Adaptive Graph Kolmogorov-Arnold Network for 3D Human Pose Estimation

Abu Taib Mohammed Shahjahan and A. Ben Hamza
Concordia University, Montreal, Canada
abutaibmohammed.shahjahan@mail.concordia.ca

Abstract

Graph convolutional network (GCN)-based methods have shown strong performance in 3D human pose estimation by leveraging the natural graph structure of the human skeleton. However, their local receptive field limits their ability to capture long-range dependencies essential for handling occlusions and depth ambiguities. They also exhibit spectral bias, which prioritizes low-frequency components while struggling to model high-frequency details. In this paper, we introduce PoseKAN, an adaptive graph Kolmogorov-Arnold Network (KAN), framework that extends KANs to graph-based learning for 2D-to-3D pose lifting from a single image. Unlike GCNs that use fixed activation functions, KANs employ learnable functions on graph edges, allowing data-driven, adaptive feature transformations. This enhances the model's adaptability and expressiveness, making it more expressive in learning complex pose variations. Our model employs multi-hop feature aggregation, ensuring the body joints can leverage information from both local and distant neighbors, leading to improved spatial awareness. It also incorporates residual PoseKAN blocks for deeper feature refinement, and a global response normalization for improved feature selectivity and contrast. Extensive experiments on benchmark datasets demonstrate the competitive performance of our model against state-of-the-art methods. Code is available at: https://github.com/shahjahan0275/PoseKAN

1. Introduction

3D human pose estimation aims to infer the 3D coordinates of body joints from 2D images or videos, a task that provides critical insights into human movement but remains challenging due to inherent depth ambiguities and occlusions in the input data. Mainstream approaches to 3D human pose estimation can be broadly categorized into one- and two-stage methods. The former aim to predict 3D joint coordinates directly from input images or video frames, bypassing any intermediate representation. However, they often struggle with occlusions, leading to inaccurate pose predictions [18, 19, 26, 27]. In contrast, two-stage methods decompose the problem into two separate stages: (1) detecting 2D joint locations in the image and (2) lifting these 2D detections into 3D space. The modular nature of two-stage pipelines helps optimize each component, selecting the best-performing 2D pose detectors and combining them with robust 2D-to-3D lifting models, ultimately leading to better 3D pose estimations and greater flexibility [6, 14, 16, 32]. Recently, GCN-based methods have emerged as a powerful approach to 3D human pose estimation, effectively leveraging the inherent skeletal structure of the human body [13, 37, 38, 41, 43]. By formulating pose estimation as a graph-based learning problem, GCN-based models can effectively capture spatial dependencies between body parts. However, they still encounter several limitations that hinder their effectiveness in challenging 3D pose estimation scenarios. One of the primary limitations is their inability to model global context effectively due to their reliance on aggregating information mostly from immediate neighbors, thereby struggling to capture long-range dependencies between distant joints. They also incorporate Multi-Layer Perceptrons (MLPs) as their core components for feature transformation, inheriting a fundamental drawback of MLPs, namely spectral bias [22]. As a result, these models tend to prioritize low-frequency components while struggling to capture high-frequency details, which are crucial for accurately modeling rapid movements and fine-grained joint interactions. This limitation can significantly impact pose estimation accuracy, particularly in highly articulated actions such as sports activities, where fine-grained motion details are essential for accurate predictions. Furthermore, GCN-based approaches lack interpretability due to their reliance on predefined activation functions. More recently, Bresson *et al.* [3] introduce KAGNNs, a family of graph neural networks that replace MLPs in standard message-passing layers with KANs for node feature updates, evaluating them on node and graph classification, link prediction, and graph regression.

Proposed Work and Contributions. Addressing the aforementioned challenges of GCN-based methods for 3D human pose estimation requires a more adaptive and ex-

pressive approach that can dynamically adapt feature transformations, model long-range dependencies, and mitigate spectral bias. To this end, we propose an adaptive graph Kolmogorov-Arnold Network (PoseKAN), a novel graph-based framework that incorporates learnable function-based transformations. Drawing inspiration from the Kolmogorov-Arnold representation theorem [2, 23], KANs [15] have recently been introduced as a compelling alternative to MLPs, offering significant improvements in both interpretability and adaptability for function approximation tasks while mitigating spectral bias [30]. The main contributions of our work can be summarized as follows:

- We present PoseKAN, a novel adaptive graph Kolmogorov-Arnold network framework for 3D human pose estimation. It employs *2D-to-3D lifting from a single image* by levearing learnable function-based transformations, improving adaptability and generalization while maintaining computational efficiency.
- We design a multi-hop feature propagation scheme that extends beyond one-hop aggregation by introducing a scaling parameter that controls the balance between local and global feature aggregation, improving robustness to occlusions and depth ambiguities.
- We rigorously validate the effectiveness of our model through comprehensive experimental evaluation, including a comparative analysis and ablation studies against state-of-the-art methods on two benchmark datasets, demonstrating improved performance.

2. Related Work

Since our PoseKAN model follows the 2D-to-3D lifting pipeline [6, 14, 16, 32], we primarily focus in this section on relevant GCN-based methods within this category while also discussing how KANs fit into the broader landscape of graph-based learning.

3D Human Pose Estimation. The aim of 3D human pose estimation is to predict the 3D coordinates of body joints from 2D images or video frames. The 2D-to-3D lifting pipeline (i.e., two-stage) first detects 2D joint positions and then maps them to their 3D coordinates. Compared to one-stage methods, which directly regress 3D joint locations from images [19, 27] without an intermediate representation, two-stage approaches have demonstrated greater robustness, particularly in handling occlusions and depth ambiguities [6, 14, 16, 32]. Despite their success, most existing 2D-to-3D lifting methods utilize fully connected networks, which inherently lack spatial awareness and are prone to overfitting due to their densely connected architecture. Our proposed approach falls under the category of two-stage methods for *2D-to-3D lifting from a single image*, focusing on learning flexible graph representations for more accurate 2D-to-3D pose lifting.

Graph-based 3D Human Pose Estimation. Human body joints naturally form a structured graph, where nodes represent joints and edges capture relationships between them. Graph-based learning approaches, particularly GCNs [8], have been widely applied to exploit this structured representation, showing promising results by leveraging the skeletal graph structure to model body joint dependencies [38]. However, most GCN-based methods for 3D human pose estimation primarily aggregate information from one-hop neighbors, limiting their ability to model long-range dependencies. To address this, high-order GCN-based models [24, 42] extended GCNs to multi-hop neighborhoods, capturing long-range dependencies beyond immediate joint connections. Modulated GCN [41] introduced adjacency matrix modulation, enabling the model to learn additional edges beyond the predefined skeletal structure. GroupGCN [37] and Multi-hop Modulated GCN [9] further refined graph feature propagation by fusing multi-hop neighborhood information. More recently, GraphMLP [12] combined MLPs and GCNs to leverage both global and local skeletal interactions. While GCN-based approaches have led to notable improvements in 3D human pose estimation, they still face three key limitations. First, standard GCN architectures rely on one-hop neighborhood aggregation. This restricts the ability to capture long-range dependencies, which are crucial for handling occlusions and depth ambiguities. Second, like MLPs, GCNs exhibit spectral bias, meaning they favor low-frequency components while struggling to capture high-frequency details, thereby hindering their ability to accurately represent complex poses and rapid movement changes. Third, GCN-based methods apply predefined activation functions at the node level, restricting adaptability. By comparison, our proposed PoseKAN framework learns activation functions dynamically on graph edges, providing greater flexibility in feature transformation and improving interpretability, while mitigating the spectral bias thanks, in large part, to learning function-based transformations that help capture both low- and high-frequency components. Low-frequency components represent smooth, gradual variations in joint positions over time or space, whereas high-frequency components correspond to rapid, abrupt changes in motion. In the Human3.6M dataset [7], for instance, actions such as Walking and Eating predominantly exhibit low-frequency characteristics due to their smooth and predictable patterns. In contrast, high-frequency components are more prominent in actions like Greeting, where sudden movements and finer pose adjustments are required.

3. Method

In this section, we first describe the 3D human pose estimation task, followed by a preliminary background on KANs [15, 30]. Subsequently, we introduce a novel spec-

tral modulation filter, which serves as the cornerstone for our proposed adaptive graph KAN model.

Problem Description. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ be an attributed graph representing the skeletal structure of the human body, where $\mathcal{V} = \{1, \ldots, J\}$ is the set of J nodes (i.e., body joints) and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges, and $\mathbf{X}$ a $J \times F$ feature matrix of node attributes. We denote by $\mathbf{A}$ a $J \times J$ adjacency matrix whose (i, j)-th entry is equal to 1 if i and j are neighboring nodes, and 0 otherwise. We also denote by $\hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$ the normalized adjacency matrix, where $\mathbf{D} = \mathsf{diag}(\mathbf{A}\mathbf{1})$ is the diagonal degree matrix and $\mathbf{1}$ is a vector of all ones. Given a training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ comprised of 2D joint positions $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^2$ and their associated ground-truth 3D joint positions $\mathbf{y}_i \in \mathcal{Y} \subset \mathbb{R}^3$, the aim of 3D human pose estimation is to learn the parameters $\mathbf{w}$ of a regression model $f_{\mathbf{w}} : \mathcal{X} \to \mathcal{Y}$ by finding a minimizer of a loss function.

Kolmogorov-Arnold Networks. KANs are inspired by the Kolmogorov-Arnold representation theorem [2, 23], which states that any continuous multivariate function on a bounded domain can be represented as a finite composition of continuous univariate functions of the input variables and the binary operation of addition. A KAN layer is a fundamental building block of KANs [15], and is given by a matrix $\mathbf{\Phi} = (\phi_{q,p})$ of 1D functions, where each trainable activation function ϕ is defined as a weighted combination, with learnable weights, of a sigmoid linear unit (SiLU) function and a spline function:

$$\phi(x) = w_b b(x) + w_s \text{spline}(x), \tag{1}$$

where $b(x) = \text{SiLU}(x) = x/(1 + e^{-x})$ and $\text{spline}(x) = \sum_i c_i B_i(x)$ is a weighted sum of B-splines basis functions with trainable coefficients c_i. During training, the weights w_b and w_s are learned to optimize performance. Given a input feature vector $\mathbf{x}^{(\ell)} \in \mathbb{R}^{F_\ell}$, the output of the ℓ-th KAN layer is an $F_{\ell+1}$-dimensional feature vector given by

$$\mathbf{x}^{(\ell+1)} = \text{KAN}^{(\ell)}(\mathbf{x}^{(\ell)}). \tag{2}$$

3.1. Spectral Modulation Filtering

Spectral graph filtering employs filters defined as functions of the graph normalized Laplacian (or its eigenvalues). The goal of these filters, often referred to as frequency responses or transfer functions, is to reduce or eliminate high-frequency noise in the graph signal. These functions basically describe how a filter affects the input graph signal to produce the output graph signal. We define a spectral modulation filter as follows:

$$h_s(\lambda) = \frac{1}{(1+s)\lambda - s\lambda^2}, \tag{3}$$

where $s \in (0, 1)$ is a positive scaling parameter that allows for the modulation or adjustment of the spectral characteristics, indicating its capability to control the filtering effect on different frequency components of the graph signal. The filter h_s is a rational polynomial function of the eigenvalues of the normalized Laplacian matrix. It is an adaptive-pass filter in the sense that it exhibits low-pass characteristics, as it allows low-frequency components (corresponding to small eigenvalues) to adaptively pass through with little attenuation, while attenuating high-frequency components (associated with large eigenvalues). The attenuation behavior of the filter is determined by the scalar s, which serves as a tuning or modulation parameter. By adjusting s, we can control the trade-off between preserving low-frequency structural information and reducing high-frequency noise or variations in the graph signal. When s is large, the filter is less selective, allowing a wider range of frequencies to pass through with less attenuation. As s decreases, the filter becomes more selective and significantly attenuates higher frequencies, effectively filtering out more of the high-frequency noise or variations in the graph signal.

Graph Filtering System. Applying the spectral modulation filter on the graph signal (i.e., feature matrix) $\mathbf{X} \in \mathbb{R}^{N \times F}$ yields a filtered graph signal $\mathbf{H} \in \mathbb{R}^{N \times F}$ given by

$$\mathbf{H} = h_s(\mathbf{L})\mathbf{X} = ((1+s)\mathbf{L} - s\mathbf{L}^2)^{-1}\mathbf{X}, \tag{4}$$

which can be rewritten as

$$((1+s)\mathbf{L} - s\mathbf{L}^2)\mathbf{H} = \mathbf{X}, \tag{5}$$

or equivalently

$$\begin{aligned} \mathbf{H} &= (\mathbf{I} - (1+s)\mathbf{L} + s\mathbf{L}^2)\mathbf{H} + \mathbf{X} \\ &= (\mathbf{I} - s\mathbf{L})(\mathbf{I} - \mathbf{L})\mathbf{H} + \mathbf{X}. \end{aligned} \tag{6}$$

Since $\mathbf{L} = \mathbf{I} - \hat{\mathbf{A}}$, where $\hat{\mathbf{A}}$ is the normalized adjacency matrix, the spectral modulation filter equation becomes

$$\mathbf{H} = ((1-s)\mathbf{I} + s\hat{\mathbf{A}})\hat{\mathbf{A}}\mathbf{H} + \mathbf{X}, \tag{7}$$

which can be solved using, for instance, the fixed point iteration method [4], where $\mathbf{H} = \varphi(\mathbf{H})$ with the function φ defined by the right-hand side term of Eq. (7).

Fixed Point Iterative Solution. Fixed-point iteration is an iterative numerical method used to find a fixed point of a given function [4]. The process involves repeatedly applying the function to an initial guess or estimate and updating this estimate in each iteration until it converges to the fixed point. For the spectral modulation filter equation $\mathbf{H} = \varphi(\mathbf{H})$, the fixed point iterative solution is given by

$$\mathbf{H}^{(t+1)} = ((1-s)\mathbf{I} + s\hat{\mathbf{A}})\hat{\mathbf{A}}\mathbf{H}^{(t)} + \mathbf{X}, \tag{8}$$

with some initial guess $\mathbf{H}^{(0)}$, and $t \in \mathbb{N}$ denotes the iteration number.

3.2. Proposed Graph KAN Model

While GCN-based methods have shown strong performance in 3D human pose estimation, they, however, face several limitations that restrict their effectiveness. First, GCNs rely on fixed activation functions and trainable weights, restricting their ability to dynamically adapt to variations in human poses. Second, GCNs primarily aggregate information from one-hop neighbors, limiting their receptive field and making it challenging to capture long-range dependencies that are crucial for handling occlusions and depth ambiguities in 3D pose estimation. Third, GCNs exhibit spectral bias due to their use of MLPs for feature transformation.

Adaptive Graph KAN. Inspired by our fixed point iterative solution given by Eq. (8) and to overcome the aforementioned limitations of GCN-based methods, we introduce an adaptive graph KAN (PoseKAN) model, which employs learnable function-based transformations in lieu of fixed activation functions and trainable weight matrices. Specifically, we define the layer-wise update rule of PoseKAN for node feature propagation as follows:

$$\mathbf{H}^{(\ell+1)} = \text{KAN}^{(\ell)}\Big(((1-s)\hat{\mathbf{A}} + s\hat{\mathbf{A}}^2)\mathbf{H}^{(\ell)} + \mathbf{X}\Big), \quad (9)$$

where $\ell \in \{0, \ldots, L-1\}$, L is the total number of layers, $\text{KAN}^{(\ell)}$ is a learnable single KAN layer, $s \in (0, 1)$ is a positive scaling parameter that adjusts the contribution of immediate and second-order neighbors, and $\mathbf{H}^{(\ell)} \in \mathbb{R}^{J \times F_\ell}$ is the input feature matrix at the ℓ-th layer with embedding dimension F_ℓ. The input of the first layer is $\mathbf{H}^{(0)} = \mathbf{X}$. Note that unlike GCN, which uses fixed activation functions, PoseKAN learns its own activation functions dynamically, making it more expressive.

The PoseKAN update rule can be decomposed into two main operations: feature propagation and feature embedding. Feature propagation is given by

$$\mathbf{G}^{(\ell)} = \mathbf{P}\mathbf{H}^{(\ell)} + \mathbf{X}, \quad (10)$$

where $\mathbf{P} = (1-s)\hat{\mathbf{A}} + s\hat{\mathbf{A}}^2$ is the propagation matrix defined as a weighted combination of the normalized adjacency matrix and its square, ensuring that multi-hop dependencies are captured. The addition of a residual connection ensures that information from the initial feature matrix is preserved throughout the layers. The parameter s plays a crucial role in controlling the balance between local and higher-order information. After propagation, the feature embedding step refines the node representations by applying a KAN layer as follows:

$$\mathbf{H}^{(\ell+1)} = \text{KAN}^{(\ell)}(\mathbf{G}^{(\ell)}), \quad (11)$$

where each edge in the graph is associated with a learnable univariate function via $\text{KAN}^{(\ell)}$, providing greater flexibility in feature adaptation.

While both PoseKAN and its GCN counterpart use learnable transformations, GCN employs learnable weight matrices for feature transformations, but these transformations are still linear mappings applied at each node. PoseKAN, on the other hand, employs learnable activation functions on edges rather than nodes. This not only allows for greater flexibility and adaptability, but also leads to improved expressiveness and mitigates spectral bias.

Model Architecture. The overall framework of our PoseKAN model architecture is depicted in Figure 1. The input to the model consists of 2D keypoints, obtained via an off-the-shelf 2D pose detector [5]. Unlike GCN-based methods that use fixed activation functions, PoseKAN employs learnable function-based transformations, enabling adaptive, expressive, and interpretable feature learning. The PoseKAN architecture consists of start and end PoseKAN layers, and four residual PoseKAN blocks. The PoseKAN layer is the core computational unit of the model. The start PoseKAN layer maps input pose representations into latent space for effective feature learning, while the end PoseKAN layer projects refined feature representations back to pose space for final 3D pose predictions. Each residual PoseKAN block is comprised of five PoseKAN layers that learn hierarchical pose features, a layer normalization for stabilized training, an additional PoseKAN layer followed by the Gaussian Error Linear Unit (GELU) nonlinearity, and a residual connection to retain the original information and prevent gradient vanishing. GELU effectively preserves input magnitudes by smoothly blending linear and nonlinear transformations, thereby enhancing adaptability and expressiveness. The end PoseKAN layer generates the 3D pose, and is preceded by global response normalization (GRN) [31] to ensure feature contrast before prediction.

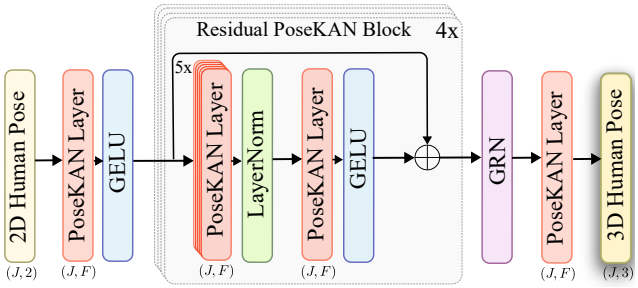

Figure 1. **Overview of Model Architecture**. The model takes 2D pose coordinates as input and produces 3D pose predictions as output, where J is the number of joints and F is the embedding dimension. The architecture consists of a start PoseKAN layer, four residual FG-GCN blocks, and an end PoseKAN layer. A global response normalization is also used.

Model Prediction. The end PoseKAN layer serves as the output transformation stage, where the learned feature representations are projected back into the 3D pose space to

generate the final predicted joint coordinates. The output of this layer consists of the final node embeddings, $\hat{\mathbf{y}}_i$, which is the predicted 3D pose coordinates of i-th joint. Global response normalization is applied before the output projection to ensure that the output feature magnitudes are well-calibrated, reducing noise, and improving generalization.

Model Training. The parameters (i.e., univariate activation functions) of the PoseKAN model are learned by minimizing the following loss function:

$$\mathcal{L} = \frac{1}{N}\Big[(1-\alpha)\sum_{i=1}^{N}\|\mathbf{y}_i-\hat{\mathbf{y}}_i\|_2^2+\alpha\sum_{i=1}^{N}\|\mathbf{y}_i-\hat{\mathbf{y}}_i\|_1\Big], \quad (12)$$

which is a weighted combination of the mean squared error (MSE) and mean absolute error (MAE) between the true 3D joint coordinates $\mathbf{y}_i$ and the predicted 3D joint coordinates $\hat{\mathbf{y}}_i$ across N training joints. The weight $\alpha \in [0, 1]$ controls the contribution of each error term. Our proposed loss function $\mathcal{L}$ draws inspiration from the penalty function used in the elastic net regression model [40], which is a weighted combination of lasso and ridge regularization.

Model Complexity Analysis. For simplicity, we assume the embedding dimensions are the same across all layers, i.e., $F_\ell = F$ for all ℓ. The computational cost of multiplying the propagation matrix $\mathbf{P}$ with the embedding $\mathbf{H}^{(\ell)}$ is $\mathcal{O}(\|\hat{\mathbf{A}}\|_0 F)$ in time, where $\|\hat{\mathbf{A}}\|_0$ represents the number of nonzero entries in the sparse normalized adjacency matrix $\hat{\mathbf{A}}$, effectively corresponding to the number of edges in the graph. This term quantifies the complexity of message passing in the graph structure. Applying a KAN layer incurs a computational cost of $\mathcal{O}(GF^2)$, where G is the grid size. Consequently, the overall *time complexity* of an L-layer PoseKAN is $\mathcal{O}(L\|\hat{\mathbf{A}}\|_0 F + LGF^2)$, where the first term corresponds to feature propagation, while the second term arises from KAN-based feature transformations. Regarding *memory complexity*, an L-layer PoseKAN requires $\mathcal{O}(LJF+2^kGLF^2)$ in memory, where $\mathcal{O}(LJF)$ accounts for storing feature embeddings across all layers and $\mathcal{O}(2^kGLF^2)$ represents memory usage in L KAN layers, with 2^k stemming from the recursive computation of order k splines used in function approximation. By comparison, a standard GCN has a relatively lower memory complexity of $\mathcal{O}(LJF + LF^2)$, as it lacks the function-based transformations introduced by KAN layers. However, since k and G are typically small in practical implementations, the additional cost remains manageable. A key computational optimization in PoseKAN is that it avoids explicitly computing the square of the normalized adjacency matrix. Instead, we apply a right-to-left multiplication strategy, where $\hat{\mathbf{A}}$ is first multiplied with the feature embedding before the second multiplication is performed. This optimization eliminates unnecessary matrix exponentiation, significantly reducing computational overhead while preserving the benefits of multi-hop feature propagation. As a result, PoseKAN achieves a strong balance between computational efficiency and enhanced model expressiveness.

4. Experiments

4.1. Experimental Setup

Datasets. We conduct experimental evaluations on two standard datasets: Human3.6M [7] and MPI-INF-3DHP [17]. We follow standard protocols [16, 38] for data preprocessing and splitting on these benchmark datasets.

Evaluation Protocols and Metrics. We employ two standard evaluation protocols for training and testing on Human 3.6M designated as Protocol #1 and Protocol #2 [16], with associated metrics mean per-joint position error (MPJPE) and Procrustes-aligned mean per-joint position error (PA-MPJPE), respectively. For MPI-INF-3DHP, we use the Area Under Curve (AUC) and Percentage of Correct Keypoint (PCK) as assessment metrics.

Baseline Methods. We compare the performance of our model with several state-of-the-art methods for 3D pose estimation, including semantic GCN (SemGCN) [38], High-order GCN [42], Compositional GCN (CompGCN) [43], Higher-Order Implicit Fairing Network (HOIF-Net) [21], Multi-hop Modulated GCN (MM-GCN) [9], Group GCN [37], Modulated GCN [41], Flexible GCN (Flex-GCN) [24], and GraphMLP [12].

Implementation Details. Our model is implemented in PyTorch and all experiments are conducted on a single NVIDIA RTX A4500 GPU with 20GB of memory. We employ the AMSGrad optimizer for training, running for 30 epochs on both 2D ground truth and 2D pose detections [5]. The initial learning rate is set to 0.001, with a decay factor of 0.99 every four epochs. The batch size and embedding dimension are set to 64 and $F = 240$, respectively. The scaling parameter $s = 0.2$ and weighting factor $\alpha = 0.03$ are determined via grid search. To prevent overfitting, we apply dropout with a factor of 0.2 after each PoseKAN layer. The spline order and grid size are set to 3 and 5, respectively.

4.2. Results and Analysis

Quantitative Results on Human3.6M. In Tables 1 and 2, we report performance comparison of our PoseKAN model and strong baselines for 3D pose estimation on Human3.6M using the detected 2D pose as input. In both tables, we present the results of all 15 actions, as well as the average performance. Table 1 demonstrates that under Protocol #1, PoseKAN consistently outperforms state-of-the-art models across multiple action categories. Compared to GraphMLP, the strongest competing baseline, PoseKAN achieves a relative error reduction of 2.7%. Against Modulated GCN,

Table 1. Comparison of our model and baseline methods in terms of Mean Per Joint Position Error (MPJPE) in millimeters, computed between the ground truth and estimated poses on the Human3.6M dataset under Protocol #1. The last column displays the average errors. The best results are shown in **bold**, and the second best results are underlined.

	Action															
Method	Dire.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
Ordinal Depth [20]	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
MultiPoseNet [25]	48.6	54.5	54.2	55.7	62.2	72.0	50.5	54.3	70.0	78.3	58.1	55.4	61.4	45.2	49.7	58.0
SemGCN [38]	47.3	60.7	51.4	60.5	61.1	**49.9**	47.3	68.1	86.2	**55.0**	67.8	61.0	**42.1**	60.6	45.3	57.6
WSGN [11]	62.0	69.7	64.3	73.6	75.1	84.8	68.7	75.0	81.2	104.3	70.2	72.0	75.0	67.0	69.0	73.9
PoseGraphNet [1]	51.0	55.3	54.0	54.6	62.4	76.0	51.6	52.7	79.3	87.1	58.4	56.0	61.8	48.1	44.1	59.5
PGDA [34]	47.1	52.8	54.2	54.9	63.8	72.5	51.7	54.3	70.9	85.0	58.7	54.9	59.7	43.8	47.1	58.1
High-order GCN [42]	49.0	54.5	52.3	53.6	59.2	71.6	49.6	49.8	66.0	75.5	55.1	53.8	58.5	40.9	45.4	55.6
HOIF-Net [21]	47.0	53.7	50.9	52.4	57.8	71.3	50.2	49.1	63.5	76.3	54.1	51.6	56.5	41.7	45.3	54.8
CompGCN [43]	48.4	53.6	49.6	53.6	57.3	70.6	51.8	50.7	62.8	74.1	54.1	52.6	58.2	41.5	45.0	54.9
Modulated GCN [41]	45.4	49.2	45.7	49.4	50.4	58.2	47.9	46.0	57.5	63.0	49.7	46.6	52.2	38.9	40.8	49.4
MM-GCN [9]	46.8	51.4	46.7	51.4	52.5	59.7	50.4	48.1	58.0	67.7	51.5	48.6	54.9	40.5	42.2	51.7
Group GCN [37]	45.0	50.9	49.0	49.8	52.2	60.9	49.1	46.8	61.2	70.2	51.8	48.6	54.6	39.6	41.2	51.6
GraphMLP [12]	43.7	49.3	45.5	47.9	50.5	56.0	46.3	44.1	55.9	59.0	48.4	45.7	51.2	**37.1**	39.1	48.0
PoseKAN (ours)	**40.9**	**45.6**	**44.4**	**47.4**	**48.4**	52.8	**44.1**	**41.6**	**54.6**	63.8	**46.1**	**45.1**	49.0	38.1	**39.0**	**46.7**

Table 2. Comparison of our model and baseline methods in terms of Procrustes-aligned Mean Per Joint Position Error (PA-MPJPE), computed between the ground truth and estimated poses on the Human3.6M dataset under Protocol #2.

	Action															
Method	Dire.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
p-LSTMs [10]	38.0	39.3	46.3	44.4	49.0	55.1	40.2	41.1	53.2	68.9	51.0	39.1	**33.9**	56.4	38.5	46.2
WSGN [11]	38.5	41.7	39.6	45.2	45.8	46.5	37.8	42.7	52.4	62.9	45.3	40.9	45.3	38.6	38.4	44.3
PoseGraphNet [1]	38.4	43.1	42.9	44.0	47.8	56.0	39.3	39.8	61.8	67.1	46.1	43.4	48.4	40.7	35.1	46.4
PGDA [34]	36.7	39.5	41.5	42.6	46.9	53.5	38.2	36.5	52.1	61.5	45.0	42.7	45.2	35.3	40.2	43.8
High-order GCN [42]	38.6	42.8	41.8	43.4	44.6	52.9	37.5	38.6	53.3	60.0	44.4	40.9	46.9	32.2	37.9	43.7
HOIF-Net [21]	36.9	42.1	40.3	42.1	43.7	52.7	37.9	37.7	51.5	60.3	43.9	39.4	45.4	31.9	37.8	42.9
CompGCN [43]	38.4	41.1	40.6	42.8	43.5	51.6	39.5	37.6	49.7	58.1	43.2	39.2	45.2	32.8	38.1	42.8
Modulated GCN [41]	35.7	38.6	36.3	40.5	39.2	44.5	37.0	35.4	46.4	51.2	40.5	35.6	41.7	30.7	33.9	39.1
MM-GCN [9]	35.7	39.6	37.3	41.4	40.0	44.9	37.6	36.1	46.5	54.1	40.9	36.4	42.8	31.7	34.7	40.3
Group GCN [37]	35.3	39.3	38.4	40.8	41.4	45.7	36.9	35.1	48.9	55.2	41.2	36.3	42.6	30.9	33.7	40.1
GraphMLP [12]	35.1	38.2	36.5	**39.8**	39.8	43.5	35.7	34.0	45.6	**47.6**	39.8	**35.1**	41.1	**30.0**	33.4	38.4
PoseKAN (ours)	**34.0**	**37.2**	**36.0**	40.0	**39.0**	**42.9**	**35.1**	**33.3**	**45.4**	54.5	**38.3**	36.0	**40.5**	30.1	**32.4**	**38.3**

our model achieves a 5.47% reduction in error and performs better in 14 out of 15 action categories, emphasizing the superiority of learnable function-based transformations over pre-defined activation functions in traditional GCNs. Furthermore, PoseKAN significantly outperforms High-Order GCN, achieving a relative error reduction of 15.99%, showcasing its ability to capture multi-hop dependencies more effectively without relying on fixed high-order convolutions. Our model not only excels in standard poses but also demonstrates superior generalization on challenging actions involving significant self-occlusions, such as Eating, Sitting, and Smoking. PoseKAN effectively mitigates these challenges through multi-hop information aggregation, enabling the model to recover occluded joint positions more accurately than prior GCN-based methods.

Under Protocol #2, Table 2 demonstrates that PoseKAN achieves the lowest PA-MPJPE of 38.3mm, outperforming all baseline models. When compared to GraphMLP, PoseKAN achieves a relative error reduction of 0.26%, indicating a modest yet consistent improvement over the best competing baseline. Moreover, PoseKAN outperforms GraphMLP in 11 out of 15 action categories, reinforcing its ability to generalize across a diverse range of human activities. Similarly, PoseKAN reduces the error by 2.04% compared to Modulated GCN and achieves better results in 13 out of 15 actions. Notably, PoseKAN demonstrates su-

perior performance in occlusion-prone actions, outperforming Modulated GCN by 1.23% in Greeting, 2.15% in Sitting, and 5.43% in Smoking, highlighting its robustness in handling self-occlusions and complex body poses. Furthermore, PoseKAN surpasses Modulated GCN on the challenging Photo action, achieving a relative error reduction of 3.59%, indicating better feature learning in scenarios with complex backgrounds and fine-grained pose details. Most significantly, PoseKAN outperforms High-Order GCN with a substantial relative error reduction of 12.35%, achieving lower errors across all action categories.

Qualitative Results. Figure 2 presents qualitative visualizations of PoseKAN's predictions on the Human3.6M dataset across various action categories, illustrating the model's effectiveness in accurately estimating 3D human poses from monocular 2D inputs. The predicted 3D poses exhibit a high degree of alignment with the ground truth, demonstrating the model's ability to capture spatial dependencies and maintain structural consistency even in complex motion scenarios. The precise joint placements and natural articulation of limbs in our predictions further validate PoseKAN's ability to mitigate depth ambiguities and occlusions, which are inherent challenges in 2D-to-3D pose estimation. A comparative analysis with GraphMLP further highlights PoseKAN's advantages in handling challenging cases involving self-occlusion, where certain body parts obscure others, making pose reconstruction significantly harder. Unlike GraphMLP, which occasionally produces unnatural joint positions or misaligned skeletal structures, PoseKAN generates pose estimations that closely resemble the ground truth pose, capturing fine-grained joint interactions more effectively.

Table 3. Results on MPI-INF-3DHP.

Method	PCK (↑)	AUC (↑)
Ordinal Depth [20]	71.9	35.3
Pose Attribute [28]	71.9	35.8
HOIF-Net [21]	72.8	36.5
LCN [6]	74.0	36.7
HEMlets [39]	75.3	38.0
SRNet [35]	77.6	43.8
Weight Unsharing [13]	79.3	47.6
CompGCN [43]	79.3	45.9
GraphSH [33]	80.1	45.8
HCSF [36]	82.1	46.2
MM-GCN [9]	81.6	50.3
Group GCN [37]	81.1	49.9
Flex-GCN [24]	85.2	51.8
ICFNet [29]	85.6	**54.3**
PoseKAN (ours)	**86.0**	52.9

Cross-Dataset Results on MPI-INF-3DHP. In Table 3, we evaluate the generalization ability of our method by comparing it against strong baselines. Our model is trained on Human3.6M and evaluated on the MPI-INF-3DHP dataset. Results demonstrate that our approach achieves the highest PCK and second best AUC scores, consistently outperforming the baseline methods across various indoor and outdoor scenes. Compared to ICFNet, the best performing baseline, our model shows a relative improvement of 0.5% in terms of the PCK metric. While ICFNet achieves a slightly higher AUC score, PoseKAN remains highly competitive.

Model Efficiency. PoseKAN achieves superior pose estimation performance while being significantly more efficient, utilizing only 5.72M parameters, much lower than GraphMLP's 9.49M parameters. Although our model incurs a slight computational overhead compared to GCN-based methods, it significantly improves performance.

4.3. Ablation Study

Effect of Residual Connection. We analyze the impact of the initial residual connection (IRC) within the layer-wise propagation rule on our model's performance. The results, summarized in Table 4, indicate that incorporating IRC leads to notable performance improvements, reducing MPJPE and PA-MPJPE by 1.65% and 1.49%, respectively. By preserving and reinforcing the initial node features across layers, IRC not only stabilizes the learning process but also mitigates information loss, allowing for more reliable pose predictions.

Table 4. Effect of initial residual connection (IRC) on model performance.

Method	MPJPE (↓)	PA-MPJPE (↓)
Without IRC	34.44	28.79
With IRC	**33.51**	**28.01**

Effect of Spline Order and Grid Size. In Figure 3 (left), we observe that the best performance is achieved at a spline order of 3. Increasing the spline order beyond this point does not lead to further improvements and may introduce unnecessary complexity. Similarly, Figure 3 (right) shows that the best results are obtained at a grid size of 5. While a sufficiently high spline order and grid size allow for more adaptive function approximations, excessively large values lead to increased computational costs.

Effect of Scaling Factor. By varying the scaling factor s from 0 and 1, we control how much weight is given to local versus global dependencies within the graph structure. This ability to adjust the influence of distant nodes is particularly valuable when learning graph representations that need to

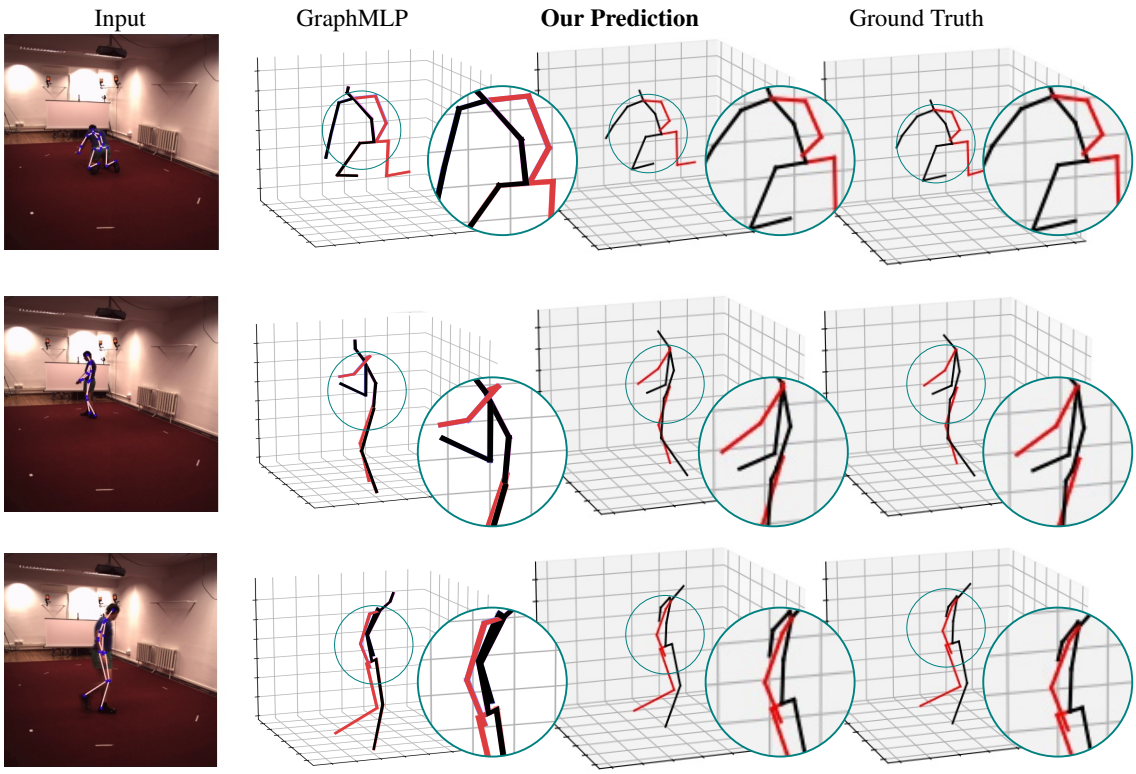

Figure 2. Visual comparison between PoseKAN and GraphMLP on sample actions from the Human3.6M dataset.

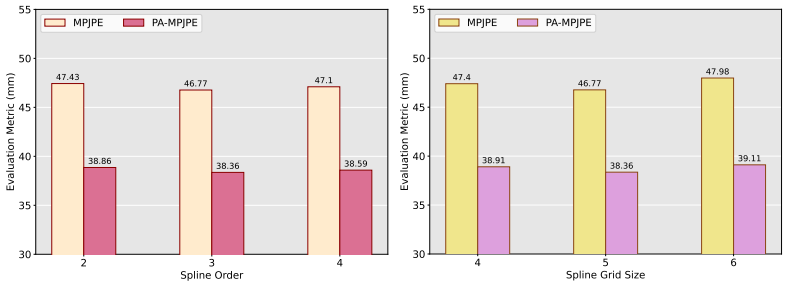

Figure 3. Effect of spline order and grid size.

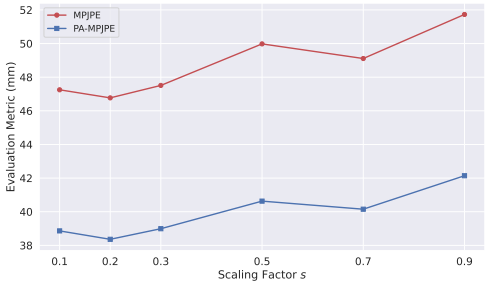

Figure 4. Effect of scaling factor.

capture global patterns and long-range dependencies, which are essential for tasks like 3D human pose estimation. Figure 4 illustrates that smaller scaling parameters often lead to better results, with our model achieving the lowest error values at $s = 0.2$. This suggests that a moderate emphasis on local information, while still considering distant nodes, strikes the good balance for our model.

5. Conclusion

In this paper, we introduced an adaptive graph Kolmogorov-Arnold Network (PoseKAN), a novel approach for 3D human pose estimation that leverages learnable function-based transformations. Unlike GCN-based methods that rely on fixed activation functions, PoseKAN employs learnable univariate functions on graph edges, providing greater flexibility and expressiveness in modeling human skeletal structures. By incorporating multi-hop feature propagation, our model effectively captures both local and long-range dependencies, mitigating the challenges posed by occlusions and depth ambiguities in 3D pose estimation. Our extensive experiments on benchmark datasets demonstrate that PoseKAN outperforms state-of-the-art methods, while also achieving good performance in cross-dataset evaluations, proving its generalization ability to unseen data. For future work, we aim to extend PoseKAN to multi-person pose estimation tasks and explore its applicability to other graph-based learning problems, such as action recognition and anomaly detection.

References

[1] Soubarna Banik, Alejandro Mendoza Gracia, and Alois Knoll. 3D human pose regression using graph convolutional network. In *Proc. IEEE International Conference on Image Processing*, pages 924–928, 2021. 6

[2] Jürgen Braun and Michael Griebel. On a constructive proof of Kolmogorov's superposition theorem. *Constructive Approximation*, 30:653–675, 2009. 2, 3

[3] Roman Bresson, Giannis Nikolentzos, George Panagopoulos, Michail Chatzianastasis, Jun Pang, and Michalis Vazirgiannis. KAGNNs: Kolmogorov-Arnold networks meet graph learning. *Transactions on Machine Learning Research*, 2025. 1

[4] Richard L. Burden, J. Douglas Faires, and Annette M. Burden. *Numerical Analysis*. Cengage Learning, 2015. 3

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018. 4, 5

[6] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3D human pose estimation. In *Proc. IEEE International Conference on Computer Vision*, pages 2262–2271, 2019. 1, 2, 7

[7] 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, 2013. 2, 5

[8] Thomas N. Kipf and Max Welling. Semi supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. 2

[9] Jae Yung Lee and I Gil Kim. Multi-hop modulated graph convolutional networks for 3D human pose estimation. In *Proc. British Machine Vision Conference*, 2022. 2, 5, 6, 7

[10] Kyoungoh Lee, Inwoong Lee, and Sanghoon Lee. Propagating LSTM: 3D pose estimation based on joint interdependency. In *Proc. European Conference on Computer Vision*, pages 119–135, 2018. 6

[11] Chen Li and Gim Hee Lee. Weakly supervised generative network for multiple 3D human pose hypotheses. In *Proc. British Machine Vision Conference*, 2020. 6

[12] Wenhao Li, Mengyuan Liu, Hong Liu, Tianyu Guo, Ti Wang, Hao Tang, and Nicu Sebe. GraphMLP: A graph MLP-like architecture for 3D human pose estimation. *Pattern Recognition*, 2025. 2, 5, 6, 1

[13] Kenkun Liu, Rongqi Ding, Zhiming Zou, Le Wang, and Wei Tang. A comprehensive study of weight sharing in graph networks for 3D human pose estimation. In *Proc. European Conference on Computer Vision*, pages 318–334, 2020. 1, 7, 2

[14] Kenkun Liu, Zhiming Zou, and Wei Tang. Learning global pose features in graph convolutional networks for 3D human pose estimation. In *Proc. Asian Conference on Computer Vision*, 2020. 1, 2

[15] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljacic, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov-Arnold networks. In *International Conference on Learning Representations*, 2025. 2, 3

[16] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3D human pose estimation. In *Proc. IEEE International Conference on Computer Vision*, pages 2640–2649, 2017. 1, 2, 5

[17] 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 *Proc. International Conference on 3D Vision*, 2017. 5

[18] Sungheon Park, Jihye Hwang, and Nojun Kwak. 3D human pose estimation using convolutional neural networks with 2D pose information. In *Proc. European Conference on Computer Vision*, pages 156–169. Springer, 2016. 1

[19] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3D human pose. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 7025–7034, 2017. 1, 2

[20] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3D human pose estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 7307–7316, 2018. 6, 7

[21] Jianning Quan and A Ben Hamza. Higher-order implicit fairing networks for 3D human pose estimation. In *Proc. British Machine Vision Conference*, 2021. 5, 6, 7, 1, 2

[22] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred A. Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *Proc. International Conference on Maching Learning*, 2019. 1

[23] Johannes Schmidt-Hieber. The Kolmogorov-Arnold representation theorem revisited. *Neural Networks*, 137:119–126, 2021. 2, 3

[24] Abu Taib Mohammed Shahjahan and A. Ben Hamza. Flexible graph convolutional network for 3D human pose estimation. In *Proc. British Machine Vision Conference*, 2024. 2, 5, 7

[25] Saurabh Sharma, Pavan Teja Varigonda, Prashast Bindal, Abhishek Sharma, and Arjun Jain. Monocular 3D human pose estimation by generation and ordinal ranking. In *Proc. IEEE International Conference on Computer Vision*, pages 2325–2334, 2019. 6

[26] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional human pose regression. In *Proc. IEEE International Conference on Computer Vision*, pages 2602–2611, 2017. 1

[27] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proc. European Conference on Computer Vision*, pages 529–545, 2018. 1, 2

[28] Jue Wang, Shaoli Huang, Xinchao Wang, and Dacheng Tao. Not all parts are created equal: 3D pose estimation by modeling bi-directional dependencies of body parts. In *Proc. IEEE International Conference on Computer Vision*, pages 7771–7780, 2019. 7

[29] Yong Wang, Peng Liu, Hongbo Kang, Doudou Wu, and Duoqian Miao. ICFNet: Interactive-complementary fusion network for monocular 3D human pose estimation. *Neurocomputing*, 2025. 7

[30] Yixuan Wang, Jonathan W. Siegel, Ziming Liu, and Thomas Y. Hou. On the expressiveness and spectral bias of KANs. In *International Conference on Learning Representations*, 2025. 2

[31] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. ConvNeXt V2: Co-designing and scaling ConvNets with masked autoencoders. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 16133–16142, 2023. 4

[32] Haiping Wu and Bin Xiao. 3D human pose estimation via explicit compositional depth maps. In *Proc. AAAI Conference on Artificial Intelligence*, pages 12378–12385, 2020. 1, 2

[33] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3D human pose estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 16105–16114, 2021. 7

[34] Yuanlu Xu, Wenguan Wang, Tengyu Liu, Xiaobai Liu, Jianwen Xie, and Song-Chun Zhu. Monocular 3D pose estimation via pose grammar and data augmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 6

[35] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. SRNet: Improving generalization in 3D human pose estimation with a split-and-recombine approach. In *Proc. European Conference on Computer Vision*, pages 507–523, 2020. 7

[36] Ailing Zeng, Xiao Sun, Lei Yang, Nanxuan Zhao, Minhao Liu, and Qiang Xu. Learning skeletal graph neural networks for hard 3D pose estimation. In *Proc. IEEE International Conference on Computer Vision*, pages 11436–11445, 2021. 7

[37] Zijian Zhang. Group graph convolutional networks for 3D human pose estimation. In *Proc. British Machine Vision Conference*, 2022. 1, 2, 5, 6, 7

[38] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3D human pose regression. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 3425–3435, 2019. 1, 2, 5, 6

[39] Kun Zhou, Xiaoguang Han, Nianjuan Jiang, Kui Jia, and Jiangbo Lu. HEMlets pose: Learning part-centric heatmap triplets for accurate 3D human pose estimation. In *Proc. IEEE International Conference on Computer Vision*, pages 2344–2353, 2019. 7

[40] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B*, 60(1):301–320, 2005. 5

[41] Zhiming Zou and Wei Tang. Modulated graph convolutional network for 3D human pose estimation. In *Proc. IEEE International Conference on Computer Vision*, pages 11477–11487, 2021. 1, 2, 5, 6

[42] Zhiming Zou, Kenkun Liu, Le Wang, and Wei Tang. High-order graph convolutional networks for 3D human pose estimation. In *Proc. British Machine Vision Conference*, 2020. 2, 5, 6, 1

[43] Zhiming Zou, Tianqi Liu, Dapeng Wu, and Wei Tang. Compositional graph convolutional networks for 3D human pose estimation. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, 2021. 1, 5, 6, 7