

SGAFormer: Skeleton-Graph and Agent-guided Transformer network for monocular video-based 3D human pose estimation

Min Li, Lina Du, Yuanyao Lu, Ning Wang, Zixuan Xu

North China University of Technology, Beijing, China
limin@ncut.edu.cn

Abstract. Monocular video-based 3D human pose estimation remains challenging due to depth ambiguity, noisy 2D observations, and complex spatio-temporal dependencies. Existing Transformer-based methods can capture global relationships, but they often lack explicit skeletal topology constraints and may introduce redundant dense temporal interactions. To address these issues, this paper proposes a Skeleton-Graph and Agent-guided Transformer network (SGAFormer) for skeleton-aware and temporally coherent 3D human pose estimation from 2D pose sequences. In spatial modeling, the proposed Skeleton-constrained Adaptive Graph-order Attention Module (SAGA) introduces skeleton-constrained dynamic graph attention and multi-order graph propagation to capture joint self-information, direct skeletal connections, and indirect structural dependencies. By adaptively fusing different graph-order branches, SAGA enhances dynamic joint interaction modeling under physical skeleton constraints. In temporal modeling, the proposed Agent-token Global-local Temporal Attention Module (AGTA) reorganizes dense frame-wise interactions through agent tokens and combines temporal positional bias with depthwise separable temporal convolution to enhance global-local motion representation. Experiments on Human3.6M show that SGAFormer remains competitive under detected 2D keypoint inputs, achieves superior temporal consistency in terms of MPJVE, and obtains strong reconstruction accuracy with ground-truth 2D keypoints, demonstrating the effectiveness of the proposed spatial-temporal modeling framework.

Keywords: Monocular 3D human pose estimation, Skeleton-constrained graph attention, Transformer, Agent token, Spatio-temporal modeling.

1 Introduction

Monocular 3D Human Pose Estimation (3D HPE) is a fundamental task in the field of computer vision, aiming to recover the three-dimensional coordinates of human joints from monocular images or video sequences. This technology has broad application value in action recognition [1] and related scenarios such as augmented reality, human-computer interaction, and sports rehabilitation [2]. Compared with multi-view or depth-sensor-based approaches, monocular methods offer advantages such as low hardware cost and convenient deployment. However, because monocular input inherently lacks

depth information and is easily affected by human self-occlusion, errors in 2D keypoint detection, motion blur, and complex motion patterns, recovering 3D human structure from 2D projections remains a highly ill-posed problem.

Existing monocular 3D pose estimation methods mainly include end-to-end image regression and two-stage pose lifting. End-to-end methods directly regress 3D joints from images but are sensitive to background, illumination, and occlusion. In contrast, two-stage methods first estimate 2D keypoints and then lift them to 3D poses [3–5]. Martinez et al. demonstrated the effectiveness of a simple residual 2D-to-3D network [3], while VideoPose3D exploited temporal continuity to alleviate depth ambiguity [4].

Recently, Transformer-based methods have achieved significant progress through self-attention modeling [6–8]. PoseFormer introduced spatial and temporal Transformers for intra-frame and inter-frame modeling [8]. MHFormer reduced 2D-to-3D uncertainty using multi-hypothesis representations [9], and MixSTE enhanced spatiotemporal modeling by alternately stacking spatial and temporal Transformers [10]. Subsequent methods, including STCFormer, PoseFormerV2, MotionAGFormer, and KTPFormer, further improved performance through spatiotemporal cross-attention, frequency-domain modeling, Transformer-GCN fusion, and kinematic priors [11–14].

Despite these advances, several limitations remain. First, monocular 3D pose estimation is highly dependent on the quality of 2D observations, and noisy 2D keypoints caused by occlusion, motion blur, or detection errors can significantly degrade the accuracy of 3D pose lifting. Second, existing spatial modeling methods often rely on fixed skeleton topology, unconstrained graph learning, or global self-attention. Although graph-based methods improve adaptive spatial modeling [15–17], fixed topology lacks flexibility, while unconstrained graph learning may introduce invalid or physically implausible joint connections. Third, standard temporal self-attention directly models all pairwise frame relationships, which may introduce redundant temporal interactions and provides limited explicit modeling of local motion patterns [18–20]. Recent studies further indicate that both long-range temporal dependencies and local motion details are important for video-based 3D pose estimation [21,22].

Motivated by these observations, we propose SGAFormer, a Skeleton-Graph and Agent-guided Transformer network for monocular video-based 3D human pose estimation. Specifically, the proposed SAGA module leverages skeletal topology priors and order-aware graph attention to capture dynamic joint relationships and multi-order spatial dependencies under physical skeleton constraints. In the temporal dimension, AGTA introduces agent tokens as intermediate temporal representations to reorganize dense frame-wise interactions, and combines temporal positional bias with depthwise separable temporal convolution to represent global temporal context and local motion patterns. By integrating skeleton-constrained spatial modeling with agent-guided global-local temporal modeling, SGAFormer improves the representation of spatio-temporal pose dependencies from 2D pose sequences. The main contributions of this paper are as follows:

- (1) A Skeleton-constrained Adaptive Graph-order Attention Module (SAGA) is designed to couple anatomical topology constraints with adaptive multi-order graph reasoning. By constructing and fusing self-loop, first-order, and second-

- order graph branches, SAGA captures joint self-information, direct skeletal connections, and indirect structural dependencies.
- (2) An Agent-token Global-local Temporal Attention Module (AGTA) is introduced to reorganize dense frame-wise temporal interactions through agent tokens while preserving local motion dynamics. By combining agent-guided global aggregation with temporal positional bias and depthwise separable temporal convolution, AGTA enhances global-local temporal representation.
 - (3) Experiments are conducted on the Human3.6M dataset [23]. The results demonstrate the effectiveness of the proposed SGAFormer.

2 Related Work

Monocular 3D human pose estimation has been widely studied in recent years. Existing methods can be divided into end-to-end regression methods and two-stage pose lifting methods. Since two-stage approaches decouple 2D pose detection from 3D reconstruction, they have become a popular paradigm for video-based 3D pose estimation [3–5]. Martinez et al. proposed a simple residual network to lift 2D joints into 3D space [3], and VideoPose3D further introduced temporal convolutions to capture motion dependencies across frames [4]. Subsequent studies improve robustness by exploiting motion-guided temporal modeling and feature fusion under noisy inputs [24,25]. These works show that accurate 3D pose estimation requires not only effective spatial dependency modeling among joints, but also reliable temporal modeling across frames.

To explicitly describe spatial relationships, graph-based methods have been introduced into skeleton representation learning. The human skeleton naturally forms a graph structure, where joints are regarded as nodes and bones are regarded as edges. Graph Convolutional Networks (GCNs) propagate features along skeletal connections [26], while Graph Attention Networks (GATs) assign adaptive weights to neighboring joints [15]. ST-GCN further modeled skeleton sequences as spatio-temporal graphs for human motion representation [27]. In 3D pose estimation, early graph-based methods usually rely on fixed anatomical adjacency matrices, which preserve physical skeleton topology but are less flexible in modeling pose-dependent joint interactions. To improve adaptability, Modulated GCN introduces learnable node and adjacency modulation [28], while Conditional Directed GCN models hierarchical motion dependencies through directed skeletal graphs [29]. Recent methods further integrate graph structures with attention mechanisms, such as GLA-GCN, GraFormer, MotionAGFormer, HDFormer, and STGFormer [13,16,17,30,31]. However, many of them rely on fixed or implicitly learned graph relations, or use attention mainly for feature aggregation.

Driven by advances in sequence modeling, Transformers have been widely adopted for video-based 3D human pose estimation. Compared with convolutional and graph-based models, Transformers are effective in capturing long-range dependencies through self-attention [7]. PoseFormer introduced spatial and temporal Transformers to model joint-level relations and frame-level dependencies [8]. MHFormer alleviates depth ambiguity through multi-hypothesis representations [9], while MixSTE alternately models spatial correlations and temporal trajectories [10]. Subsequent studies

further improve long-sequence modeling and pose robustness by incorporating strided temporal modeling, frequency-domain representations, cross attention, motion pre-training, and kinematic priors [6,11,12,14,18,21,22,32]. Although these methods enhance global spatio-temporal representation learning, most of them mainly rely on data-driven attention for joint interaction modeling. Consequently, explicit skeletal topology is not always sufficiently emphasized in these methods.

Beyond the issue of skeletal structural priors, temporal self-attention computes dense frame-to-frame relationships, which may introduce redundant interactions in long video sequences. To reorganize temporal interactions, StridedFormer reduces computation through temporal downsampling [18], while the Hourglass Tokenizer selects informative tokens to balance accuracy and representation compactness [19]. Agent Attention introduces a small number of agent tokens to aggregate global information and propagate it back to original tokens [20], providing a useful perspective for modeling temporal dependencies. However, monocular 3D pose estimation requires both global motion context and fine-grained local temporal cues, such as short-term joint displacement and velocity variation. Therefore, effective temporal modeling should capture global frame-level dependencies and local motion dynamics, which remains challenging for existing methods.

Despite recent progress, two issues remain insufficiently explored: adaptively modeling multi-order joint dependencies while preserving anatomical topology, and capturing local motion details while reducing redundant temporal interactions. To address these issues, we propose SGAFormer for monocular video-based 3D human pose estimation. SGAFormer introduces SAGA for skeleton-constrained dynamic multi-order spatial modeling and AGTA for agent-guided global-local temporal interaction. By combining skeleton-aware spatial modeling with agent-guided global-local temporal modeling, SGAFormer learns effective spatio-temporal representations for accurate and temporally consistent 3D pose estimation.

3 Methodology

3.1 Overview

The overall procedure of the proposed SGAFormer is illustrated in Fig. 1. Given an input 2D pose sequence $X_{in} \in \mathbb{R}^{F \times J \times 2}$, where F denotes the number of frames and J denotes the number of joints, the objective is to predict the corresponding 3D pose sequence $\hat{Y} \in \mathbb{R}^{F \times J \times 3}$. First, the input 2D joint sequence X_{in} is fed into the Skeleton-constrained Adaptive Graph-order Attention (SAGA) module, which introduces human skeletal topology constraints and performs multi-order graph propagation to extract skeleton-aware spatial features. Then, the obtained features are processed by the Spatial Transformer encoder, where intra-frame joint dependencies are further modeled through spatial self-attention. After that, the spatial features are rearranged along the temporal dimension and passed into the AGTA-based Temporal Transformer encoder. In this stage, agent tokens are used to reorganize frame-wise temporal interactions, while temporal positional bias and depthwise separable temporal convolution are employed to represent global temporal context and local motion patterns. The spatial and

temporal encoders can be stacked to progressively refine spatio-temporal representations. Finally, the encoded features are fed into a regression head, which maps the learned representations to 3D joint coordinates and outputs the predicted pose sequence $\hat{Y}$. A detailed explanation of our proposed method is as follows.

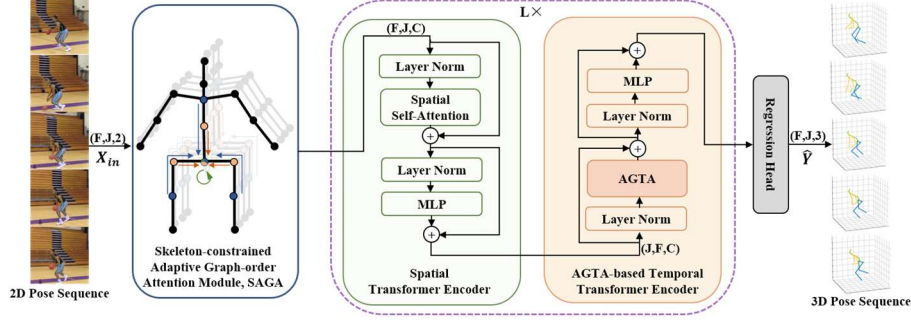


Fig. 1. Overview of the proposed SGAFormer. SAGA first extracts skeleton-aware spatial features from the input 2D pose sequence. The resulting features are then processed by alternating Spatial Transformer and AGTA-based Temporal Transformer encoders for spatiotemporal representation learning, followed by a regression head for 3D pose prediction.

3.2 Skeleton-constrained Adaptive Graph-order Attention Module

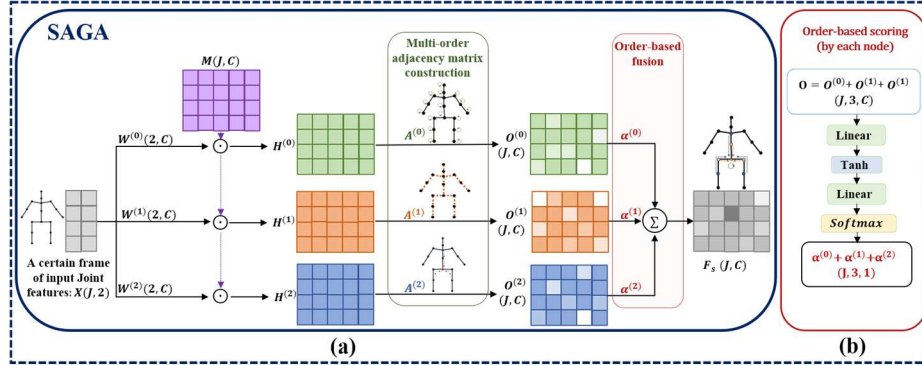


Fig. 2. Architecture of the proposed SAGA module. Given a single-frame 2D joint feature X , SAGA first constructs multi-order adjacency matrices, then performs graph-order propagation on self-loop, first-order, and second-order branches, and finally fuses the propagated features through order-aware attention to obtain the skeleton-aware spatial representation F_s .

The human skeleton can be naturally represented as a graph, where joints are treated as nodes and bones are treated as edges. To extract skeleton-aware spatial features from 2D joint coordinates, we propose the Skeleton-constrained Adaptive Graph-order Attention (SAGA) module. Different from fixed graph methods or unconstrained learnable graphs, SAGA uses the physical human skeleton as a structural prior and learns adaptive spatial relations within anatomically valid connections. As shown in Fig. 2,

SAGA follows a three-stage procedure: first, it constructs multi-order adjacency matrices, including the self-loop adjacency $A^{(0)}$, the first-order dynamic adjacency $A^{(1)}$, and the second-order adjacency $A^{(2)}$; Then, the input joint features are transformed by order-specific projections and propagated through three parallel graph-order branches to obtain self-loop, first-order, and second-order spatial features; finally, it adaptively fuses the propagated features through an order-aware attention mechanism. In this way, SAGA jointly models joint self-information, direct skeletal connections, and indirect structural dependencies.

Given the 2D joint feature of a single frame $X \in \mathbb{R}^{J \times 2}$, where J denotes the number of human joints, SAGA constructs multi-order adjacency matrices for spatial feature propagation, as illustrated in Fig. 3. The self-loop adjacency $A^{(0)}$ preserves each joint’s own information, the first-order dynamic adjacency $A^{(1)}$ models adaptive interactions between physically connected joints, and the second-order adjacency $A^{(2)}$ further captures indirect two-hop dependencies. In this way, SAGA preserves valid skeletal topology while enhancing dynamic multi-order spatial modeling.

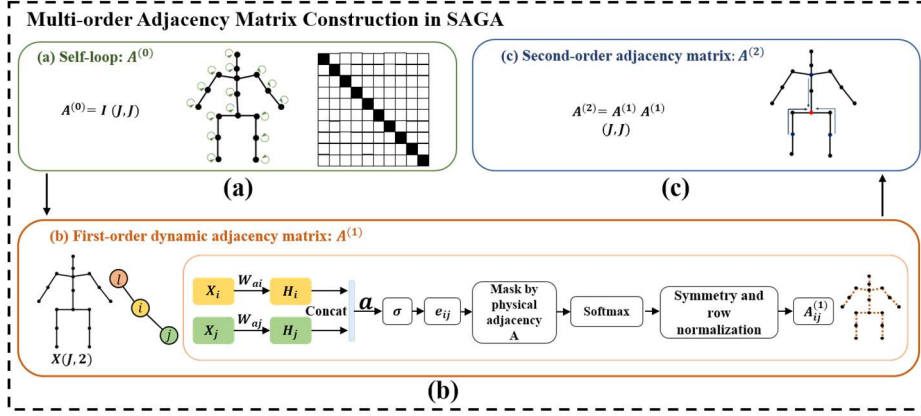


Fig. 3. Multi-order adjacency matrix construction in SAGA. (a) $A^{(0)}$ preserves each joint’s own information. (b) $A^{(1)}$ learns adaptive weights between physically connected joints under the skeleton mask. (c) $A^{(2)}$ is derived from $A^{(1)}$ to capture indirect two-hop joint dependencies.

As shown in Fig. 3(a), the self-loop adjacency matrix $A^{(0)}$ is defined as an identity matrix:

$$A^{(0)} = I(J, J), \quad (1)$$

where each joint is connected only to itself. This branch is used to retain the original information of each joint during spatial propagation and prevents the joint representation from being dominated by neighboring joints.

Fig. 3(b) shows the construction of the first-order dynamic adjacency matrix $A^{(1)}$. SAGA first defines a physical adjacency matrix $A \in \mathbb{R}^{J \times J}$ according to the human skeleton topology. If joint i and joint j are directly connected by a bone, then $A[i, j] = 1$; otherwise, $A[i, j] = 0$. This physical adjacency matrix is used as a structural mask to

ensure that dynamic relation learning is performed only between anatomically valid neighboring joints. The input joint features X are first projected into a latent space: $H = XW_a$, where $W_a \in \mathbb{R}^{2 \times d}$ is a learnable projection matrix, For any two joints i and j , their correlation score is computed as:

$$e_{ij} = \text{LeakyReLU}(a^\top [H_i || H_j]), \quad (2)$$

where $[\cdot || \cdot]$ denotes feature concatenation, and a is a learnable attention vector. Then, the physical adjacency matrix A is used to mask invalid non-skeletal connections:

$$\tilde{e}_{ij} = \begin{cases} e_{ij}, & A_{ij} = 1 \\ -\infty, & A_{ij} = 0 \end{cases} \quad (3)$$

In this way, only physically connected joints are retained for dynamic adjacency learning, while invalid non-skeletal connections are excluded. For each joint i , a soft-max operation is then performed over its valid first-order neighboring joints:

$$R_{ij}^{(1)} = \frac{\exp(\tilde{e}_{ij})}{\sum_{k \in \mathcal{N}(i)} \exp(\tilde{e}_{ik})}, \quad (4)$$

where $\mathcal{N}(i)$ denotes the set of first-order neighboring joints of joint i . Thus, $R^{(1)}$ forms an input-adaptive first-order adjacency matrix, where each valid skeletal edge is assigned a dynamic propagation weight according to the current pose. To stabilize graph propagation, $R^{(1)}$ is further symmetrized and row-normalized:

$$\bar{R}^{(1)} = \frac{R^{(1)} + (R^{(1)})^\top}{2}, \quad (5)$$

$$A_{ij}^{(1)} = \frac{\bar{R}_{ij}^{(1)}}{\sum_{j=1}^J \bar{R}_{ij}^{(1)} + \epsilon}, \quad (6)$$

where ϵ is a small constant used to avoid division by zero. The resulting $A^{(1)}$ is the first-order dynamic adjacency matrix used in SAGA.

Although $A^{(1)}$ captures adaptive interactions between directly connected joints, many human pose dependencies are established through multi-hop skeletal paths. For example, the wrist may be related to the shoulder through the elbow, and the ankle may be influenced by both the knee and hip. Therefore, as shown in Fig. 3(c), SAGA further constructs the second-order adjacency matrix $A^{(2)}$ from the first-order dynamic adjacency matrix: $A^{(2)} = A^{(1)}A^{(1)}$. At the element-wise level, the second-order relationship between joint i and joint j is computed as:

$$A_{ij}^{(2)} = \sum_{k=1}^J A_{ik}^{(1)} A_{kj}^{(1)}, \quad (7)$$

where k denotes an intermediate joint. This formulation accumulates the path weights from joint i to joint j through all possible intermediate joints. Since $A^{(2)}$ is derived from the dynamic first-order adjacency matrix, it remains sample-adaptive and can capture indirect joint dependencies under different motion patterns.

After obtaining $A^{(0)}$, $A^{(1)}$, and $A^{(2)}$, SAGA performs multi-order graph propagation, as shown in Fig. 2. Three learnable transformation matrices $W^{(0)}$, $W^{(1)}$, $W^{(2)}$ are used to project the input features for different graph-order branches. The transformed node features are computed as:

$$H^{(k)} = (XW^{(k)}) \odot M, \quad k \in \{0, 1, 2\}, \quad (8)$$

where $M \in \mathbb{R}^{J \times C}$ is a node-channel modulation parameter, $\odot$ denotes element-wise multiplication, and C is the output feature dimension. The modulation parameter M adaptively adjusts the response strength of different joints and feature channels. The three graph propagation branches are then formulated as:

$$O^{(0)} = A^{(0)}H^{(0)}, O^{(1)} = A^{(1)}H^{(1)}, O^{(2)} = A^{(2)}H^{(2)}. \quad (9)$$

The self-loop branch $O^{(0)}$ preserves joint-specific information, the first-order branch $O^{(1)}$ models dynamic interactions between directly connected joints, and the second-order branch $O^{(2)}$ captures indirect dependencies between joints. However, different actions and joints rely on different graph orders. Local movements may depend more on first-order adjacency, full-body coordination may benefit from second-order dependencies, while noisy joints may require stronger self-node information for stability. Therefore, SAGA introduces an order-aware attention mechanism to adaptively fuse the outputs of the three propagation branches. The branch outputs are first stacked as: $O = [O^{(0)}, O^{(1)}, O^{(2)}]$, where $O \in \mathbb{R}^{J \times 3 \times C}$. Then, a lightweight attention network assigns an importance score to each graph order for every joint:

$$s^{(k)} = f_{\text{attn}}(O^{(k)}), \quad (10)$$

$$\alpha^{(k)} = \frac{\exp(s^{(k)})}{\sum_{r=0}^2 \exp(s^{(r)})}, \quad (11)$$

where $f_{\text{attn}}(\cdot)$ consists of a linear layer, a nonlinear activation function, and a linear projection, and $\alpha^{(k)}$ denotes the importance weight of the k -th order propagation branch. Finally, the fused spatial feature of a single frame is obtained by: $F_s = \sum_{k=0}^2 \alpha^{(k)} O^{(k)}$. For clarity, the above formulation describes SAGA on a single frame. In practice, SAGA is applied to each frame of the input sequence with shared parameters. Therefore, for the whole input sequence $X_{in} \in \mathbb{R}^{F \times J \times 2}$, the output of SAGA is denoted as $F_s \in \mathbb{R}^{F \times J \times C}$. The resulting sequence feature F_s is further passed through batch normalization and a ReLU activation function to obtain the final skeleton-aware spatial representation. Through the above design, SAGA preserves the physical skeleton prior while adaptively modeling direct and indirect joint dependencies, providing structured spatial features for the subsequent Spatial Transformer encoder.

3.3 Spatial Transformer Encoder

The skeleton-aware feature F_s produced by SAGA is fed into a Spatial Transformer encoder to further capture global dependencies among joints within each frame. After

adding spatial positional encoding, the features are projected into Q, K and V , and standard multi-head self-attention is applied:

$$head_i = softmax(\frac{Q_i K_i^T}{\sqrt{d}}) V_i, \quad (12)$$

$$MHSA(Q, K, V) = Concat(head_1, head_2, \dots, head_h) W_o, \quad (13)$$

where W_o is a learnable output projection matrix. The output of MHSA is then combined with the input via a residual connection followed by layer normalization, and an MLP are used to obtain the encoded spatial feature $X_S \in \mathbb{R}^{F \times J \times C}$, which is subsequently rearranged for temporal modeling.

3.4 AGTA Based Temporal Transformer Encoder

After spatial encoding, the feature sequence $X_S \in \mathbb{R}^{F \times J \times C}$ is rearranged as $X_T \in \mathbb{R}^{J \times F \times C}$, so that the temporal evolution of each joint can be modeled independently. For a specific joint sequence $X_t \in \mathbb{R}^{F \times C}$, the proposed Agent-token Global-local Temporal Attention (AGTA) module aims to reorganize dense frame-to-frame interactions into an agent-guided temporal interaction process. As shown in Fig. 4, AGTA first generates compact agent tokens from the temporal sequence to summarize global motion context, then propagates the aggregated information back to each frame. Meanwhile, a local temporal convolution branch is introduced to preserve short-term motion details. In this way, AGTA enhances global-local temporal representation while avoiding purely dense frame-wise interaction.

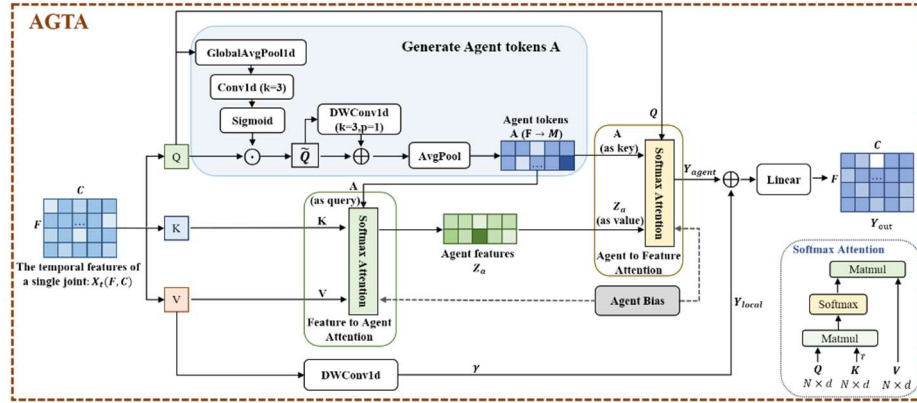


Fig. 4. Architecture of the proposed AGTA module. AGTA generates agent tokens from enhanced query features to aggregate global temporal context and propagate it back to frame features through a two-stage attention process. Meanwhile, a depthwise separable temporal convolution branch captures local motion patterns from value features, and the global and local temporal features are fused to obtain the final temporal representation.

Given the input temporal feature of a specific joint $X_t \in \mathbb{R}^{F \times C}$, AGTA first generates query, key, and value representations: $Q = X_t W_q, K = X_t W_k, V = X_t W_v$, where W_q, W_k , and W_v are learnable projection matrices. The features are then split into multiple attention heads: $Q, K, V \in \mathbb{R}^{H \times F \times D}$, where H is the number of attention heads and $D=C/H$. Different from standard temporal self-attention, which directly computes all $F \times F$ frame-wise relationships, AGTA uses a small number of agent tokens as intermediate temporal representations. To generate more informative agent tokens, AGTA first enhances the query features with channel-aware reweighting:

$$\tilde{Q} = Q \odot \text{sigmoid}(\text{Conv}(\text{GAP}(Q))). \quad (14)$$

This operation emphasizes informative motion channels before agent generation. Then, a depthwise separable temporal convolution is applied to introduce local temporal dynamics:

$$\hat{Q} = \tilde{Q} + \text{DWConv}(\tilde{Q}), \quad (15)$$

where $\text{DWConv}(\cdot)$ denotes depthwise separable temporal convolution. Adaptive average pooling is further used to compress the temporal dimension and generate a fixed number of agent tokens:

$$A = \text{Pool}(\hat{Q}), \quad (16)$$

where $A \in \mathbb{R}^{H \times M \times D}$, and M is the number of agent tokens. In this work, M is set to 32. These agent tokens do not correspond to specific frames, instead, they serve as compact semantic centers that summarize representative temporal patterns from the whole sequence. After obtaining the agent tokens, AGTA performs two-stage attention. In the first stage, the agent tokens act as queries to aggregate global temporal information from the full sequence:

$$Z_a = \text{Softmax}\left(\frac{AK^T}{\sqrt{D}} + B_{a \rightarrow t}\right)V, \quad (17)$$

where $B_{a \rightarrow t}$ is a learnable agent-to-time positional bias, and Z_a denotes the aggregated agent representation. This stage compresses long-range temporal context into a small set of agent tokens. In the second stage, the temporal frame features retrieve relevant information from the agent representations:

$$Y_{\text{agent}} = \text{Softmax}\left(\frac{QA^T}{\sqrt{D}} + B_{t \rightarrow a}\right)Z_a, \quad (18)$$

where $B_{t \rightarrow a}$ is a learnable time-to-agent positional bias. The two positional biases are interpolated when the input sequence length changes, allowing AGTA to adapt to different temporal lengths. Although the agent branch models global temporal context, fine-grained pose changes usually occur within local temporal ranges, such as short-term joint displacement and instantaneous motion variation. Therefore, AGTA introduces an additional local temporal branch on the value feature:

$$Y_{\text{local}} = \text{DWConv}(V), \quad (19)$$

where dilated depthwise temporal convolution is used to enlarge the local temporal receptive field and capture short-term motion patterns. The global agent-attention output and the local convolution output are fused in a residual manner:

$$Y = Y_{agent} + \gamma Y_{local}, \quad (20)$$

where γ is a learnable scaling coefficient initialized to a small value, preventing the local branch from dominating the global temporal representation at the early training stage. Finally, the fused multi-head features are concatenated and projected to obtain the output:

$$Y_{out} = Linear(Y), Y_{out} \in \mathbb{R}^{F \times C}. \quad (21)$$

For all joints, AGTA produces the temporal feature sequence $Y_{OUT} \in \mathbb{R}^{J \times F \times C}$, which is then rearranged back to $F \times J \times C$ for subsequent prediction. From the perspective of attention interaction scale, standard temporal self-attention computes $F \times F$ frame-wise interactions, while AGTA changes the main attention interaction scale to approximately $2FM$ through agent aggregation and information readout. When $F = 243$ and $M = 32$, this corresponds to 15552 interactions, about 26.3% of the 59049 interactions in standard self-attention. However, due to the additional channel reweighting, positional bias, and depthwise separable temporal convolution, the overall computational cost may not decrease. Therefore, AGTA is mainly used to reorganize dense frame-wise interactions and enhance global-local temporal representation rather than to reduce total computation.

3.5 Regression Head

After alternating Spatial Transformer and Temporal Transformer encoding, the model obtains spatiotemporal features integrating spatial structure and motion information. The regression head first applies layer normalization and then predicts 3D joint coordinates through a linear layer:

$$\hat{Y} = Linear(LN(Y_{OUT})), \quad (22)$$

where $\hat{Y} \in \mathbb{R}^{F \times J \times 3}$, representing the predicted 3D coordinates of all joints across all frames.

4 Experiments

4.1 Experimental Setup

Datasets and Evaluation Metric. We evaluate the proposed method on the Human3.6M dataset, a large-scale indoor motion capture benchmark for 3D human pose estimation. It contains 15 action categories performed by 11 subjects with 3D joint annotations captured by four synchronized high-definition cameras. Following previous works[9-10], subjects S1, S5, S6, S7, and S8 are used for training, while S9 and S11

are used for testing. Both CPN-detected and ground-truth 2D keypoints are used as input to evaluate the model under realistic detection noise and ideal 2D observation conditions. We adopt MPJPE, P-MPJPE, and MPJVE as evaluation metrics, which measure 3D joint error, Procrustes-aligned structural error, and temporal smoothness, respectively.

Implementation Details. The proposed method is implemented in PyTorch, and all experiments are conducted on a single NVIDIA GeForce RTX 3090 GPU. The model stacks $L = 8$ spatio-temporal encoder layers with $H = 8$ attention heads and feature dimension of $C = 512$. During training, we use the Adam optimizer with a batch size of 8. The initial learning rate is set to 0.00008 and decayed exponentially by a factor of 0.99 after each epoch.

4.2 Comparison with State-of-the-Art Methods

Table 1. Quantitative comparison on Human3.6M using CPN-detected 2D keypoints. The upper and lower parts report MPJPE under Protocol #1 and P-MPJPE under Protocol #2, respectively. T denotes the number of input frames. Lower values indicate better performance. The best and second-best results are shown in bold and underlined, respectively.

Protocol #1	T	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
PoseFormer[8]	81	41.5	44.8	39.8	42.5	46.5	51.6	42.1	42.0	53.3	60.7	45.5	43.3	46.1	31.8	32.2	44.3
MHFormer[9]	351	39.2	43.1	40.1	40.9	44.9	51.2	40.6	41.3	53.5	60.3	43.7	41.1	43.8	29.8	30.6	43.0
MixSTE[10]	243	37.6	<u>40.9</u>	37.3	39.7	42.3	49.9	40.1	39.8	51.7	<u>55.0</u>	<u>42.1</u>	39.8	<u>41.0</u>	27.9	27.9	40.9
GLA-GCN[30]	243	41.3	44.3	40.8	41.8	45.9	54.1	42.1	41.5	57.8	62.9	45	42.8	45.9	29.4	29.9	44.4
MSTFormer[25]	243	39.0	42.5	40.7	41.1	45.9	51.3	41.1	40.5	54.1	59.9	44.1	41.3	43.4	29.5	29.8	43.0
HMSFT[33]	81	44.8	44.0	41.0	42.6	41.3	45.5	41.8	42.1	45.6	56.2	43.3	41.3	45.9	35.3	34.7	43.0
HSA-DMT[22]	243	37.9	39.6	36.7	<u>38.3</u>	42.0	<u>48.2</u>	38.4	<u>38.2</u>	<u>50.7</u>	54.9	<u>42.1</u>	<u>38.8</u>	41.4	28.3	28.5	40.2
DMFormer[34]	243	<u>37.2</u>	41.3	37.3	39.1	43.1	49.8	40.1	37.7	52.9	57.2	<u>42.1</u>	39.6	41.7	<u>28.0</u>	28.6	41.0
SGA-Former(ours)	81	38.7	42.1	38.7	40.3	44.1	51.6	40.1	39.9	53.2	56.7	44.1	41.2	43.2	29.6	29.2	42.2
SGA-Former(ours)	243	36.7	39.6	<u>37.0</u>	37.7	<u>41.9</u>	48.7	<u>38.8</u>	38.7	51.3	56.5	41.6	38.6	40.7	<u>28.0</u>	<u>28.2</u>	<u>40.3</u>
Protocol #2	T	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
PoseFormer[8]	81	32.5	34.8	32.6	34.6	35.3	39.5	32.1	32.0	42.8	48.5	34.8	32.4	35.3	24.5	36.0	34.6
MixSTE[10]	243	30.8	33.1	30.3	31.8	33.1	39.1	31.1	30.5	42.5	44.5	34.0	30.8	32.7	<u>22.1</u>	22.9	32.6
GLA-GCN[30]	243	32.4	35.3	32.6	34.2	35.0	42.1	32.1	31.9	45.5	49.5	36.1	32.4	35.6	23.5	24.7	34.8
MSTFormer[25]	243	31.2	34.4	33.0	33.9	35.4	39.4	32.3	31.7	43.3	48.1	35.9	32.8	34.6	24.0	24.5	34.3
HMSFT[33]	81	30.9	32.4	32.4	32.0	30.7	33.0	29.6	30.7	36.4	44.1	33.4	29.3	32.9	25.2	25.6	31.9
HSA-DMT[22]	243	30.7	<u>32.5</u>	30.4	<u>31.5</u>	<u>32.4</u>	<u>37.6</u>	<u>30.1</u>	<u>29.7</u>	41.1	<u>44.5</u>	33.9	30.0	32.8	<u>22.1</u>	<u>23.1</u>	<u>32.2</u>
DMFormer[34]	243	29.7	32.9	<u>30.1</u>	31.1	33.4	38.7	31.0	29.2	42.9	46.4	34.1	30.3	32.3	22.0	23.3	32.5
SGA-Former(ours)	81	30.8	33.3	31.4	32.7	34.0	39.7	31.4	30.7	42.3	44.9	35.6	30.9	33.2	22.7	23.8	33.2
SGA-Former(ours)	243	<u>29.8</u>	32.7	30.0	31.1	<u>32.4</u>	38.2	30.7	29.8	<u>40.9</u>	46.6	<u>33.8</u>	<u>29.9</u>	<u>32.4</u>	22.2	23.2	32.3

Using 2D Keypoints Detected by CPN as Input. To evaluate the performance under practical 2D detection inputs, we use CPN-detected 2D keypoints and compare SGAFormer with representative methods on Human3.6M. As shown in Table 1, under Protocol #1, SGAFormer achieves an average MPJPE of 40.3 mm with 243-frame input, ranking second among all compared methods and only 0.1 mm higher than HSA-DMT. Compared with MixSTE and DMFormer, SGAFormer reduces the average error by 0.6 mm and 0.7 mm, respectively. Although SGAFormer does not obtain the best average

result, its action-level performance shows competitive or better results on several motion patterns. In particular, compared with HSA-DMT, SGAFormer achieves better or comparable results on actions such as Dir., Disc., Greet, Phone, Smoke, Wait, WalkD., Walk, and WalkT. These actions usually contain relatively stable skeletal trajectories or clear kinematic chains, indicating that SAGA can preserve valid anatomical topology while AGTA effectively captures continuous temporal dependencies.

For actions with severe occlusion, body folding, or fine-grained lower-limb ambiguity, such as Photo, SitD., and some sitting-related cases, SGAFormer still leaves room for improvement. As shown in Table 1, these actions are also difficult for most compared methods, indicating the inherent ambiguity under CPN-detected 2D inputs. A possible reason is that occlusion leads to inaccurate 2D joint localization, which propagates to 3D lifting. Although SAGA helps maintain valid skeletal topology and AGTA improves temporal coherence, they are still constrained by the quality of 2D observations. Nevertheless, SGAFormer remains competitive with HSA-DMT on average, suggesting that the proposed spatial-temporal modeling is effective under noisy inputs.

Under Protocol #2, SGAFormer achieves an average P-MPJPE of 32.3 mm with 243-frame input, which is only 0.1 mm higher than HSA-DMT. Although it does not obtain the best result, the gap is marginal, indicating that SGAFormer still maintains competitive structural consistency after Procrustes alignment. Moreover, increasing the input length from 81 to 243 frames improves MPJPE from 42.2 mm to 40.3 mm and P-MPJPE from 33.2 mm to 32.3 mm, demonstrating the benefit of longer temporal context for 3D pose estimation.

Overall, the results in Table 1 show that SGAFormer is competitive but not uniformly superior under CPN-detected 2D inputs. Its performance depends partly on the quality of detected 2D keypoints. Therefore, MPJPE and P-MPJPE alone do not fully reflect the temporal stability and intrinsic 3D reconstruction ability of the model. To further evaluate these aspects, we report MPJVE in Table 2 and conduct experiments using ground-truth 2D keypoints in Table 3. These complementary evaluations help distinguish whether the remaining errors mainly come from 2D detection noise or from the 3D pose estimation model itself.

Table 2. MPJVE comparison on Human3.6M using CPN-detected 2D keypoints. Lower values indicate better temporal consistency. The best results in each column are highlighted in bold.

MPJVE	T	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
PoseFormer[8]	81	2.7	2.8	2.0	3.1	2.0	2.4	2.4	2.8	1.8	2.4	2.0	2.1	3.4	2.7	2.4	2.5
MHFormer[9]	351	2.6	2.7	1.9	2.8	1.9	2.3	2.3	2.6	1.7	2.4	2.0	2.1	3.2	2.7	2.3	2.4
MixSTE[10]	243	2.5	2.7	1.9	2.8	1.9	2.2	2.3	2.6	1.6	2.2	1.9	2.0	3.1	2.6	2.2	2.3
HMSFT[33]	81	2.6	2.6	2.1	3.0	1.9	2.3	2.4	2.6	1.7	2.2	2.0	2.1	3.1	2.7	2.5	2.4
SGAFormer(ours)	243	2.5	2.6	1.9	2.7	1.8	2.2	2.2	2.6	1.5	2.1	1.8	2.0	3.1	2.5	2.1	2.2

In terms of temporal consistency, we use MPJVE to evaluate the inter-frame smoothness of predicted pose sequences. As shown in Table 2, SGAFormer achieves the lowest average MPJVE of 2.2 among all compared methods, reducing the average velocity error by 8.3% compared with HMSFT. It also obtains the best or tied-best results on most action categories, especially those involving continuous or subtle motion changes, such as Greet, Phone, Sit, SitD., Smoke, Walk, and WalkT. Since MPJVE measures

temporal stability beyond frame-wise joint accuracy, this result indicates that SGAFormer produces more coherent motion trajectories. Although SGAFormer is not consistently the best in all MPJPE or P-MPJPE results under noisy CPN inputs, its lower MPJVE suggests that AGTA helps enhance temporal coherence by combining global motion context with local temporal cues.

Table 3. Quantitative comparison on Human3.6M under Protocol #1 using ground-truth 2D keypoints as input. Lower MPJPE indicates better performance. The best and second-best results are shown in bold and underlined, respectively.

Protocol #1	T	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
PoseFormer[8]	81	30.0	33.6	29.9	31.0	30.2	33.3	34.8	31.4	37.8	38.6	31.7	31.5	29.0	23.3	23.1	31.3
MixSTE[10]	243	21.6	22.0	20.4	21.0	20.8	24.3	24.7	21.9	26.9	24.9	21.2	21.5	20.8	14.7	15.7	21.6
GLA-GCN[30]	243	20.1	21.2	20.0	19.6	21.5	26.7	23.3	<u>19.8</u>	27.0	29.4	20.8	20.1	<u>19.2</u>	12.8	13.8	21.0
MSTFormer[25]	243	27.9	31.9	28.7	28.3	29.8	33.8	33.4	30.2	37.2	37.7	30.4	29.8	28.6	21.1	22.5	30.1
HMSFT[33]	81	29.9	28.9	22.5	29.7	25.4	26.2	33.6	29.1	25.2	29.7	27.1	29.2	28.0	26.5	27.3	27.9
HSA-DMT[22]	243	<u>20.3</u>	<u>20.2</u>	<u>19.2</u>	<u>19.2</u>	<u>19.5</u>	22.2	<u>21.8</u>	20.8	23.1	23.9	19.5	<u>19.2</u>	19.8	13.8	15.0	<u>19.8</u>
DMFormer[34]	243	20.7	21.3	19.5	20.0	22.0	25.0	23.9	<u>19.8</u>	25.9	27.2	21.5	20.4	20.3	<u>13.5</u>	<u>14.4</u>	21.0
SGA-Former(ours)	243	20.1	19.5	18.6	18.6	19.2	<u>22.7</u>	20.8	19.4	<u>23.5</u>	<u>24.4</u>	<u>20.1</u>	18.6	18.8	14.3	15.2	19.6

Using Ground-Truth 2D Keypoints as Input. To evaluate the upper-bound performance under ideal 2D observations, we further test SGAFormer using ground-truth 2D keypoints. Different from CPN-detected inputs, ground-truth 2D keypoints largely eliminate detection errors and provide a clearer evaluation of the 3D lifting capability of each model. As shown in Table 3, SGAFormer achieves the best average MPJPE of 19.6 mm with 243-frame input, outperforming all compared methods. In particular, compared with HSA-DMT and HMSFT, SGAFormer reduces the average MPJPE by 1.0% and 29.7%, respectively. This result indicates that when the influence of 2D detection noise is removed, the proposed model demonstrates stronger intrinsic 3D reconstruction ability.

Compared with HSA-DMT, SGAFormer obtains better or comparable results on several actions, including Dir., Disc., Eat, Greet, Phone, Pose, Pur., Wait, and WalkD. These improvements suggest that the skeleton-constrained adaptive graph modeling in SAGA is effective for capturing multi-order joint dependencies under reliable 2D observations. Meanwhile, the global-local temporal representation introduced by AGTA further helps maintain coherent motion cues across frames. Compared with HMSFT, SGAFormer shows a clear average improvement, indicating that the proposed spatial-temporal design is effective when accurate 2D structural information is available. Therefore, the relatively small gaps observed under CPN-detected inputs in Table 1 are likely related to the quality of 2D keypoint detection rather than a fundamental limitation of the 3D pose estimation model. This finding further supports the effectiveness of SGAFormer in 3D pose lifting under reliable 2D inputs.

4.3 Ablation Studies

Effectiveness of Each Module. To evaluate the contribution of SAGA and AGTA, we conduct ablation studies on Human3.6M, as shown in Table 4. we report both pose

accuracy and computational cost (parameters and FLOPs) for each module. The baseline model adopts the same overall framework as SGAFormer, but removes SAGA and employs a standard Transformer encoder for temporal modeling, where multi-head self-attention is used to capture frame-wise temporal dependencies. Based on this baseline, SAGA and AGTA are introduced separately and jointly to evaluate their individual and combined contributions.

Table 4. Ablation study and complexity comparison of SAGA and AGTA on Human3.6M under Protocol #1 using CPN-detected 2D keypoints.

Methods	SAGA	AGTA	Params(M)	FLOPs(G)	MPJPE↓
Baseline			33.651	139.038	40.9
+ SAGA	✓		33.913	142.298	40.4
+ AGTA		✓	33.697	139.174	40.6
SGAFormer (ours)	✓	✓	33.942	142.433	40.3

From the results in Table 4, we can observe that the baseline model obtains 40.9 mm MPJPE. SAGA alone reduces MPJPE to 40.4 mm, indicating that skeleton-constrained dynamic graph modeling effectively enhances spatial structure representation. When only AGTA is used to replace the standard temporal self-attention, the MPJPE decreases to 40.6 mm, showing that the agent-guided temporal interaction also improves temporal representation compared with the baseline. When SAGA and AGTA are jointly adopted, the full SGAFormer achieves the best result of 40.3 mm.

In terms of complexity, SAGA introduces a modest increase in parameters and FLOPs, while AGTA adds negligible computational overhead compared with the baseline. The full SGAFormer increases the parameter count by only 0.291M (0.86%) and the FLOPs by 3.4G (2.44%), achieving the lowest MPJPE in this ablation setting. These results demonstrate that SAGA and AGTA are complementary: SAGA primarily enhances skeleton-aware spatial representation, whereas AGTA improves global-local temporal modeling. Overall, the proposed design achieves consistent performance gains with only marginal additional computational cost.

4.4 Qualitative Visualization Analysis.

Figure 5 shows the qualitative visualization results on the Purchases action. The predictions of MixSTE and SGAFormer are generally similar, indicating that both methods can recover the main body configuration in this action. Nevertheless, SGAFormer still shows slightly better structural consistency in several challenging details. For example, in the third row, where the elbow and knee appear visually close in the projected view, our result is slightly more consistent with the ground truth in terms of limb arrangement and joint alignment. This suggests that the proposed SAGA module helps model skeleton-aware spatial dependencies, while AGTA provides complementary temporal cues for more stable pose estimation. Therefore, although the visual differences are not dramatic, the proposed method still shows better structural consistency in challenging poses with possible limb overlap or body folding.

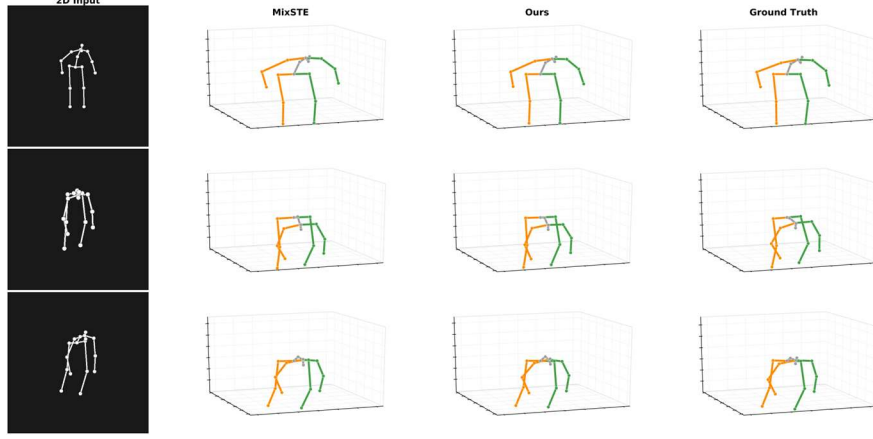


Fig. 5. Qualitative visualization results on the Purchases action. Each row shows one representative frame, and the columns from left to right show the 2D input, MixSTE prediction, SGAFormer prediction, and Ground Truth.

5 Conclusion

In this paper, we propose SGAFormer, a skeleton-graph and agent-guided Transformer network for monocular video-based 3D human pose estimation. The proposed framework aims to enhance spatial-temporal representation learning by integrating skeleton-constrained spatial reasoning with agent-guided temporal interaction modeling. Specifically, SAGA introduces skeleton-constrained graph attention and multi-order graph propagation to capture dynamic joint interactions and structural dependencies. Meanwhile, AGTA organizes temporal information through agent-guided interactions and local temporal modeling to enhance motion representation. Experiments on Human3.6M show that SGAFormer achieves competitive performance under CPN-detected 2D inputs and strong reconstruction accuracy with ground-truth 2D inputs, while improving temporal consistency. Ablation studies verify that SAGA and AGTA provide complementary spatial and temporal representations, and their combination yields the lowest MPJPE. These results demonstrate the effectiveness of integrating skeleton-constrained spatial reasoning with agent-guided temporal modeling for video-based 3D human pose estimation.

References

1. Zhou, Y., Yan, X., Cheng, Z.-Q., Yan, Y., Dai, Q., Hua, X.-S.: BlockGCN: redefine topology awareness for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2049–2058 (2024).
2. Ji, X., Fang, Q., Dong, J., Shuai, Q., Jiang, W., Zhou, X.: A survey on monocular 3D human pose estimation. *Virtual Reality & Intelligent Hardware* 2(6), 471–500 (2020).

3. Martinez, J., Hossain, R., Romero, J., Little, J.J.: A simple yet effective baseline for 3D human pose estimation. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2640–2649 (2017).
4. Pavlo, D., Feichtenhofer, C., Grangier, D., Auli, M.: 3D human pose estimation in video with temporal convolutions and semi-supervised training. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7753–7762 (2019).
5. Chen, C.H., Tyagi, A., Agrawal, A., Drover, D., MV, R., Stojanov, S., Rehg, J.M.: Unsupervised 3D pose estimation with geometric self-supervision. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5714–5724 (2019).
6. Zhu, W., Ma, X., Liu, Z., Liu, L., Wu, W., Wang, Y.: MotionBERT: a unified perspective on learning human motion representations. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15085–15099 (2023).
7. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: *Advances in Neural Information Processing Systems*, vol. 30 (2017).
8. Zheng, C., Zhu, S., Mendieta, M., Yang, T., Chen, C., Ding, Z.: 3D human pose estimation with spatial and temporal transformers. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11656–11665 (2021).
9. Li, W., Liu, H., Tang, H., Wang, P., Van Gool, L.: MHFormer: Multi-hypothesis transformer for 3D human pose estimation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13147–13156 (2022).
10. Zhang, J., Tu, Z., Yang, J., Chen, Y., Yuan, J.: MixSTE: Seq2seq mixed spatio-temporal encoder for 3D human pose estimation in video. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13232–13242 (2022).
11. Tang, Z., Qiu, Z., Hao, Y., Hong, R., Yao, T.: 3D human pose estimation with spatiotemporal criss-cross attention. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4790–4799 (2023).
12. Zhao, Q., Zheng, C., Liu, M., Wang, P., Chen, C.: PoseFormerV2: Exploring frequency domain for efficient and robust 3D human pose estimation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8877–8886 (2023).
13. Mehraban, S., Adeli, V., Taati, B.: MotionAGFormer: enhancing 3D human pose estimation with a Transformer-GCNFormer network. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6920–6930 (2024).
14. Peng, J., Zhou, Y., Mok, P.Y.: KTPFormer: Kinematics and trajectory prior knowledge-enhanced transformer for 3D human pose estimation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1123–1132 (2024).
15. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: *International Conference on Learning Representations* (2018).
16. Chen, H., He, J.Y., Xiang, W., Liu, H., Liu, W., Luo, B.: HDFormer: High-order directed Transformer for 3D human pose estimation. *arXiv preprint arXiv:2302.01825* (2023).
17. Liu, Y., Zhang, Z.: STGFormer: Spatio-temporal GraphFormer for 3D human pose estimation in video. *Pattern Recognition* 171, 112239 (2026).
18. Li, W., Liu, H., Ding, R., Liu, M., Wang, P., Yang, W.: Exploiting temporal contexts with strided transformer for 3D human pose estimation. *IEEE Trans. Multimedia* 25, 1282–1293 (2023).
19. Li, W., Liu, M., Liu, H., Tang, H., Wang, P., Van Gool, L.: Hourglass tokenizer for efficient transformer-based 3D human pose estimation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 604–613 (2024).

20. Han, D., Ye, T., Han, Y., Xia, Z., Pan, S., Wan, P., Song, S., Huang, G.: Agent Attention: On the integration of softmax and linear attention. In: European Conference on Computer Vision. Springer, Cham (2024).
21. Qian, X., Tang, Y., Zhang, N., Han, M., Xiao, J., Huang, M., Lin, R.: HSTFormer: Hierarchical spatial-temporal transformers for 3D human pose estimation. arXiv preprint arXiv:2301.07322 (2023).
22. Chen, J., Zhou, Z., Xie, B., Zhu, J., Li, Y.: HSA-DMT: Hierarchical structure-aware embedding and dual-mode temporal modeling for challenging 3D human pose estimation. *Multimedia Systems* 32(2), 101 (2026).
23. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Trans. Pattern Anal. Mach. Intell.* 36(7), 1325–1339 (2014).
24. Wang, J., Yan, S., Xiong, Y., Lin, D.: Motion guided 3D pose estimation from videos. In: European Conference on Computer Vision, pp. 764–780. Springer, Cham (2020).
25. Lin, H., Xu, S., Su, C.: MSTFormer: Multi-granularity spatial-temporal transformers for 3D human pose estimation. *J. King Saud Univ. Comput. Inf. Sci.* 37(3), 15 (2025).
26. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (2017).
27. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32(1), pp. 7444–7452. (2018).
28. Zou, Z., Tang, W.: Modulated graph convolutional network for 3D human pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 11477–11487 (2021).
29. Hu, W., Zhang, C., Zhan, F., Zhang, L., Wong, T.T.: Conditional directed graph convolution for 3D human pose estimation. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 602–611 (2021).
30. Yu, X., Zhang, Z., Liu, Y., Zhong, S., Liu, Y., Chen, C.W.: GLA-GCN: Global-local adaptive graph convolutional network for 3D human pose estimation from monocular video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8818–8829 (2023).
31. Zhao, W., Wang, W., Tian, Y.: GraFormer: Graph-oriented transformer for 3D pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20438–20447 (2022).
32. Shan, W., Liu, Z., Zhang, X., Wang, S., Ma, S., Gao, W.: P-STMO: Pre-trained spatial temporal many-to-one model for 3D human pose estimation. In: European Conference on Computer Vision, pp. 461–478. Springer, Cham (2022).
33. Zhang, H., Hu, Z., Bi, S., Di, J., Sun, Z.: HMSFT: Hierarchical multiscale spatial-frequency-temporal collaborative transformer for 3D human pose estimation. *Pattern Recognit.* 164, 111562 (2025).
34. Pan, S., Zhi, Y., Wang, S., Chen, A.: Convolution-enhanced dual-stream mixed transformer for 3D human pose estimation in video. *Signal Image Video Process.* 20, 253 (2026).