HiPART: Hierarchical Pose AutoRegressive Transformer for Occluded 3D Human Pose Estimation

Hongwei Zheng[*], Han Li[*], Wenrui Dai[†], Ziyang Zheng[†], Chenglin Li, Junni Zou, Hongkai Xiong
Shanghai Jiao Tong University, Shanghai, China
{1424977324, qingshi9974, daiwenrui, zhengziyang, lcl1985, zoujunni, xionghongkai}@sjtu.edu.cn

Abstract

Existing 2D-to-3D human pose estimation (HPE) methods struggle with the occlusion issue by enriching information like temporal and visual cues in the lifting stage. In this paper, we argue that these methods ignore the limitation of the sparse skeleton 2D input representation, which fundamentally restricts the 2D-to-3D lifting and worsens the occlusion issue. To address these, we propose a novel two-stage generative densification method, named Hierarchical Pose AutoRegressive Transformer (HiPART), to generate hierarchical 2D dense poses from the original sparse 2D pose. Specifically, we first develop a multi-scale skeleton tokenization module to quantize the highly dense 2D pose into hierarchical tokens and propose a Skeleton-aware Alignment to strengthen token connections. We then develop a Hierarchical AutoRegressive Modeling scheme for hierarchical 2D pose generation. With generated hierarchical poses as inputs for 2D-to-3D lifting, the proposed method shows strong robustness in occluded scenarios and achieves state-of-the-art performance on the single-frame-based 3D HPE. Moreover, it outperforms numerous multi-frame methods while reducing parameter and computational complexity and can also complement them to further enhance performance and robustness.

1. Introduction

3D human pose estimation (HPE) from monocular images has attracted wide attention due to its flexibility in adaptation on various devices. It is usually decoupled into the *2D estimation* and *3D lifting* stages in existing methods [27, 52], where the *2D estimation* stage estimates 2D poses from input images via off-the-shelf 2D detectors [5, 28, 33], and subsequently, the *3D lifting* stage achieves 2D-to-3D estimation with a lifting model. Such decoupling enables large-scale training with a large amount of 3D MOCAP data [27] by circumventing the scarce labeled 2D/3D ground truth poses for monocular images. However, the real-world application of 3D HPE faces a challenge of the occlusion problem (*i.e.*, self-occlusion and object occlusion), where the unreliable occluded 2D poses could lead to corrupted 3D lifted poses.

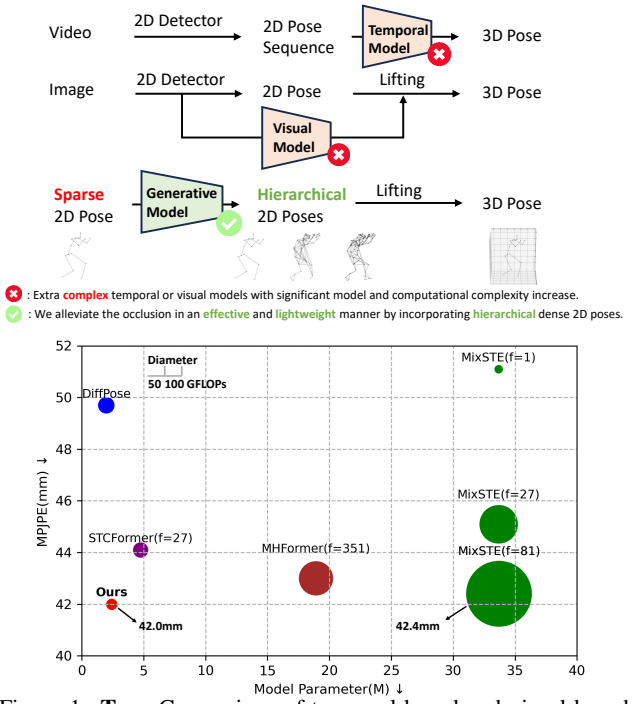


Figure 1. **Top:** Comparison of temporal-based and visual-based methods with our densification approach at the framework level. These methods enrich information in the lifting stage, while ours address a more fundamental issue, *i.e.*, the sparse 2D pose input. **Bottom:** Comparison of parameters, GFLOPs, and MPJPE across various methods on Human3.6M. The circle size indicates GFLOPs for inference. Our method achieves the SOTA result with reduced complexity compared to the temporal-based methods.

Recent works have introduced additional information in the lifting stage to alleviate the occlusion issues. Some methods leverage topology priors (*e.g.*, kinematic priors [21, 31, 41] and geometric constraints [2, 46]) to im-

[*]Equal Contribution
[†]Corresponding Authors

prove the structural understanding of joint correlations, but often prove ineffective in severe occlusion. Others utilize enriched context (*e.g.*, temporal [25, 30, 54] and visual cues [26, 35, 52, 55]) to improve the occluded joints inference, which require extra complex temporal or visual encoders with increased parameter and computational complexity (see top of Fig. 1). In essence, all these methods primarily focus on enriching the information in the lifting stage while overlooking the inherent limitation of the current input 2D representation, ***the sparse 2D skeleton representation***, which fundamentally constrains the performance of 2D-to-3D lifting and worsens the occlusions. Specifically, current 2D pose datasets typically represent the human body with few keypoints (*e.g.*, 17 joints for Human3.6M [18]). This sparse input for lifting inherently limits the ability to exploit local context. In this paper, we argue that *developing a* ***hierarchical dense 2D skeleton representation*** *is essential for exploring the intricate skeletal context, thus enhancing the robustness of occluded 2D-to-3D lifting.*

We conduct a toy experiment to demonstrate the benefit of this hierarchical representation. We collect the dataset of hierarchical dense 2D poses by progressively coarsening the 3D ground truth mesh of Human3.6M (6890 vertices $\rightarrow$ 96 joints $\rightarrow$ 48 joints) and projecting them into the 2D pixel space. When integrating these hierarchical dense 2D poses into the 2D-to-3D lifting stage, MPJPE is significantly improved by 55% from 37.6mm to 17.5mm, as shown in Appendix E). This indicates the significance of richer skeletal information. For example, as shown in the bottom of Fig. 2, when the right wrist is occluded, the sparse 2D pose only provides one elbow joint for reference, while the denser 2D poses offer multiple arm joints to help predict the occlusion region. However, acquiring such hierarchical dense 2D poses remains a challenge, as the 3D ground truth mesh is not available in real-world 3D HPE scenarios.

Inspired by recent autoregressive models [10, 24, 32, 36, 38], in this paper, we propose a novel hierarchical 2D pose densification method named Hierarchical Pose AutoRegressive Transformer (HiPART) to address the challenge. HiPART obtains reliable hierarchical dense 2D poses conditioned on the original sparse 2D pose input in a generative fashion with two main modules. First, we design a Multi-Scale Skeletal Tokenization (MSST) module to progressively quantize a highly dense 2D pose into hierarchical discrete tokens via VQ-VAE [38], and introduce a Skeleton-aware Alignment strategy to strengthen connections across multi-scale tokens. Second, we develop a Hierarchical AutoRegressive Modeling (HiARM) scheme for hierarchical dense 2D pose generation. Unlike the conventional next-token prediction widely used in image and video generation [3, 10, 45], we present novel *sparse-to-dense* and *center-to-periphery* strategies specifically designed for the non-Euclidean skeletal structure and show their efficiency.

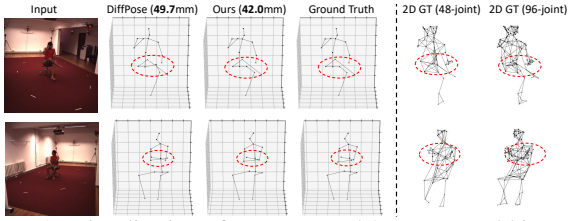


Figure 2. Visualization of reconstructed 3D poses and hierarchical 2D poses on Human3.6M under occlusions. Ours outperforms DiffPose [13] due to the rich skeletal context in hierarchical poses.

Therefore, HiPART generates hierarchical 2D poses to provide enhanced local context for subsequent 2D-to-3D lifting.

We perform extensive experiments to validate the effectiveness and efficiency of HiPART. Specifically, we show that, using only a *vanilla spatial transformer* for lifting, HiPART achieves state-of-the-art performance in the single-frame setting on various 3D HPE benchmarks, including Human3.6M [18], 3DPW [39], and 3DPW-Occ [50], with strong robustness in occluded scenarios. Compared to methods that rely on complex temporal encoders to incorporate temporal cues, HiPART is superior or comparable with significantly reduced complexity, as shown in the bottom of Fig. 1. Note that HiPART is orthogonal to existing temporal methods and can integrate these methods to further enhance performance and robustness. The contributions of this paper are summarized below.

- To our best knowledge, we achieve the first hierarchical dense 2D pose generation scheme to address the occlusion problem for 3D HPE from single monocular image.
- We develop a multi-scale skeleton quantization model via VQ-VAE along with a Skeleton-aware Alignment to enhance token connections.
- We design a novel hierarchical pose autoregression strategy specifically for non-Euclidean skeleton structures.
- We employ the proposed 2D pose generation scheme in 2D-to-3D lifting to achieve state-of-the-art performance on the single-frame setting and outperforms numerous multi-frame methods.

2. Related Work

Occlusions in 3D HPE. Existing approaches typically incorporate additional information in the lifting stage. Topology priors (*e.g.*, kinematic priors [21, 31, 41] and geometric constraints [2, 46]) are utilized to enhance joint correlations, often failing in severely occluded scenarios. Other approaches introduce enriched context(*e.g.*, temporal context [25, 30, 34, 54] and visual cues [26, 53, 55]) to improve the prediction accuracy for occluded joints. While highly effective, these methods consist of advanced temporal or image encoders, significantly increasing model and computational complexity. We alleviate the occlusion problem in

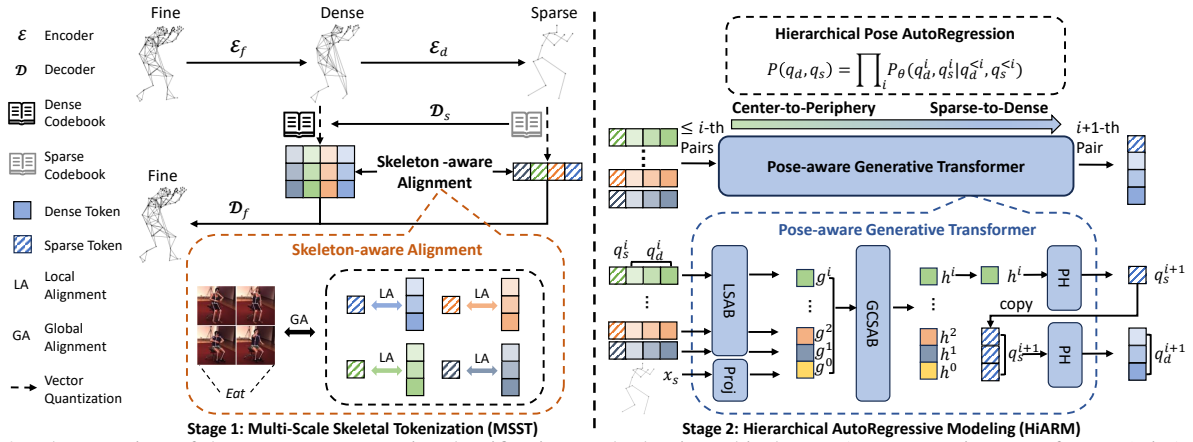


Figure 3. The overview of the two-stage generative densification method, Hierarchical Pose AutoRegressive Transformer (HiPART). In Stage 1, the MSST module progressively quantizes the fine 2D pose into hierarchical tokens with Skeleton-aware Alignment. We omit the reconstruction of the dense 2D pose $\hat{\mathbf{x}}_d$ for simplicity. In Stage 2, we propose a HiARM scheme to generate $(i+1)$-th pair tokens based on a series of indices less than $i+1$. One pair of discrete tokens contains a single sparse token and r dense tokens corresponding to the related part. Finally, generated hierarchical 2D poses are fed to a vanilla spatial transformer for subsequent 2D-to-3D lifting.

an effective and lightweight manner by incorporating hierarchical dense 2D poses as input. It addresses the sparse 2D pose input issue, which is overlooked by existing works.

Hierarchy in 3D HPE. Hierarchical information is essential in Human Mesh Recovery (HMR) [19, 20, 29, 49], as the human body can be represented at multi-level, from sparse joints to dense vertices, which contains rich skeletal context and can help address occlusions in 3D HPE. For instance, Pose2Mesh [7] uses a learnable transformation matrix to estimate poses in a coarse-to-fine manner. HGN [22] employs a hierarchical network to reconstruct poses, transitioning from sparse to dense representations. However, both methods regard the hierarchical 3D pose prediction as an auxiliary task, neglecting the accuracy of the hierarchical 2D poses, and leading to minimal performance gains. We propose a generative densification approach that enhances 2D hierarchical pose estimation and 3D lifting precision.

Generative Models for 3D HPE. Generative models have shown promising results in various domains, including image, text, and audio generation [4, 14, 15, 17, 44], and have recently been applied in 3D HPE. For example, diffusion models progressively refine pose distributions, reducing uncertainty during estimation [8, 13, 16]. However, they suffer from low inference speed due to reliance on numerous sampling steps. A vector-quantized-based method, *i.e.*, PCT [12], uses a two-stage process to construct a codebook capturing sub-structures and casts pose estimation as a classification task. While effective, the classification approach struggles to fully model the latent distribution of discrete tokens. We address this in an auto-regression manner in Stage 2 and introduce Skeleton-aware Alignment to strengthen token connections. Besides, our method has a faster generation speed than diffusion models due to the posed-based auto-regression. More discussion is in the Appendix D.

3. Collection of Hierarchical 2D Poses

The goal of our method is to generate reliable hierarchical 2D poses with denser joints conditioned on the original sparse 2D pose $\mathbf{x}_s \in \mathbb{R}^{J_s \times 2}$ with J_s joints, benefiting subsequent 2D-to-3D lifting. However, existing datasets lack hierarchical dense 2D poses that could serve as ground truth for training. To address this, we use the 3D human mesh from Human3.6M [18] and apply the dense vertices coarsening method [7] along with camera projection to create the dense and fine (*i.e.*, highly dense) 2D poses, denoted as $\mathbf{x}_d \in \mathbb{R}^{J_d \times 2}$ and $\mathbf{x}_f \in \mathbb{R}^{J_f \times 2}$, where $J_f > J_d > J_s$. The detailed process is described in the Appendix B.

4. Hierarchical Pose AutoRegressive Transformer (HiPART)

The overview of our HiPART is shown in Fig. 3. We first learn the Multi-Scale Skeletal Tokenization (MSST) module to quantize the fine $\mathbf{x}_f \in \mathbb{R}^{J_f \times 2}$ into hierarchical discrete tokens, *i.e.,* sparse tokens $\mathbf{q}_s \in \mathbb{Z}^{J_s}$ and dense token $\mathbf{q}_d \in \mathbb{Z}^{J_d}$, where different colors in Fig. 3 indicate different parts of the human body, and different saturations represent the number (r) of dense tokens. Then, we learn a pose-aware generative transformer by the Hierarchical AutoregRessive Modeling (HiARM) scheme to generate hierarchical tokens conditioned on the sparse 2D pose $\mathbf{x}_s$. With the decoders of HiPART, we can reconstruct the hierarchical dense 2D poses from the hierarchical tokens $\mathbf{q}_s$ and $\mathbf{q}_d$.

4.1. Multi-Scale Skeletal Tokenization (MSST)

Multi-Scale Skeletal Quantized-Autoencoder. We use an architecture similar to VQ-VAE-2 [32] to construct our multi-scale skeletal quantized autoencoder. Encoders and decoders are implemented with MLP-Mixer [37], which

is well-suited for skeletal data. Specifically, two encoders $\mathcal{E}_f(\cdot)$ and $\mathcal{E}_d(\cdot)$ sequentially project the ground truth $\mathbf{x}_f$ into dense and sparse pose embeddings, $\mathbf{z}_d \in \mathbb{R}^{J_d \times D}$ and $\mathbf{z}_s \in \mathbb{R}^{J_s \times D}$, where D is the embedding dimension.

Next, $\mathbf{z}_s$ is quantized into sparse tokens $\mathbf{q}_s \in \mathbb{Z}^{J_s}$ via a quantizer $\mathcal{Q}(\cdot)$ using the sparse codebook $\mathbf{C}_s \in \mathbb{R}^{K_s \times D}$, which contains K_s learnable vectors. Each element of $\mathbf{q}_s$ then looks up $\mathbf{C}_s$ to obtain $\hat{\mathbf{z}}_s$, *i.e.*, the approximation of $\mathbf{z}_s$:

$$\mathbf{q}_s = \mathcal{Q}(\mathbf{z}_s; \mathbf{C}_s), \quad \hat{\mathbf{z}}_s = \mathbf{C}_s[\mathbf{q}_s]. \tag{1}$$

Subsequently, $\hat{\mathbf{z}}_s$ is upsampled into $\hat{\mathbf{z}}'_s \in \mathbb{R}^{J_d \times D}$ using decoder $\mathcal{D}_s(\cdot)$. This upsampling embedding is then concatenated with $\mathbf{z}_d$ to form $\mathbf{z}'_d$, which is quantized into dense tokens $\mathbf{q}_d \in \mathbb{Z}^{J_d}$ using the dense codebook $\mathbf{C}_d \in \mathbb{R}^{K_d \times D}$, which contains K_d vectors. The approximated dense pose embeddings $\hat{\mathbf{z}}_d$ is obtained by looking up $\mathbf{C}_d$ from $\mathbf{q}_d$.

$$\mathbf{q}_d = \mathcal{Q}(\mathbf{z}'_d; \mathbf{C}_d), \quad \hat{\mathbf{z}}_d = \mathbf{C}_d[\mathbf{q}_d], \tag{2}$$

Finally, the reconstructed dense pose $\hat{\mathbf{x}}_d$ is decoded using decoder $\mathcal{D}_d(\cdot)$ given $\hat{\mathbf{z}}_s$, and the reconstructed fine pose $\hat{\mathbf{x}}_f$ is decoded using decoder $\mathcal{D}_f(\cdot)$ given $\hat{\mathbf{z}}_s$ and $\hat{\mathbf{z}}_d$.

Skeleton-aware Alignment. To enhance the tokenization process for skeletal data, we propose a Skeleton-aware Alignment strategy. This strategy incorporates skeletal structure into the tokenization process, improving the mutual connections across different skeleton levels.

Part-wise Local Alignment (LA): The part-wise Local Alignment is introduced to align the sparse and dense tokens within the same local body part, forming a more unified representation. The Info Noise Contrastive Estimation (Info-NCE) loss is used to match part-wise pairs:

$$\mathcal{L}_{local} = \sum_{i=1}^{J_s} -\log\left(\frac{\exp(\hat{\mathbf{z}}_s^i \cdot \text{avg}(\hat{\mathbf{z}}_d^i)/\tau)}{\sum_{k=1}^{J_s}\exp(\hat{\mathbf{z}}_s^i \cdot \text{avg}(\hat{\mathbf{z}}_d^k)/\tau)}\right), \tag{3}$$

where τ is the temperature parameter, $\text{avg}(\cdot)$ is the average function. The r approximated dense pose embeddings $\hat{\mathbf{z}}_d^i = \{\hat{\mathbf{z}}_d^{(i,j)}\}_{j=1,2,\ldots,r}$ correspond to the related part of single $\hat{\mathbf{z}}_s^i$, where the ratio $r = J_d/J_s$.

Action-wise Global Alignment (GA): We further introduce an action-wise Global Alignment to align the hierarchical tokens with the action label of the pose (contained in the Human3.6M [18] dataset), thus offering consistent semantic information for the discrete spaces. We define the global loss $\mathcal{L}_{global}$ as a cross-entropy loss between the predicted classification vector $\mathbf{p}_A$ and the action label $\mathbf{y}_A$:

$$\mathbf{p}_A = \mathcal{P}_A(\text{concat}(\hat{\mathbf{z}}_s, \hat{\mathbf{z}}_d)), \tag{4}$$

$$\mathcal{L}_{global} = \text{CrossEntropy}(\mathbf{y}_A, \mathbf{p}_A), \tag{5}$$

where $\mathcal{P}_A(\cdot)$ is the action projector composed of two MLP-Mixer blocks, $\text{concat}(\cdot)$ is the concatenate operation.

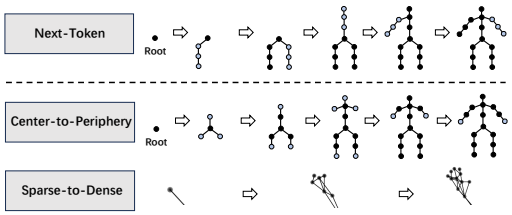


Figure 4. Standard auto-regressive modeling (**Top**) vs Our proposed hierarchical pose auto-regressive modeling (**Bottom**).

Loss Function. The Stage 1 loss $\mathcal{L}_1$ includes the reconstruction loss for the poses and hierarchical pose embeddings, as well as the loss of Skeleton-aware Alignment.

$$\mathcal{L}_1 = \|\mathbf{x}_f - \hat{\mathbf{x}}_f\|_2^2 + \|\mathbf{x}_d - \hat{\mathbf{x}}_d\|_2^2 + \sum_{j\in\{s,d\}}\Big(\|\text{sg}(\mathbf{z}_j) - \hat{\mathbf{z}}_j\|_2^2 + \beta\|\mathbf{z}_j - \text{sg}(\hat{\mathbf{z}}_j)\|_2^2\Big) + \lambda_l \cdot \mathcal{L}_{local} + \lambda_g \cdot \mathcal{L}_{global}, \tag{6}$$

where β controls the commitment loss, λ_l and λ_g are weighting factors for the local and Global Alignment losses, $sg()$ is the stop gradient operation. Codebooks are updated via Exponential Moving Average.

4.2. Hierarchical AutoRegressive Modeling (HiARM)

In Stage 2, we propose Hierarchical AutoRegressive Modeling (HiARM) for hierarchical dense 2D pose generation. We first define a hierarchical pose auto-regressive objective function and then present the model structure details.

Objective Function. We generate the discrete tokens in an auto-regression manner based on the learned codebooks in MSST. Our objective function aims at maximizing the log-likelihood of the i-th pair discrete tokens based on a series of indices less than i:

$$\mathbb{P}_\theta(\mathbf{q}_d, \mathbf{q}_s) = \prod_{i=1}^{J_s}\mathbb{P}_\theta(\mathbf{q}_d^i, q_s^i | \mathbf{q}_d^{<i}, \mathbf{q}_s^{<i}) \tag{7}$$

However, we argue that the commonly used next-token prediction order in image generation [3, 10] is unsuitable for pose generation. This is because it disregards the skeletal topology of human poses and suffers from slow generation speeds. As shown in Fig. 4, we propose a novel autoregressive manner specifically designed for the non-Euclidean skeletal structure.

Center-to-periphery. [23] has revealed that the joints farther from the root joint tend to exhibit larger uncertainty. Inspired by it, we generate joints progressively from the center to the periphery, thus ensuring joints with lower uncertainty are first generated and alleviating the depth uncertainty.

Sparse-to-dense. Sparse tokens $\mathbf{q}_s$ capture comprehensive and coarse-grained information, while dense tokens $\mathbf{q}_d$ represent localized and fine-grained details. Based on the assumption that the local details of the pose can be generated

conditionally on the global summary, we can transform the expression of the likelihood into the following form:

$$\mathbb{P}_\theta(\mathbf{q}_d^i, q_s^i|\mathbf{q}_d^{<i}, \mathbf{q}_s^{<i}) = \mathbb{P}_\theta(\mathbf{q}_d^i|q_s^i, \mathbf{q}_d^{<i}, \mathbf{q}_s^{<i}) \cdot \mathbb{P}_\theta(q_s^i|\mathbf{q}_d^{<i}, \mathbf{q}_s^{<i}), \tag{8}$$

where r dense tokens are generated in parallel conditionally on the corresponding sparse token. We simplify the hierarchical generation process, reducing the original $1 + r$ steps to 2 steps, thus speeding up the pose generation process.

Model Structure. We detail the components of the pose-aware generative transformer as follows:

Local Self-Attention Blocks (LSAB). The approximation of i-th pair sparse and dense pose embeddings $\hat{\mathbf{z}}_s^i$ and $\{\hat{\mathbf{z}}_d^{(i,j)}\}_{j=1,2,\ldots,r}$ are refined with the LSAB, which models hierarchical tokens within the local body part, fully leveraging the information contained in multi-scale embeddings:

$$\{\mathbf{g}_j^i\}_{j=0,1,\ldots,r} \leftarrow \mathrm{LSAB}(\hat{\mathbf{z}}_s^i, \hat{\mathbf{z}}_d^{(i,1)}, \ldots, \hat{\mathbf{z}}_d^{(i,r)}), \tag{9}$$

where we adopt a bidirectional transformer with a self-attention block followed by an MLP block. The refined embeddings $\{\mathbf{g}_j^i\}_{j=0,1,\ldots,r}$ are then averaged to obtain the average refined embedding $\mathbf{g}^i$.

Global Causal Self-Attention Blocks (GCSAB). GCSAB generates a sequence of hidden embeddings $\{\mathbf{h}^k\}_{k=0,1,\ldots,i}$, capturing spatial relationships across the overall pose. Instead of using traditional next-token prediction order, we model the discrete tokens from the center to the periphery:

$$\{\mathbf{h}^k\}_{k=0,1,\ldots,i} \leftarrow \mathrm{GCSAB}(\mathbf{g}^0, \mathbf{g}^1, \ldots, \mathbf{g}^i), \tag{10}$$

where $\mathbf{g}^0$ represents the projection of sparse 2D pose $\mathbf{x}_s$ serving as the start-of-sentence (SOS) token.

Prediction Head (PH). This module predicts sparse and dense tokens in a hierarchical order, starting from the sparse to the dense. The sparse token is generated from the hidden embedding, followed by the parallel generation of r dense tokens based on the corresponding sparse token:

$$\{p_j^{i+1}\}_{j=0,1,\ldots,r} \leftarrow \mathrm{PH}(\mathbf{h}^i, \hat{\mathbf{z}}_s^i, \ldots, \hat{\mathbf{z}}_s^i), \tag{11}$$

where p_0^{i+1} and $\{p_j^{i+1}\}_{j=1,\ldots,r}$ denote the $(i+1)$-th pair predicted vectors of the sparse token and dense tokens. PH consists of a self-attention block with an MLP block.

Finally, the Stage 2 loss $\mathcal{L}_2$ model the latent distributions of sparse and dense tokens via the cross-entropy loss:

$$\mathcal{L}_2 = \mathrm{CrossEntropy}(\mathbf{q}_s, \mathbf{p}_s) + \lambda_d \cdot \mathrm{CrossEntropy}(\mathbf{q}_d, \mathbf{p}_d), \tag{12}$$

where $\mathbf{p}_s$ and $\mathbf{p}_d$ denote the predicted vectors for all sparse and dense tokens, and λ_d controls the dense token loss.

4.3. Inference Process

As shown in Algorithm 1, we first use the projection of the sparse 2D pose as a class-conditional SOS token to predict the initial sparse and dense tokens. The LSAB then encodes them for subsequent generations. This process is repeated in a hierarchical pose auto-regression manner until all discrete tokens are generated. Finally, we reconstruct the hierarchical dense 2D poses via the MSST decoder.

Algorithm 1 Inference process.

Input: The sparse 2D pose $\mathbf{x}_s$, the sparse and dense codebooks $\mathbf{C}_s$ and $\mathbf{C}_d$.
Output: The reconstructed dense and fine 2D pose $\hat{\mathbf{x}}_d$, $\hat{\mathbf{x}}_f$.

for $i = 0$ to J_s **do**
  if $i = 0$ **then**
    {Obtain initial tokens based on a SOS token}
    $\mathbf{g}^0 \leftarrow \mathcal{P}_c(\mathbf{x}_s)$
    $\mathbf{h}^0 \leftarrow \mathrm{GCSAB}(\mathbf{g}^0)$
    $q_s^1, \{q_d^{(1,j)}\}_{j=1,2,\ldots,r} \leftarrow \mathrm{PH}(\mathbf{h}^0, \hat{\mathbf{z}}_s^0, \ldots, \hat{\mathbf{z}}_s^0)$
  else
    {Center-to-periphery and Sparse-to-dense}
    $\hat{\mathbf{z}}_s^i \leftarrow \mathbf{C}_s(q_s^i)$, $\hat{\mathbf{z}}_d^i \leftarrow \mathbf{C}_d(\mathbf{q}_d^i)$
    $\{\mathbf{g}_j^i\}_{j=0,1,\ldots,r} \leftarrow \mathrm{LSAB}(\hat{\mathbf{z}}_s^i, \hat{\mathbf{z}}_d^{(i,1)}, \ldots, \hat{\mathbf{z}}_d^{(i,r)})$
    $\mathbf{g}^i = \mathrm{avg}(\mathbf{g}_0^i, \mathbf{g}_1^i, \ldots, \mathbf{g}_r^i)$
    $\{\mathbf{h}^k\}_{k=0,1,\ldots,i} \leftarrow \mathrm{GCSAB}(\mathbf{g}^0, \mathbf{g}^1, \ldots, \mathbf{g}^i)$
    $q_s^{i+1}, \{q_d^{(i+1,j)}\}_{j=1,2,\ldots,r} \leftarrow \mathrm{PH}(\mathbf{h}^i, \hat{\mathbf{z}}_s^i, \ldots, \hat{\mathbf{z}}_s^i)$
  end if
end for
$\hat{\mathbf{x}}_d \leftarrow \mathcal{D}_d(\mathbf{q}_d)$, $\hat{\mathbf{x}}_f \leftarrow \mathcal{D}_f(\mathbf{q}_d, \mathcal{D}_s(\mathbf{q}_s))$ **{Decode}**
return $\hat{\mathbf{x}}_d$, $\hat{\mathbf{x}}_f$

5. Experiments

5.1. Experimental Settings

Datasets. Human3.6M [18] is the most commonly used indoor dataset for 3D HPE. We follow prior works [30, 52, 53] to use five subjects (S1, S5, S6, S7, S8) for training and two subjects (S9, S11) for testing. 3DPW [39] is a popular in-the-wild dataset. We train our model on Human3.6M and test it on 3DPW to evaluate the generalization ability. Furthermore, a subset of 3DPW, *i.e.,* 3DPW-Occ [50], is used to validate the model's robustness against occlusions.

Evaluation Metrics. We calculate the Mean Per Joint Position Error (MPJPE) under two protocols: Protocol 1 measures MPJPE by aligning the root (pelvis) keypoint between predicted and ground truth poses, while Protocol 2 (PA-MPJPE) first aligns through translation, rotation, and scaling. We also follow [51] to use the 3DPW-AdvOcc protocol to evaluate performance under occlusion. Textured patches from the Describable Textures Dataset (DTD) [9] are applied to input images to assess degradation on visible joints and identify the worst prediction. Two patch sizes, Occ@40 and Occ@80, are used with a stride of 10.

Implementation Details. The encoder and decoder of

Table 1. Comparison of the proposed method with state-of-the-art methods on Human3.6M using MPJPE. The table is split into 3 groups. The top group uses SH detected [28] 2D poses, the middle group uses CPN detected [5] 2D poses, and the bottom group uses group truth 2D poses as inputs. † denotes using temporal context, ⋆ denotes using visual cues and ∗ denotes using hierarchical information. f denotes the number of frames used in temporal-based methods. Best in **bold**, second best in underlined.

Method	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
Learning [11]	50.1	54.3	57.0	57.1	66.6	73.3	53.4	55.7	72.8	88.6	60.3	57.7	62.7	47.5	50.6	60.4
SemGCN [52]⋆	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
Monocular [43]⋆	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
Graformer [53]	49.3	53.9	54.1	55.0	63.0	69.8	51.1	53.3	69.4	90.0	58.0	55.2	60.3	47.4	50.6	58.7
Lifting by Image [55]⋆	48.3	51.5	**46.1**	48.5	**53.7**	**42.8**	47.3	59.9	71.0	**51.6**	52.7	46.1	**39.8**	53.0	43.9	51.0
Ours∗	**46.9**	**50.4**	56.8	**47.1**	56.4	46.2	**42.3**	**49.2**	**68.7**	54.9	**51.9**	**43.1**	41.0	45.2	**40.1**	**49.3**
VideoPose [30]†($f = 243$)	45.1	47.4	42.0	46.0	49.1	56.7	44.5	44.4	57.2	66.1	47.5	44.8	49.2	32.6	34.0	47.1
GraphSH [42]	45.2	49.9	47.5	50.9	54.9	66.1	48.5	46.3	59.7	71.5	51.4	48.6	53.9	39.9	44.1	51.9
POT [23]	47.9	50.0	47.1	51.3	51.2	59.5	48.7	46.9	56.0	61.9	51.1	48.9	54.3	40.0	42.9	50.5
DiffPose [13]	42.8	49.1	45.2	48.7	52.1	63.5	46.3	45.2	58.6	66.3	50.4	47.6	52.0	37.6	40.2	49.7
Di^{2}Pose [40]	41.9	47.8	45.0	49.0	51.5	62.2	45.7	45.6	57.6	67.1	50.1	45.3	51.4	37.3	40.9	49.2
Lifting by Image [55]⋆	44.9	46.4	42.4	44.9	48.7	**40.1**	44.3	55.0	58.9	**47.1**	48.2	42.6	**36.9**	48.8	40.1	46.4
PoseFormer [54]†($f = 81$)	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 [25]†($f = 351$)	**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 [47]†($f = 81$)	39.8	43.0	38.6	**40.1**	43.4	50.6	40.6	41.4	52.2	56.7	43.8	40.8	43.9	**29.4**	**30.3**	42.4
Ours∗	42.8	**42.7**	**38.1**	41.3	**42.7**	46.3	**37.2**	44.2	**51.0**	51.4	**40.9**	**38.3**	40.0	39.9	33.7	**42.0**
SemGCN [52]⋆	37.8	49.4	37.6	40.9	45.1	41.4	40.1	48.3	50.1	42.2	53.5	44.3	40.5	47.3	39.0	43.8
VideoPose [30]†($f = 243$)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	37.2
Pose2Mesh [7]∗	38.1	41.7	38.3	37.5	39.2	45.4	37.5	36.2	45.7	50.1	39.8	39.2	40.2	35.2	37.6	40.1
HGN [22]∗	35.4	40.2	31.1	38.2	38.3	41.1	36.1	32.7	42.1	48.4	37.1	36.9	37.1	30.5	32.4	37.2
DiffPose [13]	28.8	32.7	27.8	30.9	32.8	38.9	32.2	28.3	33.3	41.0	31.0	32.1	31.5	25.9	27.5	31.6
Lifting by Image [55]⋆	29.5	30.1	25.0	29.0	28.5	**28.6**	**26.9**	30.5	31.1	**27.7**	32.4	27.7	**24.8**	30.0	25.9	28.6
PoseFormer [54]†($f = 81$)	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
MHFormer [25]†($f = 351$)	27.7	32.1	29.1	28.9	30.0	33.9	33.0	31.2	37.0	39.3	30.0	31.0	29.4	22.2	23.0	30.5
MixSTE [47]†($f = 81$)	**25.6**	**27.8**	**24.5**	**25.7**	**24.9**	29.9	28.6	**27.4**	**29.9**	29.0	**26.1**	**25.0**	25.2	**18.7**	**19.9**	**25.9**
Ours∗	30.4	29.7	26.3	27.2	28.7	29.1	28.2	29.2	30.9	33.1	29.6	26.2	27.2	21.9	26.2	28.3

Table 2. Comparison of our method with state-of-the-art methods on 3DPW, 3DPW-Occ, and 3DPW-AdvOcc using MPJPE (Protocol 1) and PA-MPJPE (Protocol 2). We use CPN detected [5] 2D poses as inputs. The 40 and 80 in 3DPWAdv-Occ represent the occluded size.

Method	3DPW		3DPW-Occ		3DPW-AdvOcc@40		3DPW-AdvOcc@80	
	MPJPE	PA-MPJPE	MPJPE	PA-MPJPE	MPJPE	PA-MPJPE	MPJPE	PA-MPJPE
STRGCN [1]	112.9	69.6	115.8	72.3	241.1	101.4	355.9	116.3
VideoPose [30]	101.8	63.0	106.7	67.1	221.6	99.4	334.3	112.9
GnTCN [6]	—	64.2	—	85.7	279.4	113.2	371.4	119.8
PoseFormer [54]	118.2	73.1	132.8	80.5	247.9	106.2	359.6	115.5
Learning [48]	91.1	54.3	94.6	56.7	142.5	83.5	251.8	103.9
PCT [12]	83.1	53.9	82.8	53.7	127.2	72.2	192.5	92.1
DiffPose [13]	82.7	53.8	82.1	53.5	121.4	70.9	189.3	92.4
Di^{2}Pose [40]	79.3	50.1	79.6	50.7	**108.4**	**59.8**	153.6	78.7
Ours	**77.2**	**48.8**	**75.4**	**47.3**	124.6	64.3	**143.2**	**72.1**

MSST comprise four and one MLP-Mixer [12] blocks with the embedding dimensions of 128 and 96, respectively. The joint quantities of the sparse, dense, and fine 2D pose are set as $J_s = 16$, $J_d = 48$, and $J_f = 96$. Both codebook sizes are 2048 with a dimension of 128. In HiARM, the LSAB, GCSAB, and PH consist of 1, 12, and 4 transformer blocks with the same dimension of 128. The lifting model is a 12-block vanilla spatial transformer with 6 heads and a dimension of 96. Detailed settings are in the Appendix C.

5.2. Comparison with State-of-the-art Methods

Results on Human3.6M. We evaluate our method on Human3.6M against three types of SOTA methods that incorporate additional information. Table 1 shows that our method achieves the best performance in the single-frame settings including the hierarchy-based (w/ ∗) and visual-based (w/ ⋆) methods. Moreover, our approach outperforms the temporal-based (w/ †) methods when using detected 2D poses, which is closely to the real-world scenarios. Table6 further demonstrates that hierarchical information can com-

Table 3. Ablation study on each component of MSST. H36M denotes the Huamn3.6 dataset.

Method	H36M	3DPW
Our method	**42.0**	**77.2**
w/o hierarchical codebooks	45.3△3.3	81.5△4.3
w/o Local Alignment	43.6△1.6	80.2△3.0
w/o Global Alignment	42.4△0.4	78.9△1.7

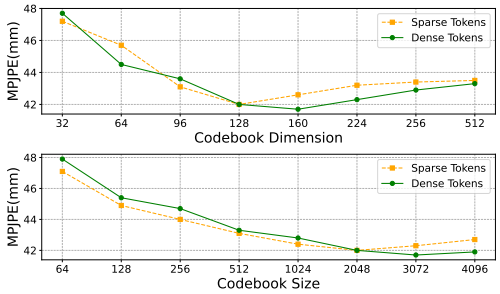


Figure 5. Impact of the codebook dimension (**top**) and size (**bottom**) for sparse and dense tokens on Humman3.6M.

plement temporal-based methods.

Results on 3DPW. We evaluate our model pretrained on Human3.6M to the 3DPW dataset. As shown in Table 2, our method achieves SOTA performance with an average improvement of 2.1mm in MPJPE and 1.3mm in PA-MPJPE over Di^{2}Pose [40], demonstrating strong generation ability. Our method also maintains superiority on 3DPW-Occ. Under the extreme occlusion scenarios of 3DPW-AdvOcc, our method shows a smaller performance drop (77% in MPJPE and 44% in PA-MPJPE) compared to the average drop of 120% and 67%, highlighting the superiority of hierarchical information in enhancing robustness under occlusions.

5.3. Ablation Studies and Analysis

We perform ablation studies on Human3.6M and 3DPW using CPN detected 2D poses to understand how our method enhances 3D HPE. More results are in the Appendix E.

Ablation Study on each Component of MSST. We verify each component of MSST by either removal or standard component replacement in Table 3. The accuracy on both datasets reduces sharply when replacing hierarchical codebooks with one shared codebook. It highlights the importance of hierarchical design in token learning. Then, we sequentially remove local and Global Alignment, which results in respective performance loss of 1.6mm and 0.4mm on Human3.6M, indicating that these alignments enhance discrete space learning and strengthen token connectivity.

Ablation Study on the Codebook Dimension and Size. We examine the effects of the codebook dimension and size on Human3.6M. Fig. 5 shows that increasing the sparse and dense codebook dimension from 32 to 128 leads to significant performance gains, but further increases introduce redundancy without additional improvement. Testing different codebook sizes reveals peak performance at 2048 for the sparse and 3072 for the dense. Larger sizes increase the learning complexity without notable performance gains.

Table 4. Ablation study on HiARM. "Both" denotes using both "Center-to-periphery" and "Sparse-to-dense" strategies. Frame per second (FPS) is based on one NVIDIA Tesla V100 GPU.

Framework Type	Auto-Regression Strategy	H36M	3DPW	Params(M)	FPS
Classification	None	46.3	81.9	2.2	432
ARM	Next-token	43.6↓2.7	80.2↓1.7	2.4	265
ARM	Center-to-periphery	42.5↓3.8	78.4↓3.5	2.4	265
ARM	Sparse-to-dense	43.1↓3.2	79.3↓2.6	2.4	396
ARM	Both	**42.0**↓4.3	**77.2**↓4.7	2.4	396

Table 5. Ablation study on the level of hierarchical input 2D poses.

48	96	192	H36M	Params(M)
			52.1	0.9
✓			46.7↓5.4	1.4
	✓		43.2↓8.9	1.8
		✓	42.7↓9.4	2.5
✓	✓		42.0↓10.1	2.4
✓	✓	✓	**41.4**↓10.7	4.4

Table 6. Ablation study on lifting models. We compare model complexity, inference speed, and MPJPE on Human3.6M.

Lifting Model	Params(M)	FPS	MPJPE
vanilla spatial transformer	0.9	458	52.1
vanilla spatial transformer w/ ours	2.0	396	42.0↓10.1
PoseFormer [54] ($f = 81$)	9.7	381	44.3
PoseFormer [54] ($f = 81$) w/ ours	10.7	185	40.8↓3.5
MixSTE [47] ($f = 81$)	33.7	1134	42.4
MixSTE [47] ($f = 81$) w/ ours	34.8	681	39.3↓3.1

Ablation Study on HiARM. We then conduct an ablation study on the framework type and auto-regression strategy of HiARM in Table 4. Following PCT [12], we reformulate pose estimation in Stage 2 as a classification task by predicting the categories of hierarchical tokens. When replacing it with an auto-regressive model (ARM) and employing a next-token prediction order, MPJPE reduces by 2.7mm, demonstrating that ARM better captures the token latent distribution. We further adopt different auto-regression strategies. "Center-to-periphery", "Sparse-to-dense" and their combination yield great performance improvements, reaching a final best MPJPE of 42.0mm. "Sparse-to-dense" speeds up the ARM inference process by 49%.

Ablation Study on the Effect of Different Levels of 2D Poses. We validate the effect of different levels of 2D poses by incorporating the original sparse 2D pose (17 joints) with additional 2D poses at three levels (48, 96, and 192 joints). Table 5 reveals that denser 2D pose inputs improve 3D HPE accuracy. Incorporating a 48-joint pose results in a 5.4mm gain (52.1mm to 46.7mm), and a 96-joint pose further improves 4.7mm (46.7mm to 42.0mm). However, adding a 192-joint pose only enhances MPJPE by 0.6mm but significantly increases the complexity. We analyze that the distinct positional information from the 48-joint and 96-joint poses together boosts performance, but the already rich context in these two levels lets the 192-joint pose bring minimal gains.

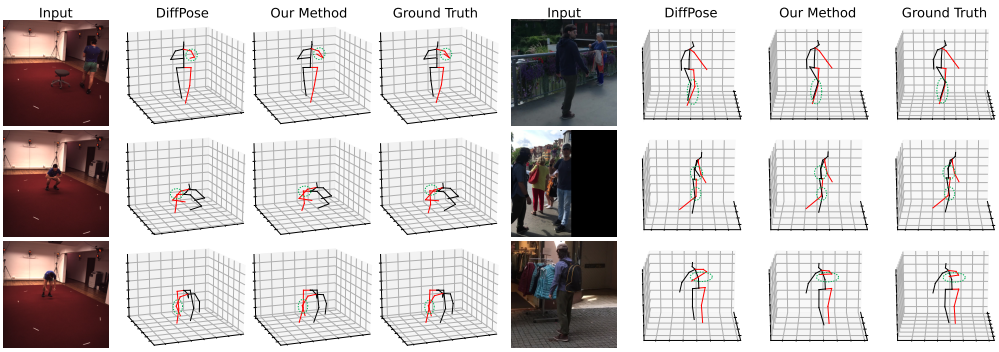
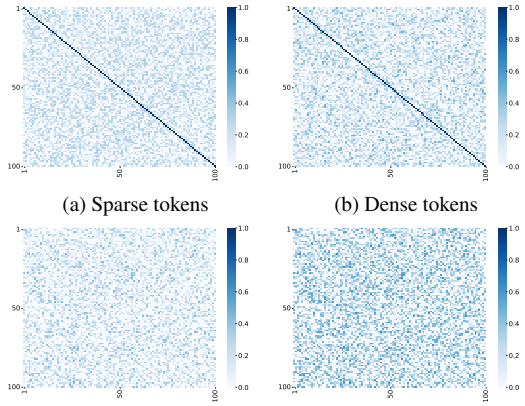


Figure 6. Qualitative results compared with DiffPose [13] on Human3.6M (**left**) and 3DPW (**right**).



(a) Sparse tokens (b) Dense tokens

(c) Sparse and dense tokens w/o local alignment (d) Sparse and dense tokens w/ local alignment

Figure 7. Hierarchical codebooks similarity analysis. The cosine similarity is calculated based on the first 100 tokens from both the sparse and dense codebooks. (c) (d) The x-axis represents sparse tokens, and the y-axis represents dense tokens.

Extension to Temporal-based Lifting Models. We further extend the lifting model to temporal-based methods including PoseFormer [54] and MixSTE [47]. Table 6 shows that the incorporation of hierarchical information leads to performance gains across various methods, achieving the SOTA result when combined with MixSTE with a tolerable computational complexity. More discussion about the extension to temporal-based methods is in the Appendix G.

We also conduct a series of qualitative analyses. Additional visualizations are available in the Appendix F.

Qualitative Results. Fig. 6 shows qualitative results on Human3.6M and 3DPW, comparing our method with DiffPose [13]. Our method provides accurate pose predictions, especially in heavily occluded scenarios, demonstrating that the hierarchical information helps address the depth ambiguity due to occlusion.

Analysis on the Similarity of Hierarchical Codebooks. We compute four cosine similarity matrices to evaluate the relationships among sparse tokens, dense tokens, and their interplay. Fig. 7(a) and (b) show that the connections within sparse and dense codebooks are generally low, indicating that each codebook contains relatively independent and non-redundant information. The slightly higher similarity among dense tokens compared to sparse tokens indicates that dense tokens capture more fine-grained, localized information, leading to stronger correlations. Fig. 7 (c) and (d) demonstrate that Local Alignment enhances the connection between sparse and dense tokens.

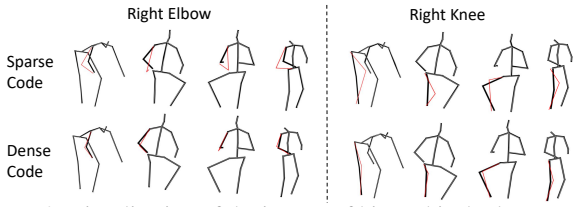


Figure 8. Visualization of the impact of hierarchical tokens on the sub-structure. We change the single sparse and dense token of two parts into another value, respectively, and the corresponding sub-structures consistently change (Highlighted in red). Black poses denote the original, while gray poses denote the change.

Impact of Hierarchical Tokens on the Sub-structure. Fig. 8 visualizes the influence of hierarchical tokens on sub-structures in Human3.6M using 16 sparse and 48 dense tokens. The pose sub-structure, represented by a pair of joints, is primarily controlled by two corresponding tokens. Modifying the sparse token leads to significant changes in the sub-structure, indicating that it captures coarse-grained information, while changes to the dense token result in minor adjustments, suggesting it focuses on fine-grained details.

6. Conclusion

This paper introduces the Hierarchical Pose AutoRegressive Transformer (HiPART), a novel generative densification approach that derives reliable hierarchical dense 2D poses from the sparse 2D pose. We employ a Multi-Scale Skeletal Tokenization module to progressively quantize a highly dense 2D pose into hierarchical tokens and establish a Hierarchical AutoRegressive Modeling scheme for the pose generation. As the first approach for hierarchical dense 2D pose generation tackling occlusions, HiPART achieves SOTA performance on the single-frame setting using only a vanilla spatial transformer for 2D-to-3D lifting and even outperforms numerous multi-frame methods.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 62125109, Grant 62431017, Grant U24A20251, Grant 62320106003, Grant 62371288, Grant 62401357, Grant 62401366, Grant 62301299, Grant 62120106007, and in part by the Program of Shanghai Science and Technology Innovation Project under Grant 24BC3200800.

References

[1] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3D pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2272–2281, 2019. 6

[2] Ching-Hang Chen, Ambrish Tyagi, Amit Agrawal, Dylan Drover, Rohith Mv, Stefan Stojanov, and James M Rehg. Unsupervised 3D pose estimation with geometric self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2019. 1, 2

[3] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1691–1703, 2020. 2, 4

[4] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. DiffusionDet: Diffusion model for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19830–19843, 2023. 3

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018. 1, 6

[6] Yu Cheng, Bo Wang, Bo Yang, and Robby T Tan. Graph and temporal convolutional networks for 3D multi-person pose estimation in monocular videos. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pages 1157–1165, 2021. 6

[7] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2Mesh: Graph convolutional network for 3D human pose and mesh recovery from a 2D human pose. In *Proceedings of the 16th European Conference on Computer Vision*, pages 769–787, 2020. 3, 6

[8] Jeongjun Choi, Dongseok Shim, and H Jin Kim. DiffuPose: Monocular 3D human pose estimation via denoising diffusion probabilistic model. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3773–3780, 2023. 3

[9] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014. 5

[10] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021. 2, 4

[11] Hao-Shu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. Learning pose grammar to encode human body configuration for 3D pose estimation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pages 6821–6828, 2018. 6

[12] Zigang Geng, Chunyu Wang, Yixuan Wei, Ze Liu, Houqiang Li, and Han Hu. Human pose as compositional tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 660–671, 2023. 3, 6, 7

[13] Jia Gong, Lin Geng Foo, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. DiffPose: Toward more reliable 3D pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13041–13051, 2023. 2, 3, 6, 8

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems 33*, pages 6840–6851, 2020. 3

[15] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022. 3

[16] Karl Holmquist and Bastian Wandt. Diffpose: Multi-hypothesis human pose estimation using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15977–15987, 2023. 3

[17] Rongjie Huang, Zhou Zhao, Huadai Liu, Jinglin Liu, Chenye Cui, and Yi Ren. ProDiff: Progressive fast diffusion model for high-quality text-to-speech. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2595–2605, 2022. 3

[18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2013. 2, 3, 4, 5

[19] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. PARE: Part attention regressor for 3D human body estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11127–11137, 2021. 3

[20] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2252–2261, 2019. 3

[21] Jogendra Nath Kundu, Siddharth Seth, MV Rahul, Mugalodi Rakesh, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Kinematic-structure-preserved representation for unsupervised 3D human pose estimation. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 11312–11319, 2020. 1, 2

[22] Han Li, Bowen Shi, Wenrui Dai, Yabo Chen, Botao Wang, Yu Sun, Min Guo, Chenlin Li, Junni Zou, and Hongkai

Xiong. Hierarchical graph networks for 3D human pose estimation. In *British Machine Vision Conference*, 2021. 3, 6

[23] Han Li, Bowen Shi, Wenrui Dai, Hongwei Zheng, Botao Wang, Yu Sun, Min Guo, Chenglin Li, Junni Zou, and Hongkai Xiong. Pose-oriented transformer with uncertainty-guided refinement for 2D-to-3D human pose estimation. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, pages 1296–1304, 2023. 4, 6

[24] Han Li, Shaohui Li, Wenrui Dai, Chenglin Li, Junni Zou, and Hongkai Xiong. Frequency-aware transformer for learned image compression. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[25] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. MHFormer: Multi-hypothesis transformer for 3D human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13147–13156, 2022. 2, 6

[26] Jun Liu, Henghui Ding, Amir Shahroudy, Ling-Yu Duan, Xudong Jiang, Gang Wang, and Alex C Kot. Feature boosting network for 3D pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):494–501, 2019. 2

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

[28] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Proceedings of the 14th European Conference on Computer Vision*, pages 483–499, 2016. 1, 6

[29] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *2018 International Conference on 3D Vision (3DV)*, pages 484–494, 2018. 3

[30] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 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*, pages 7753–7762, 2019. 2, 5, 6

[31] Ibrahim Radwan, Abhinav Dhall, and Roland Goecke. Monocular image 3D human pose estimation under self-occlusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1888–1895, 2013. 1, 2

[32] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *Advances in Neural Information Processing Systems 32*, pages 14866–14876, 2019. 2, 3

[33] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5693–5703, 2019. 1

[34] Zhenhua Tang, Zhaofan Qiu, Yanbin Hao, Richang Hong, and Ting Yao. 3D human pose estimation with spatio-temporal criss-cross attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4790–4799, 2023. 2

[35] Bugra Tekin, Pablo Márquez-Neila, Mathieu Salzmann, and Pascal Fua. Learning to fuse 2D and 3D image cues for monocular body pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3941–3950, 2017. 2

[36] Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In *Advances in Neural Information Processing Systems 37*, pages 84839–84865, 2024. 2

[37] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. MLP-Mixer: An all-MLP architecture for vision. In *Advances in Neural Information Processing Systems 34*, pages 24261–24272, 2021. 3

[38] Aaron Van Den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems 30*, pages 6309–6318, 2017. 2

[39] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using imus and a moving camera. In *Proceedings of the 15th European Conference on Computer Vision*, pages 601–617, 2018. 2, 5

[40] Weiquan Wang, Jun Xiao, Chunping Wang, Wei Liu, Zhao Wang, and Long Chen. Di2Pose: Discrete diffusion model for occluded 3D human pose estimation. In *Advances in Neural Information Processing Systems 37*, pages 98717–98741, 2024. 6, 7

[41] Jingwei Xu, Zhenbo Yu, Bingbing Ni, Jiancheng Yang, Xiaokang Yang, and Wenjun Zhang. Deep kinematics analysis for monocular 3D human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 899–908, 2020. 1, 2

[42] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3D human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16105–16114, 2021. 6

[43] 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*, 44 (10):6327–6344, 2021. 6

[44] Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. Diffsound: Discrete diffusion model for text-to-sound generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1720–1733, 2023. 3

[45] Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David A Ross, and Lu Jiang. Language model beats diffusion – Tokenizer is key to visual generation. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[46] Zhenbo Yu, Bingbing Ni, Jingwei Xu, Junjie Wang, Chenglong Zhao, and Wenjun Zhang. Towards alleviating the modeling ambiguity of unsupervised monocular 3D human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8651–8660, 2021. 1, 2

[47] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. 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*, pages 13232–13242, 2022. 6, 7, 8

[48] Jianfeng Zhang, Kehong Gong, Xinchao Wang, and Jiashi Feng. Learning to augment poses for 3D human pose estimation in images and videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10012–10026, 2023. 6

[49] Jinglei Zhang, Jiankang Deng, Chao Ma, and Rolandos Alexandros Potamias. HaWoR: World-space hand motion reconstruction from egocentric videos. *arXiv preprint arXiv:2501.02973*, 2025. 3

[50] Tianshu Zhang, Buzhen Huang, and Yangang Wang. Object-occluded human shape and pose estimation from a single color image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7376–7385, 2020. 2, 5

[51] Yi Zhang, Pengliang Ji, Angtian Wang, Jieru Mei, Adam Kortylewski, and Alan Yuille. 3D-aware neural body fitting for occlusion robust 3D human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9399–9410, 2023. 5

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

[53] Weixi Zhao, Weiqiang Wang, and Yunjie Tian. Graformer: Graph-oriented transformer for 3D pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20438–20447, 2022. 2, 5, 6

[54] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3D human pose estimation with spatial and temporal transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11656–11665, 2021. 2, 6, 7, 8

[55] Feng Zhou, Jianqin Yin, and Peiyang Li. Lifting by image–leveraging image cues for accurate 3D human pose estimation. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pages 7632–7640, 2024. 2, 6