–407, 2017.

[54] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4903–4911, 2017.