Information Fusion 130 (2026) 104100

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full Length Article

DAK-Pose: Dual-augmentor knowledge fusion for generalizable video-based 3D human pose estimation

Yachuan Wang, Bin Zhang *, Hao Yuan

Xi'an Jiaotong University, Xi'an, Shaanxi, 710049, China

ARTICLE INFO

Keywords:
Human pose estimation
Domain generalization
Feature disentanglement

ABSTRACT

Real-world deployment of video-based 3D human pose estimation remains challenging, as limited annotated data collected in constrained lab settings cannot fully capture the complexity of human motion. While motion synthesis for data augmentation has emerged as a mainstream solution to enhance generalization, existing synthesis methods suffer from inherent trade-offs: kinematics-based motion synthesis approaches preserve anatomical plausibility but sacrifice temporal coherence, while coordinate-based methods ensure motion smoothness but violate biomechanical constraints. This results in persistent domain gaps when synthetic data is directly used in the observation space to train pose estimation models. To overcome this, we propose DAK-Pose, which shifts augmentation to the feature space. We disentangle motion into structural and dynamic features, and design two complementary augmentors: (1) A structure-prioritized module enforces kinematic constraints for anatomical validity, and (2) a dynamic-prioritized module generates diverse temporal patterns. Auxiliary encoders trained on synthetic motions generated by these augmentors transfer domain-invariant knowledge to the pose estimator through adversarial alignment. Experiments on Human3.6M, MPI-INF-3DHP, and 3DPW datasets show that DAK-Pose achieves state-of-the-art cross-dataset performance.

1. Introduction

3D human pose estimation (3DHPE) aims to recover the spatial positions of human joints from 2D observations, and has attracted increasing attention in recent years due to its wide applications in motion capture, human-computer interaction, and virtual reality. However, a major challenge in this field lies in the nature of the training data, which is typically collected in controlled laboratory environments using professional actors and wearable sensors. Such constrained acquisition settings often lead to limited data diversity, resulting in a significant drop in generalization performance when models are applied to real-world scenarios.

To mitigate the distribution gap between training data and test scenarios, data augmentation has emerged as a mainstream solution. The core idea is to enhance the generalization capability of models by increasing the diversity of training samples. A recent trend focuses on pose synthesis as a form of data augmentation, aiming to reduce the domain gap between synthetic and real data by generating more realistic human poses, thereby improving model performance in real-world applications. Existing pose synthesis methods generally follow two technical paradigms: kinematics-based approaches and coordinate-based pose generation. However, both suffer from inherent limitations that pose fundamental challenges in the feature space. Kinematics-based methods [1] operate under the constraints of the Lie group for joint rotations, which ensures the physical plausibility of skeletal structures but often compromises the temporal coherence and naturalness of motion trajectories. In contrast, coordinate-based methods [2] generate joint positions independently, which helps preserve temporal continuity, but neglecting the spatial dependencies between joints leads to violations of the spatiotemporal constraints inherent in the human kinematic chain. These limitations lead to a common domain gap between the synthetic and real domains in the generated 2D-3D paired training data. Specifically, synthetic and real data exhibit systematic distribution discrepancies in terms of motion patterns and error accumulation. When synthetic data is directly used for training in the observation space, it becomes difficult to effectively improve the model's generalization to real-world scenarios.

To address this issue, we propose DAK-Pose, a domain generalization framework based on disentangled representation learning. By learning domain-invariant representations in the feature space, our framework effectively leverages the advantageous characteristics of synthetic motion while suppressing domain-specific biases, thereby enhancing the model's generalization across domains, as shown in Fig. 1. Specifically,

* Corresponding author .
E-mail addresses: smxwych@stu.edu.xjtu.cn (Y. Wang), bzhang82@xjtu.edu.cn (B. Zhang), su_xi_xjt@stu.xjtu.edu.cn (H. Yuan).

https://doi.org/10.1016/j.inffus.2025.104100
Received 2 June 2025; Received in revised form 16 November 2025; Accepted 23 December 2025
Available online 3 January 2026

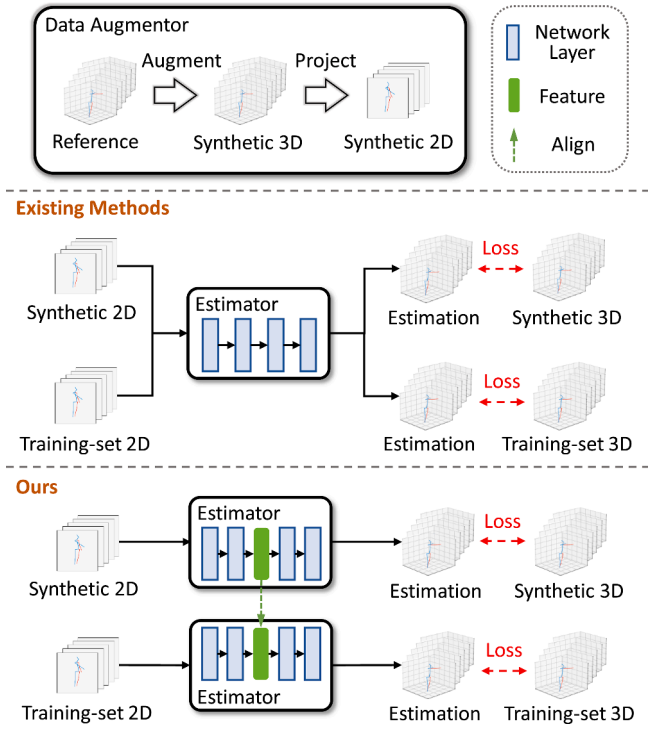

Fig. 1. Training strategies comparison. Unlike observation-space augmentation that directly adds synthetic motions to training, our method performs domain generalization in the feature space via auxiliary estimators, allowing the estimator to benefit from the informative aspects of synthetic motions while avoiding their potential noise.

prior approaches typically model human pose as a single unified representation, neglecting the fact that motion dynamics (e.g., trajectories and velocities) and structural attributes (e.g., bone lengths and joint configurations) are physically disentangled. We explicitly decompose motion into structural and dynamic features, and model them separately to allow generalize each component independently. By operating in the feature space, our method avoids the intrinsic trade-off of observation-space augmentation-namely, the balance between increased data diversity and the risk of introducing noise. This allows us to fully leverage the strengths of different types of synthetic data while mitigating their drawbacks.

To enhance the diversity of limb structures while mitigating the physical distortion of skeletal configurations in synthetic data, we design a human motion synthesis module based on kinematic chain constraints. To further amplify the skeletal consistency characteristic of kinematics-constrained synthetic motions and to suppress dynamic distortions caused by the simulation environment, we focus on the disentangled structural features. Feature alignment techniques are then applied to enhance the generalization ability of the pose estimation network toward these structural features. For dynamic features, we employ a random motion predictor to synthesize diverse motion patterns, thereby improving the variability of motion dynamics. Specifically, 3D synthetic motions with different dynamic patterns are generated based on 3D reference motions, aiming to perform domain generalization for motion dynamics and strengthen the model's capacity to model diverse dynamic characteristics. Moreover, the synthetic motions under this mode inherently retain advantages such as temporal continuity and motion smoothness. At the same time, we intentionally discard the structural features from such synthetic motions to avoid the drawbacks arising from their weak intra-frame joint correlations. Finally, through feature alignment, we effectively improve the generalization ability of the pose estimation network with respect to dynamic features.

The contributions of this work can be summarized as follows:

- We propose DAK-Pose, A novel feature-space augmentation framework that fundamentally shifts the paradigm of synthetic data utilization from observation space to latent feature space.
- We introduce two complementary augmentors for structure-prioritized and dynamics-prioritized motion synthesis, and perform adversarial alignment to fuse disentangled features into the pose estimator.
- Extensive experiments across multiple datasets demonstrate that our method achieves state-of-the-art performance on cross-domain 3D human pose estimation.

2. Related works

2.1. 3D human pose estimation

With the rapid advancement of deep learning techniques, 3DHPE has made significant progress. This work focuses on skeleton-based 3DHPE, where the input 2D poses are typically obtained from image-based pose estimators [3,4] or manual annotations. In addition to single-frame approaches [5,6], video-based methods [7,8] have also gained increasing attention due to their ability to model temporal dynamics. However, most existing methods still rely heavily on fully supervised training paradigms, which require high-quality labeled datasets. As a result, their performance tends to degrade when deployed in unconstrained real-world scenarios, where unfamiliar body shapes or unseen motion patterns often cause domain shift. To address the limitations imposed by the scarcity and homogeneity of training data, recent research has explored several alternative learning paradigms. Self-supervised methods [9,10] leverage large-scale in-the-wild unlabeled 2D data to enhance model generalization, while weakly supervised approaches [11,12] enable training with unpaired or partially labeled data. In this work, we focus on improving generalization by generating additional paired data through data augmentation.

2.2. Data augmentation for 3D human pose estimation

In cross-domain 3D human pose estimation methods, data augmentation techniques are employed to enhance data diversity and simulate potential domain discrepancies. We categorize existing data augmentation-based works into two groups: meta-augmentation and generative augmentation. Meta-augmentation methods [13–15] focus on designing augmentation policies, typically by applying a series of basic transformations to existing data to produce augmented samples. Zhang et al. [14] utilized differentiable pose transformations to generate diverse training samples. Peng et al. [15] introduced a dual-branch framework with weak and strong augmenters and employed meta-optimization to simulate cross-domain gaps. In contrast, generative augmentation methods [16–18] aim to learn a generative model capable of producing high-quality samples, which are then used to form a large-scale synthetic dataset for training downstream models. Liu et al. [17] adopted a diffusion-based architecture to simulate 3D poses from the target domain and adapted them via a teacher-student framework with low-rank constraints.

Existing data augmentation approaches for pose estimation, including both meta-augmentation and generative augmentation, typically operate in the observation space. Due to inherent trade-offs in motion synthesis, these methods often yield distorted samples. Although such augmentation can passively improve model robustness, it may also cause the model to deviate from the true data manifold. In comparison, our feature-space augmentation method offers two distinct advantages. First, by leveraging feature disentanglement, it reduces the need for manual data tuning and remains effective even with imperfect samples. Second, as high-dimensional features encapsulate richer semantics, perturbations applied in this space are more likely to remain within the real data distribution. As a result, our approach provides a more general and secure augmentation strategy for learning robust pose estimation.

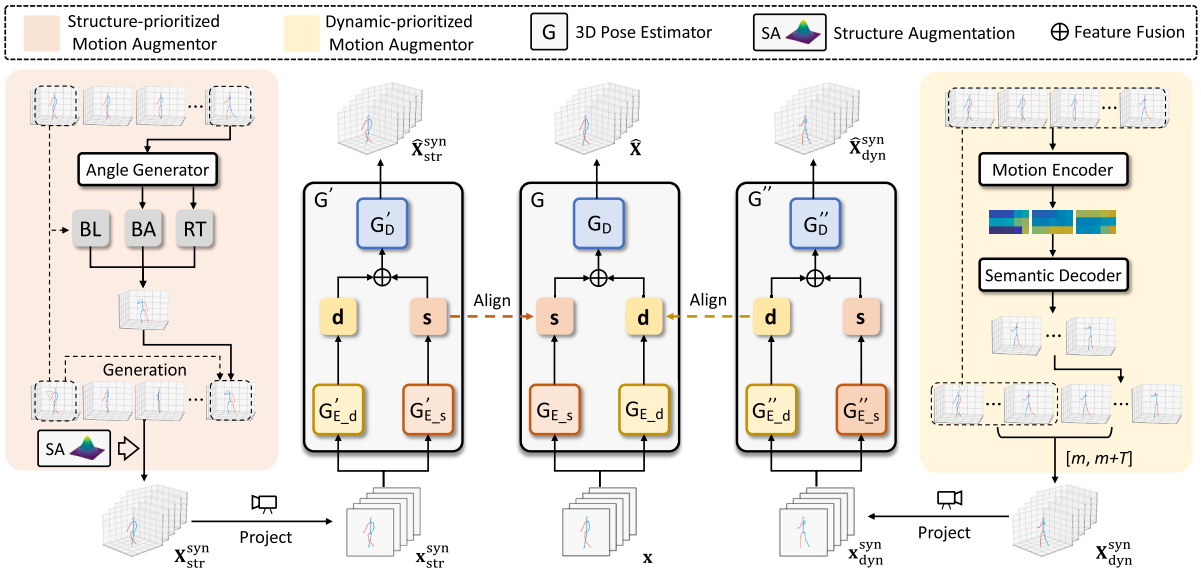

Fig. 2. Overview of DAK-Pose. Two motion augmentors generate synthetic motions from a reference motion. The structure-prioritized motion augmentor (SMA) produces motions to train G', while the dynamic-prioritized motion augmentor (DMA) trains G''. Structural features from G' and dynamic features from G'' are aligned with those in the main estimator G, enabling it to learn complementary strengths from both synthetic motions. "BL", "BA" and "RT" denote "bone length", "bone angle" and "rotation and translation".

2.3. Human motion feature disentanglement

The goal of disentanglement is to identify independent and interpretable latent features within samples, enhancing model interpretability and controllability [19,20]. Several works on human motion modeling [21,22] disentangle and separately model human pose and shape. Some motion retargeting approaches [23,24] define motion in terms of structure, view, and motion, leveraging feature invariance for disentanglement. Remelli et al. [25] disentangled camera viewpoints in the pose estimation process. Honari et al. [26] separated time-varying and time-invariant components of motion. He et al. [27] introduced Decoupled Space-Time Aggregation to disentangle human motion in videos.

3. Method

3.1. Overview

The goal of 3D human pose estimation in videos is to recover a 3D pose sequence $\mathbf{X} = [X_1, X_2, \dots, X_T]$ ($X_t \in \mathbb{R}^{J\times 3}$ represents the t-th frame pose, J denotes the number of joints) from a given 2D pose sequence $\mathbf{x} = [x_1, x_2, \dots, x_T]$ ($x_t \in \mathbb{R}^{2\times J}$). Traditional fully supervised methods learn a pose estimator G by optimizing over paired 2D-3D annotations. To address the cross-dataset challenge, our objective is to improve the generalization ability of the pose estimator G by synthesizing diverse 2D-3D paired data for data generalization.

As shown in Fig. 2, we propose DAK-Pose for robust video-based 3D human pose estimation by leveraging a synergistic mechanism of feature disentanglement and domain generalization. Within this framework, the pose estimator comprises two parameter-independent encoders: a dynamic encoder G_{E_d} and a structural encoder G_{E_s}, which extract dynamic features $\mathbf{d}$ and structural features $\mathbf{s}$, respectively, from the input 2D pose sequence. A decoder G_D then reconstructs the 3D pose sequence $\hat{\mathbf{X}}$ from the disentangled features. During training, we introduce a dynamic feature augmentation module and a structural feature augmentation module to generate augmented samples that preserve temporal continuity and anatomical plausibility. These two types of synthesized motion data are used to train two auxiliary pose estimators G' and G'', respectively. Finally, domain adversarial training is employed to enable the pose estimator G to inherit the complementary advantages of both types of synthetic data while suppressing their domain-specific noise. This design not only constructs diverse and physically plausible motion data but also facilitates effective feature space alignment between synthetic and real data, thereby significantly improving the model's generalization ability in cross-dataset scenarios.

3.2. Feature disentanglement

In DAK-Pose, we define motion as two independent features: structural features, which primarily characterize limb configurations, and dynamic features, which capture motion patterns. Inspired by Yang et al. [23], Zhu et al. [24], we adopt a disentanglement paradigm based on separate modeling and cross-reconstruction to disentangle motion features. Building upon this paradigm, we propose a structure-dynamic feature disentanglement method tailored for human motions in real-world scenarios. Specifically, our approach consists of two components: structural augmentation and cross reconstruction, as illustrated in Fig. 3.

3.2.1. Structure augmentation

Our structural augmentation process consists of two steps: limb scaling and joint perturbation. The limb scaling technique was originally proposed in Yang et al. [23] to disentangle motion features for virtual characters. However, in our task, the human body in the real world is subject to biological constraints, where limb proportions and bone lengths typically follow a normal distribution. Therefore, we adopt random scaling factors sampled from a right-truncated normal distribution to ensure that each group of limbs is symmetrically scaled in both enlargement and reduction. The resulting limb-scaled motion is assumed to share the same motion dynamics as the original sequence but differs in structural configuration.

To further reduce domain shifts potentially introduced by annotation biases or 2D pose estimators across different datasets, we introduce a joint-relative perturbation strategy. This augmentation targets the relative positions of joints that are independent of temporal dynamics. To preserve structural plausibility and reduce the complexity of generalization, we focus on perturbing the neck and hip joints, which are more susceptible to domain shifts between the upper and lower body. Specifically, these joints are randomly displaced along the direction of the bone closer to the root joint, within a predefined range.

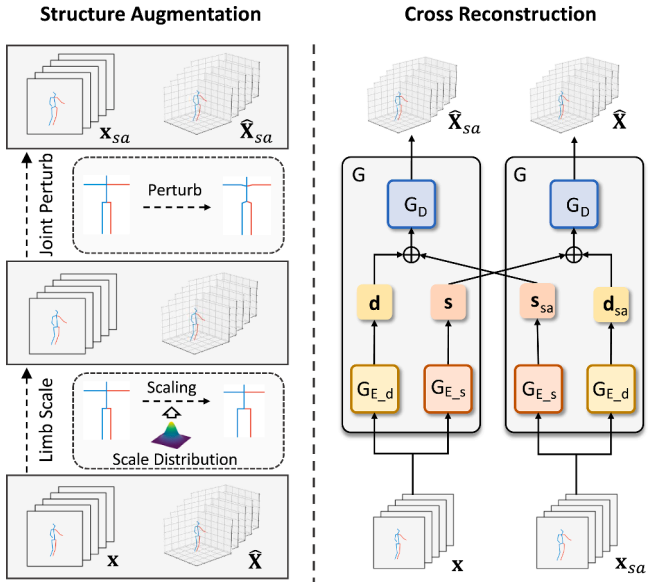

Fig. 3. Feature disentanglement. Through the structure augmentation, the original motions $\mathbf{x}$ and $\mathbf{X}$ are transformed into their structure-augmented versions $\mathbf{x}_{sa}$ and $\mathbf{X}_{sa}$. In the cross reconsturction pipeline, the encoded structure features $\mathbf{s}$ are cross-connected to guide the generation of 3D motions corresponding to $\mathbf{s}$.

3.2.2. Cross reconstruction

Our limb-scaling technique enables the generation of motion sequences that share identical dynamic characteristics while varying in skeletal structure, thereby facilitating effective structural feature disentanglement through motion dynamics preservation. Specifically, we simultaneously encode both the original input $\mathbf{x}$ and its structure-augmented version $\mathbf{x}_{sa}$ to obtain their respective dynamic feature $\mathbf{d}$ and structural features $\mathbf{s}$. Through cross-reconstruction of these four feature groups: $\mathbf{s}$, $\mathbf{s}_{sa}$, $\mathbf{d}$, $\mathbf{d}_{sa}$ with the constraint that reconstructed outputs maintain motion patterns consistent with their corresponding $\mathbf{s}$ features, we achieve dynamic feature stabilization while enhancing structural feature manipulability:

$$\hat{\mathbf{X}} = D(\mathbf{s}, \mathbf{d}_{sa}), \tag{1}$$

$$\hat{\mathbf{X}}_{sa} = D(\mathbf{s}_{sa}, \mathbf{d}). \tag{2}$$

The objective function for feature disentanglement is as follows:

$$\mathcal{L}_{fd} = \frac{1}{JT}(\| \hat{\mathbf{X}} - \mathbf{X} \|_2 + \| \hat{\mathbf{X}}_{sa} - \mathbf{X}_{sa} \|_2), \tag{3}$$

where $\mathbf{X}_{sa}$ denotes the pose sequence obtained by applying the same limb-scaling factors to $\mathbf{X}$, serving as the 3D paired data of x_{sa}.

3.3. Domain generalization

Instead of relying on the conventional fully supervised training paradigm that explicitly utilizes synthetic motion sequences in cross-dataset 3D human pose estimation, DAK-Pose leverages the specific characteristics of synthetic motion at the feature level, and adaptively integrates them into the pose estimation network. This approach effectively reduces the domain gap between synthetic and real data while enhancing the diversity of training samples in a more principled manner. In this section, we provide a detailed description of how the structural and dynamic features of motion are constructed and how these disentangled representations are adapted into the pose estimation network.

3.3.1. Structure-prioritized motion augmentor (SMA)

Since the pose estimation model outputs coordinate-independent 3D poses, the results are unconstrained by physical laws, leading to issues such as joint proportion imbalance, left-right asymmetry, and low temporal consistency between frames. For 3D motion augmentation, we adopt a two-stage process for human pose sequences. The first stage enhances the final frame of the reference sequence, and the second stage uses the first frame of the reference sequence as the starting frame and the enhanced pose as the ending frame to generate interpolated intermediate frames. In the first stage, inspired by Gong et al. [13], Peng et al. [15], Li and Pun [16], we construct internal body bone angles and bone lengths, and obtain the final augmented pose through global rotation and translation. Unlike existing methods, we replace the direct estimation of bone lengths with random values of limb distribution bone lengths in the limb-scale. This approach has three advantages: (1) it enhances the diversity of the augmented synthetic sequence structure, (2) it is compatible with our disentangling method, and (3) it reduces network complexity. In the second stage, we use an interpolation method similar to PoseAug-V [14], generating motion represented by rotational angles through a motion generation network, and employing a discriminator to ensure its validity.

3.3.2. Dynamic-prioritized motion augmentor (DMA)

We propose a motion prediction-guided dynamic feature construction module to generate pose sequences that capture the diversity and naturalness of human motion. This module consists of a motion prediction network and a pose estimation network, jointly optimized to form a closed-loop pipeline for dynamic feature generation, alignment, and constraint. The motion predictor in the augmentor follows a similar structure to Xu et al. [28]. Given a reference 3D motion $\mathbf{X}_{ref}$ and predicts future motion $\mathbf{X}_{pred}$. The encoder extracts semantic representations from $\mathbf{X}_{ref}$, capturing diverse motion trends. These semantics are decoded into multiple plausible future motions. However, due to the well-known drift and degradation issues in long-term motion prediction [29], directly using full-length predictions may adversely affect training. To mitigate this, we concatenate the reference motion $\mathbf{X}_{ref}$ and its prediction $\mathbf{X}_{pred}$ to form a complete sequence $\mathbf{X}_{com}$. We then randomly select a clip of length T starting from a time step m to construct the final synthesized dynamic sequence $\mathbf{X}_{syn\_dyn} = \mathbf{X}_{com}[m : m+T]$.

3.3.3. Adversarial fusion from dual auxiliary estimators

Following the synthesis of structure-prior and dynamics-prior motions, domain-adversarial training is applied to drive the pose estimation network to learn domain-invariant feature representations, effectively narrowing the distribution gap between synthetic and real-world data. The augmented sequence $\mathbf{X}_{str}^{syn}$ and $\mathbf{X}_{dyn}^{syn}$ is projected into $\mathbf{x}_{str}^{syn}$ and $\mathbf{x}_{dyn}^{syn}$ to obtain 2D-3D paired motions, which can be used to supervise the training of a 2D-to-3D estimator G' and G''. For the subsequent feature alignment process, the estimator G' and G'' shares the same architecture as G. During the feature alignment process, we employ two separate domain discriminators with independent parameters for structure and dynamics features, denoted as D_{str} and D_{dyn}, respectively.

The dynamic discriminator D_{dyn} has a structure similar to the dynamic encoder G_{E_d}, employing a lightweight Transformer to capture domain discrepancies along the temporal dimension. The dynamic feature map output by the dynamic encoder in the pose estimation network has a shape of $\mathbf{d} \in \mathbb{R}^{T \times C_{dyn}}$, where T denotes the number of time steps and C_{dyn} denotes the feature dimension. The structural discriminator D_{str} adopts the same architecture as G_{E_s}. The structural encoder produces a feature map of size $\mathbf{s} \in \mathbb{R}^{T \times C_{str}}$ is the number of channels. Unlike the encoding process in pose estimation, during the adversarial domain alignment for structural features, we fuse the temporal features to eliminate inconsistencies along the time axis that may affect the discriminator's loss calculation. As a result, the output feature map of the structural discriminator is reshaped to $\mathbf{d} \in \mathbb{R}^{1 \times C_{str}}$. The domain adversarial losses for structure and dynamic features can be respectively formulated as:

$$\begin{aligned}\mathcal{L}_{adv\_str} = &\mathbb{E}\left[D_{str}\left(G'_{E\_s}\left(\mathbf{x}_{str}^{syn}\right)\right)^2\right] + \\ &\mathbb{E}\left[\left(1 - D_{str}\left(G_{E\_s}(\mathbf{x})\right)\right)^2\right],\end{aligned} \tag{4}$$

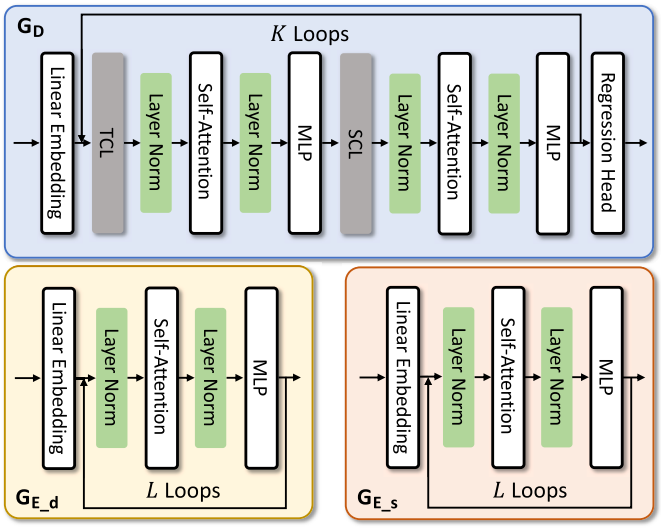

Fig. 4. Network structure. TCL/SCL denote Temporal/Spatial Correlation Learning.

$$\mathcal{L}_{adv\_dyn} = \mathbb{E}\left[D_{dyn}\left(G''_{E\_d}\left(\mathbf{x}^{syn}_{dyn}\right)\right)^2\right] + \mathbb{E}\left[\left(1 - D_{dyn}\left(G_{E\_d}(\mathbf{x})\right)\right)^2\right]. \tag{5}$$

The domain generalization loss can be expressed as:

$$\mathcal{L}_{dg} = \lambda_{str}\mathcal{L}_{adv\_str} + \lambda_{dyn}\mathcal{L}_{adv\_dyn}. \tag{6}$$

To align the features between synthetic and real data, we introduce a Gradient Reversal Layer (GRL) [30] in both the dynamic and structural feature streams. Specifically, GRL are integrated after the structure encoder and the motion encoder, respectively. During forward propagation, the GRL acts as an identity mapping, while during backpropagation, it multiplies the gradient of the domain discrimination loss by a negative weighting coefficient $-\lambda_{grl}$. To enhance the stability of adversarial training, the training process follows the classic linear warm-up strategy in the field of domain adaptation [31], where the weighting coefficient increases linearly from 0 to λ_{grl} during the first n_w epochs.

3.4. Network structure and loss function

3.4.1. Network structure

For the encoder in the pose estimation network G, we adopt a simple design consisting of cascaded Transformer blocks. As shown in Fig. 4, the dynamic encoder G_{E_d} and the structural encoder G_{E_s} share the similar architecture: both map inputs to the same feature dimension via a linear embedding layer and repeat L Transformer blocks. The key difference lies in the positional encoding strategy-while the dynamic encoder applies temporal positional encoding, the structural encoder adopts spatial positional encoding. The decoder G_D employs a cascaded Transformer architecture with temporally and spatially alternating blocks, similar to Zhang et al. [7], and is repeated for K iterations. The final 3D pose is predicted through a regression head. Within the decoder, temporal and spatial self-attention are independently implemented through dimension-wise transformations referred to as Temporal Correlation Learning (TCL) and Spatial Correlation Learning (SCL), both of which are widely used in video-based human pose modeling [7,8].

3.4.2. Loss function

The fundamental loss of DAK-Pose originates from the pose estimation losses of the three pose estimation networks within the framework, and can be formulated as:

$$\begin{aligned}\mathcal{L}_{pe} = \frac{1}{JT}(&\| \mathbf{X} - D(G_{E\_s}(\mathbf{x}), G_{E\_d}(\mathbf{x})) \|_2 + \\ &\| \mathbf{X} - D(G'_{E\_s}(x), G'_{E\_s}(x)) \|_2 + \\ &\| \mathbf{X} - D(G''_{E\_s}(x), G''_{E\_d}(x)) \|_2).\end{aligned} \tag{7}$$

The loss function used to train the pose estimation network G consists of three components: the pose estimation loss, the feature disentanglement loss, and the domain adversarial loss. The overall loss can be formulated as:

$$\mathcal{L}_{total} = \lambda_{pe}\mathcal{L}_{pe} + \lambda_{fd}\mathcal{L}_{fd} + \lambda_{dg}\mathcal{L}_{dg}. \tag{8}$$

4. Experiments

4.1. Datasets

Human3.6M [32] is a large-scale dataset for 3D human pose estimation, comprising 3.6 million 3D pose annotations and their corresponding images. It covers 15 categories of daily activities (e.g., eating, smoking, taking photos, walking, etc.), with data captured from professional actors performing predefined actions. We follow the standard data split protocol: data from five subjects (S1, S5, S6, S7, S8) are used for training, and data from two subjects (S9, S11) are used for testing. To further evaluate the cross-subject generalization ability of the pose estimation model, we follow prior work [17,33] and train the model using only subject S1, while evaluating on subjects S5, S6, S7, and S8. We adopt two standard evaluation protocols: (1) Protocol #1 (MPJPE): computes the Mean Per Joint Position Error (MPJPE), which is the average Euclidean distance between the predicted 3D joint positions and the ground truth. (2) Protocol #2 (Procrustes-MPJPE): computes the MPJPE after applying a rigid alignment using Procrustes analysis, thereby removing the effects of global rotation, translation, and scaling.

MPI-INF-3DHP [34] is another widely used benchmark for 3D human pose estimation, consisting of 1.3 million frames of multi-view video sequences. The data was captured in a professional studio environment using a commercial markerless motion capture system. It features eight subjects performing various actions across both indoor and outdoor scenes. We use this dataset for cross-dataset evaluation and adopt three standard metrics: Percentage of Correct Keypoints (PCK), Area Under the Curve (AUC), and MPJPE.

3DPW [35] is a challenging 3D human pose benchmark collected in unconstrained real-world environments. It contains 60 video sequences with annotated 2D poses and high-precision 3D poses obtained through Video-IMU fusion, maintaining accuracy even under complex outdoor conditions. Additionally, 3DPW provides diverse 3D body scans and parameterized human models under various clothing conditions. We use 3DPW for cross-dataset evaluation, employing the same evaluation metrics as in Human3.6M.

4.2. Setup

In our experiments, we set the loop number L of the encoder to 4, the decoder loop number K to 6, and the feature dimension C to 512 (detailed analysis is provided in Section 4.5.5). In the Human3.6M dataset, the input sequence length T is set to 27 and 81, while it is set to 27 for both MPI-INF-3DHP and 3DPW. For the SMA, the generator adopts a composite structure consisting of two sub-generators, namely the bone angle (BA) generator and the rotation and translation (RT) generator, as used in Gong et al. [13], Peng et al. [15]. During the motion sequence generation, we use the conditional generator introduced in Zhang et al. [14]. For DMA, the length m of the predicted sequence is set to 27 for $T = 27$ and 40 for $T = 81$ (detailed analysis see Section 4.5.3). The limb scaling process consists of two components: independent scaling of each limb group (Local Scaling) and global scaling applied uniformly to all limbs (Global Scaling). The corresponding scaling factors

follow normal distributions, denoted as $F_l \sim \mathcal{N}(1, 0.2)$ for local scaling and $F_g \sim \mathcal{N}(1, 0.3)$ for global scaling (detailed analysis can be found in Section 4.5.4). We set $\lambda_{str} = 1$, $\lambda_{dyn} = 2$ in Eq. (6) and $\lambda_{pe} = 10$, $\lambda_{fd} = 4$, $\lambda_{dg} = 4$ in Eq. (8). The parameters for domain-adversarial training were configured as $\lambda_{grl} = 1$ and $n_w = 50$.

Our implementation is based on the PyTorch framework and conducted on an Ubuntu server with Nvidia RTX 4090 GPUs. We train the model using the Adam optimizer with a batch size of 64. The initial learning rate is set to 0.00005 and decays by a factor of 0.99 after each epoch. For in-the-wild pose estimation, 2D poses are extracted from RGB images using the CPN [3] pose detector.

4.3. Quantitative results

4.3.1. Results on Human3.6M

As shown in Tables 1 and 2, our method outperforms all baseline approaches. Notably, it achieves superior performance at $T = 27$ compared to most baseline methods operating at $T = 81$, indicating that our approach is effective even with shorter input sequences. The results in Table 1 further demonstrate that our method generalizes well under limited training data, exhibiting strong cross-domain performance. Moreover, the model trained on poses captured from raw images (scenario CPN in Table 1) shows promising results, suggesting that our framework is applicable to in-the-wild scenarios. The significant improvement over single-frame methods confirms the effectiveness of our temporal modeling in capturing motion dynamics.

Table 1
Results on Human3.6M dataset (S1). *Source:* S1. Target: S5–S8. "CPN": trained on the detection by CPN [3]. "GT": trained on the groundtruth. "CD": cross-domain-based method; "2D": target 2D data used during training. "*": single-frame method.

Scenario			CPN		GT	
Method	CD	2D	MPJPE	P-MPJPE	MPJPE	P-MPJPE
MixSTE [7] ($T = 81$)			57.8	41.3	48.3	34.9
D3DP [36] ($T = 81$)			56.3	41.2	46.0	34.2
DDHPose [37] ($T = 81$)			56.0	41.3	45.1	34.5
KTPFormer [8] ($T = 81$)			54.6	39.1	43.8	34.0
PoseAug* [13]	✓		–	–	56.7	44.5
PoseDA* [38]	✓	✓	–	–	49.9	34.2
CEE-Net* [16]	✓		75.2	59.6	51.9	41.3
DAF-DG* [15]	✓		68.2	–	50.3	-
DH-AUG [39] ($T = 81$)	✓		57.0	40.2	49.5	36.6
AdaptPose [33] ($T = 81$)	✓	✓	55.8	39.8	47.0	34.6
PoseAug-V [14] ($T = 81$)	✓		55.3	40.5	46.1	36.2
PoSynDA [17] ($T = 27$)	✓	✓	57.2	44.0	45.3	34.9
PoSynDA [17] ($T = 81$)	✓	✓	55.7	43.3	43.1	33.8
BDC [18] ($T = 81$)	✓		56.4	44.1	44.6	34.7
Ours ($T = 27$)	✓		55.1	40.0	44.3	34.5
Ours ($T = 81$)	✓		**53.9**	**38.6**	**42.4**	**33.5**

Table 2
Results on Human3.6M dataset (Full). *Source:* S1, S5-S8. Target: S9, S11.

Method	2D	MPJPE	P-MPJPE
DH-AUG [39] ($T = 81$)		30.4	23.5
PoseAug-V [14] ($T = 81$)		25.7	21.6
PoSynDA ($T = 27$) [17]	✓	26.7	22.0
PoSynDA ($T = 81$) [17]	✓	25.3	**21.3**
BDC [18] ($T = 27$)		27.1	22.7
BDC [18] ($T = 81$)		25.4	21.5
Ours ($T = 27$)		26.3	22.5
Ours ($T = 81$)		**24.8**	**21.3**

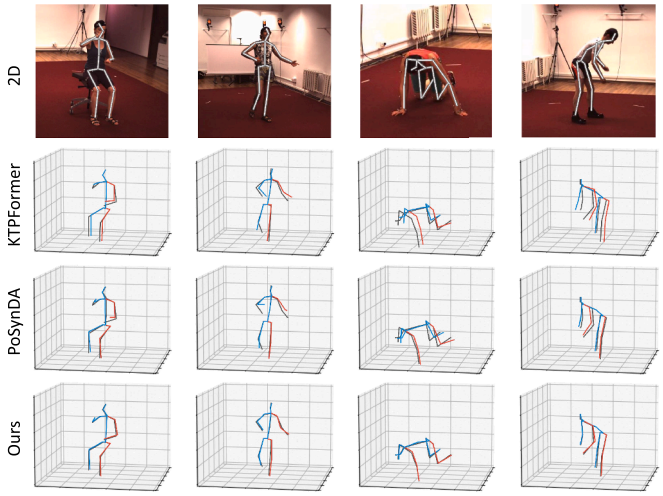

Fig. 5. Visualization on Human3.6M. *Source:* S1. *Target:* (from left to right): S5, S6, S7, and S8. The black skeleton indicates the ground truth.

4.3.2. Results on MPI-INF-3DHP and 3DPW

Table 3 presents the cross-dataset evaluation results where Human3.6M is used as the source domain and MPI-INF-3DHP as the target domain. The results demonstrate that our method maintains stable performance across datasets and consistently achieves the best results under various evaluation metrics. Furthermore, in a more challenging cross-dataset setting where 3DPW serves as the target domain, our method achieves comparable performance to competing approaches that utilize 2D annotations from the target domain-despite our method not using any target-domain 2D data. The results are shown in Table 4. This strong generalization ability can be attributed to our framework's domain-invariant modeling of both structure and dynamic features, enabling effective generalization to previously unseen domains.

Table 3
Cross-dataset evaluation on MPI-INF-3DHP dataset. *Source:* Human3.6M. Target: MPI-INF-3DHP.

Method	2D	PCK (↑)	AUC (↑)	MPJPE (↓)
DH-AUG [39]		87.7	55.8	74.2
AdaptPose [33]	✓	88.4	54.2	77.2
PoseAug-V [14]		83.6	52.1	82.7
PoSynDA [17]	✓	93.5	59.6	58.2
Ours		**95.0**	**60.2**	**56.7**

4.4. Qualitative results

4.4.1. Cross-domain pose estimation results visualization

We present partial visualization results in Fig. 5 on the Human3.6M dataset, where subject S1 is used as the source domain and subjects

Table 4
Cross-dataset evaluation on 3DPW dataset. *Source:* Human3.6M. Target: 3DPW.

Method	2D	MPJPE	P-MPJPE
VIBE [40]	✓	93.5	56.5
BOA [41]	✓	77.2	49.5
DH-AUG [39]		92.4	55.0
AdaptPose [33]	✓	81.2	46.5
PoseAug-V [14]		91.1	54.3
PoSynDA [17]	✓	75.5	**45.4**
Ours		**75.1**	45.8

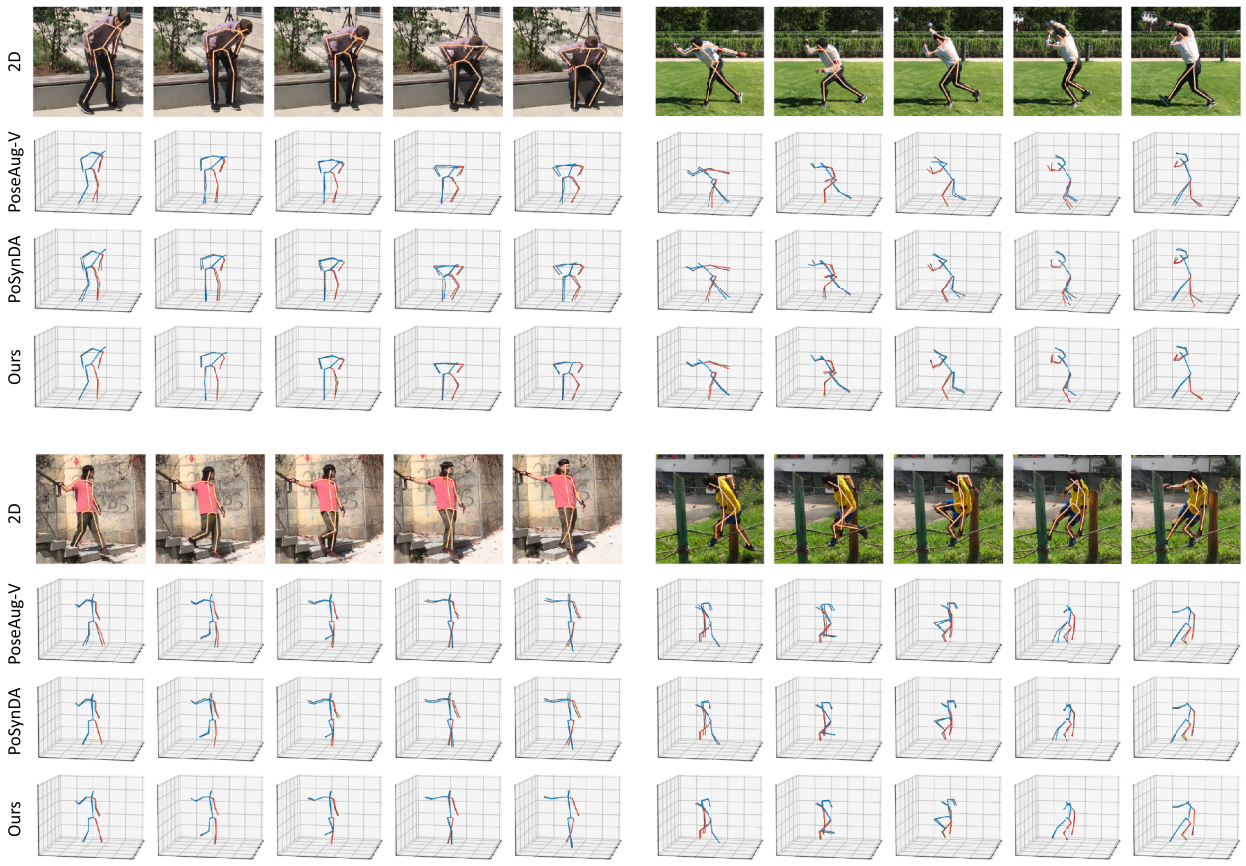

Fig. 6. Visualization on 3DPW. The time increases from left to right in each segment.

S5, S6, S7, and S8 serve as the target domains. It can be observed that our method achieves accurate pose estimations even when the skeletal structures and body proportions of the individuals in the test scenarios are unseen during training. Fig. 6 further shows visualization results from a cross-dataset setting where Human3.6M is used as the source and 3DPW as the target domain. The left column displays examples with motion patterns similar to those in the training set, while the right column shows entirely unfamiliar motion patterns. Despite the differences in both subject identities and motion types, our method still delivers robust and consistent predictions, demonstrating strong generalization ability.

Additionally, we selected challenging cases caused by external object occlusion, self-occlusion, and variations in clothing or body shape, and compared the predictions of DAK-Pose with baseline methods to evaluate the deployment capability of our approach in real-world scenarios. As shown in Fig. 7, in the first case of external occlusion where a person's legs are partially occluded by furniture, the baseline method exhibits noticeable drift at the knee joint. In the second scenario involving self-occlusion where parts of the torso occlude the limbs, DAK-Pose still maintains more reasonable pose estimates. This can be attributed to the domain-invariant feature representations learned by DAK-Pose from diverse synthetic data, which enhance its robustness to missing local observations. Furthermore, when handling non-standard body shapes and loose clothing, the human skeletons predicted by DAK-Pose align more closely with anatomical constraints in terms of body proportions and joint angles. This improvement benefits from SMA, which reinforces reasonable human body structures during training.

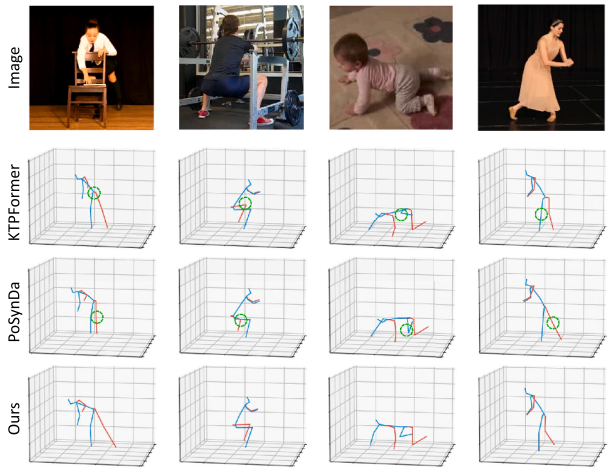

Fig. 7. Visualization of extreme cases in real-world scenarios. Improper positions are highlighted with green dashed circles.

4.4.2. *Generalization mechanism and feature space visualization*

We visualize synthesized motion sequences generated by SMA and DMA in DAK-Pose, see Fig. 8. The joint distributions indicate that SMA produces motions with strictly consistent limb lengths across frames. However, analysis of the dynamic feature maps reveals a lack of smooth and realistic temporal dynamics, with evident noise and substantial deviations from the reference motion, particularly in the regions high-

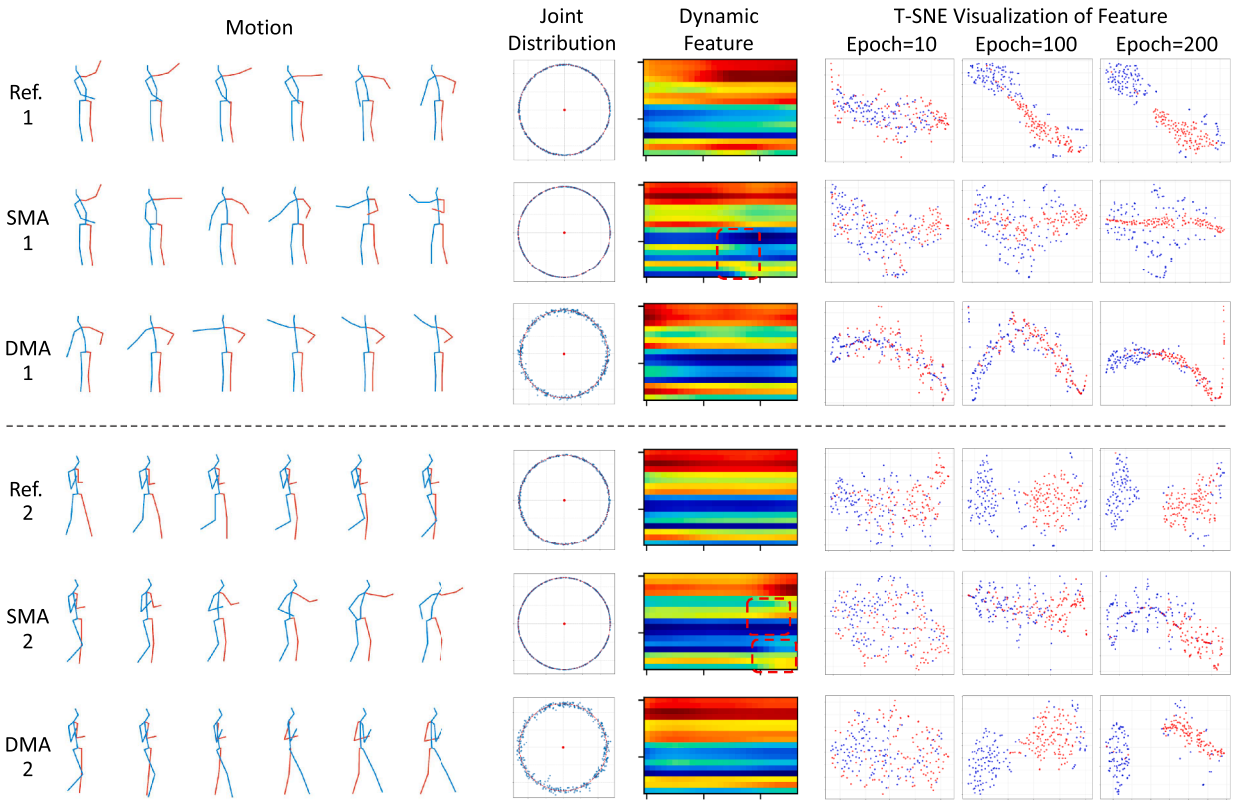

Fig. 8. Visualization of augmented motion and feature. In the joint distribution, blue dots represent the normalized positions of joints relative to their parent nodes (red centers). In the dynamic feature, the vertical axis denotes joint indices following the Human3.6M convention, and the horizontal axis represents time. In the T-SNE visualization of feature, red points for structural features and blue points for dynamic features.

lighted by red dashed boxes. In contrast, the DMA generates motions with more natural and smooth temporal transitions, although the limb proportions are less consistent. Meanwhile, we performed t-SNE visualization on the structural and dynamic features of each type of motion to intuitively analyze the feature disentanglement process of different synthesized motions and their respective characteristics. Specifically, we illustrated the motion feature extraction results of the model at three distinct stages of the training process. It can be observed that as training progresses, the degree of feature disentanglement increases accordingly. Moreover, when the features are sufficiently disentangled, the distribution of dominant features corresponding to enhancement motions with different preferences shows higher aggregation relative to their other features. These results confirm the complementary strengths and weaknesses of the two augmentation strategies, thereby validating the effectiveness of our domain adaptation design in the feature space.

To demonstrate the effectiveness of domain alignment between the features of the two types of enhanced motions and their real-motion counterparts, we conducted a t-SNE visualization in the feature space using the two disentangled features of enhanced motions generated by different enhancers, as shown in Fig. 9. In the structural feature space, the feature points of real data and the structure-prioritized enhanced motions synthesized by the SMA show significant clustering, while the structural features of the dynamic-prioritized enhanced motions produced by the DMA are noticeably separated from the main cluster. Conversely, in the dynamic feature space, the feature points of real data and dynamic-prioritized enhanced motions are highly mixed, whereas the dynamic features of structure-prioritized enhanced motions form an isolated cluster. These observations confirm that both enhancers provide high-quality knowledge and indicate that the DAK-Pose framework achieves a complementary knowledge fusion mechanism.

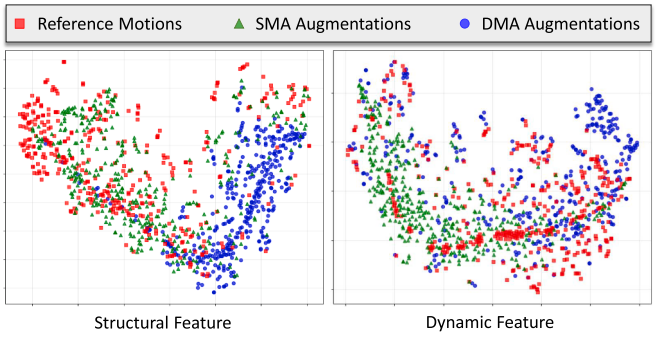

Fig. 9. T-SNE visualization of domain alignment. Each point in the plot corresponds to a specified feature of a motion, derived from the Human3.6M dataset (S1, GT).

4.5. Ablation study

4.5.1. Ablation on each module

We trained several models to individually assess the contribution of each module within our overall framework. First, we define a baseline by training the pose estimation network G using only the basic pose estimation loss $\mathcal{L}_{pe}$. To evaluate the influence of SMA, we remove the corresponding structural loss $\mathcal{L}_{adv\_str}$ during training. Similarly, we omit the dynamic loss $\mathcal{L}_{adv\_dyn}$ to assess the role of the DMA. To further evaluate the effectiveness of the adversarial domain generalization process in our framework, we replace it with a direct supervised training strategy

where the synthesized motions from both modules are used to construct 2D-3D paired data for training G directly.

As shown in Table 5, our full framework outperforms the baseline model trained solely with $\mathcal{L}_{pe}$, demonstrating the effectiveness of our pose estimator. Moreover, removing any individual module leads to performance degradation, indicating that each component plays a critical role in the overall architecture. Additionally, our adversarial domain generalization approach yields better results than directly training the pose estimation network on the synthetic paired data, highlighting its ability to mitigate the domain gap between synthetic and real data during model training. We observe that training the model via adversarial domain generalization yields better performance than direct supervision with synthetic data. This suggests that our method can more effectively utilize imperfect synthetic motions by filtering out domain-specific noise. Furthermore, we argue that this advantage stems not only from increased robustness to domain-specific artifacts but, more fundamentally, from the adversarial guidance imposed by the domain discriminator. This guidance forces the pose estimator to actively disregard superficial domain-related features and instead focus on learning deeper, domain-invariant motion patterns that are essential for 3D pose estimation. This process leads the model to capture a more intrinsic motion semantics, achieving a level of generalization that goes beyond mere data fitting pursued by direct supervision on synthetic data.

Table 5
Ablation on each module. "SA": the structure augmentor; "DA": the dynamic augmentor; "Direct": direct supervision with synthetic motions; "ADG": our adversarial domain generalization-based training. MPJPE is reported on Human3.6M (S1,GT).

ID	SA	DA	Direct	ADG	MPJPE
1	×	×	×	×	51.4
2	✓	×	✓	×	45.8
3	×	✓	✓	×	48.3
4	✓	×	×	✓	44.5
5	×	✓	×	✓	46.2
6	✓	✓	×	✓	42.4

4.5.2. Comparison of limb length modeling strategies in SMA

To evaluate the impact of incorporating prior limb length distributions in the structural enhancer, we replace our bone length sampling strategy with the baseline method used in Zhang et al. [14], Peng et al. [15], Li and Pun [16], where limb scales are predicted by a neural network. As shown in Fig. 10, although our approach converges more slowly during the early stages of training (requiring approximately 40% more iterations), it ultimately yields consistently better performance. This is because the baseline method leverages 2D annotations from the target domain as direct supervision, enabling explicit optimization for target-domain performance. In contrast, our Domain Generalization approach is trained without access to any target-domain data. Instead, it must learn more universal motion representations from multiple source domains. Although this setting increases the difficulty of model convergence, it encourages the network to extract more intrinsic and domain-invariant features, ultimately leading to stronger generalization capability.

4.5.3. Analysis of the sampling parameter m in DMA

We investigate the impact of the parameter m, which controls the sampling range of pose sequence segments in the dynamic enhancer. As shown in Fig. 11, smaller m values correspond to longer reference sequences. However, this leads to suboptimal performance, likely due to insufficient diversity in motion dynamics, which limits the model's generalization capability. On the other hand, when m exceeds 50, model performance drops significantly. This is primarily caused by the accumulation of errors in long-term motion prediction, such as joint drift or frozen poses, making the generated sequences unreliable. Based on these observations, we randomly sample m within the range of [30, 50] in our experiments, which effectively enhances the robustness of the dynamic encoder while ensuring the reliability of the extracted dynamic features.

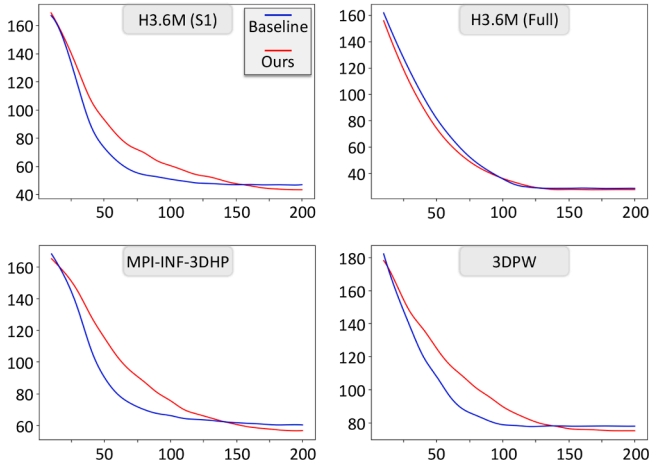

Fig. 10. Impact of bone length acquisition strategies in the structure-prioritized augmentor on model training and accuracy. Y-axis: MPJPE; X-axis: the number of training epochs.

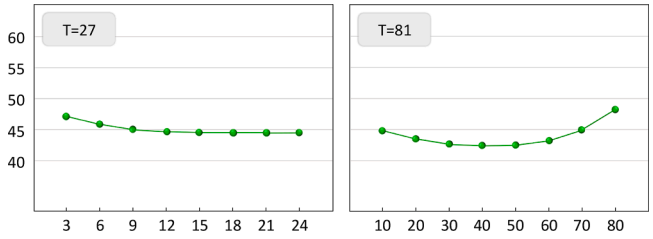

Fig. 11. Effect of parameter m in the dynamic-prioritized augmentor. Y-axis: MPJPE; X-axis: value of m.

4.5.4. Impact of parameters of structure augmentation

Limb scale techniques in Yang et al. [23] and Zhu et al. [24] are employed to address limb structure inconsistencies in virtual humanoid characters. As such, the original design of limb scaling involves sampling random values from a uniform distribution within a fixed range. To simulate the distribution of human limb proportions in the real world, we adopt normally distributed random limb scale factors F. Specifically, we decompose limb scaling into two simultaneous processes: local scaling and global scaling (F_l and F_g). And we select scaling factors from a right-truncated normal distribution $F \sim \mathcal{N}(1, \sigma)$ to ensure a balanced range of magnification and reduction. We compare our method with the uniform limb scale distribution proposed in Yang et al. [23]. In addition, we evaluate the proposed joint perturbation under cross-dataset settings and observe consistent improvements. The results are presented in Table 6.

We further analyze the suitable range of the standard deviation σ for our framework. As shown in Fig. 12, the best performance is achieved when the local scaling factor $F_l \sim \mathcal{N}(1, 0.2)$ and the global scaling factor $F_g \sim \mathcal{N}(1, 0.3)$. The results also indicate that the optimal σ for local scaling is smaller than that for global scaling. We attribute this to the observation that, in the real world, the absolute length of the same limb (e.g., leg length) varies significantly across individuals, while the relative pro-

Table 6
Ablation on limb scale distribution and joint perturbation. The values reported correspond to MPJPE. "JP": joint perturbation.

Distribution	JP	H36M (S1)	H36M (Full)	3DHP	3DPW
Uniform [23]	×	–	–	60.5	80.8
Uniform [23]	✓	44.5	25.7	59.3	78.6
Normal (Ours)	×	–	–	58.0	77.4
Normal (Ours)	✓	42.4	24.8	56.7	75.1

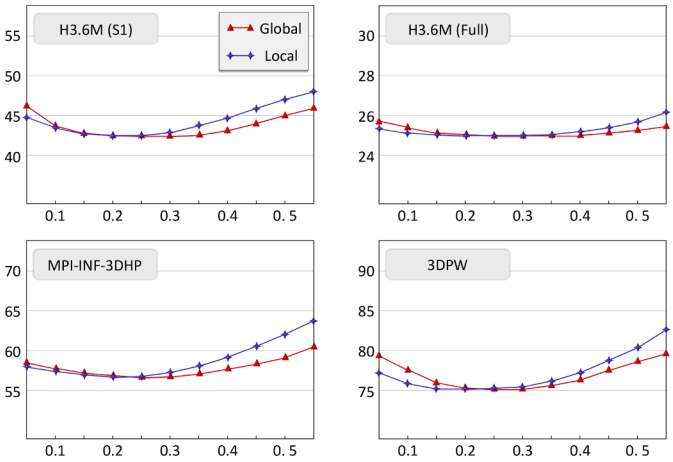

Fig. 12. Limb scale range analysis. X-axis: value of σ in the normal distribution; Y-axis: MPJPE.

Table 7
Analysis of network structural parameters. L and K indicate the number of recurrent iterations in the encoder and decoder, respectively. C denotes the feature dimension. MPJPE is reported on Human3.6M (S1,GT).

L	K	C	Params(M)	MPJPE
3	6	512	31.9	43.5
4	6	512	34.1	42.4
5	6	512	36.3	42.4
6	6	512	38.4	42.7
4	3	512	19.4	47.1
4	4	512	25.7	43.7
4	5	512	29.9	42.8
4	**6**	512	34.1	42.4
4	7	512	38.4	42.4
4	8	512	42.6	42.9
4	6	128	2.2	50.0
4	6	256	8.7	44.3
4	6	**512**	34.1	42.4
4	6	1024	135.0	42.2

portions between different limbs (e.g., thigh-to-shank ratio) within the same individual tend to be more consistent across the population.

4.5.5. Impact of network structural hyperparameters

Our network involves three major hyperparameters: the number of loop blocks in the encoder (L), the number of loop blocks in the decoder (K), and the feature dimension of the intermediate layers (C). To evaluate the influence of each hyperparameter independently, we vary one while keeping the other two fixed. As shown in Table 7, although increasing C to 1024 introduces additional model parameters, it does not yield a significant performance gain. Therefore, we identify $L = 4$, $K = 6$, and $C = 512$ as the optimal configuration for balancing model complexity and accuracy.

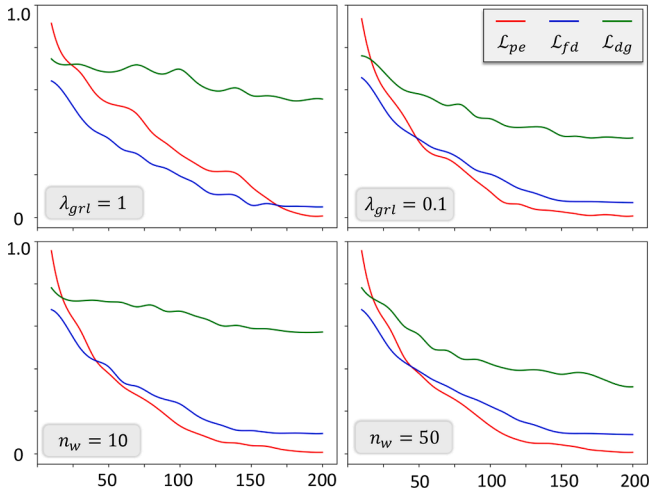

Fig. 13. Training loss. X-axis: Epoch; Y-axis: Normalized loss values from Human3.6M (S1,GT).

4.5.6. Ablation on the warm-up strategy for adversarial training

To thoroughly evaluate the training robustness of the DAK-Pose framework, we monitored and analyzed the evolution of key loss terms during training. Specifically, we compared the convergence curves of various losses when using different fixed weighting coefficients $-\lambda_{grl}$ and different warm-up epoch values n_w during backpropagation. As shown in Fig. 13, employing a fixed and large λ_{grl} caused the L_{dg} loss to exhibit significant oscillations in the early stages, which interfered with the normal decrease of L_{pe}. This indicates the occurrence of gradient conflict. In contrast, with the progressive warm-up strategy, L_{dg} decreased stably, and all loss curves descended cooperatively, eventually converging to their lowest values. This result demonstrates the critical role of the warm-up strategy in ensuring training stability.

4.5.7. Analysis of computational efficiency

We report the FLOPs, GPU memory usage, computational time (including total training time and inference speed), number of model parameters, and MPJPE metrics during training and testing to comprehensively evaluate the computational efficiency of the proposed model. Since some data augmentation-based methods primarily focus on the data synthesis process and their training pipelines are not directly optimized for 3DHPE models, we selected non-cross-domain 3DHP methods-MixSTE, D3DP, and KTPFormer-along with the data augmentation-based method PoSynDa as baseline models. Furthermore, to provide an in-depth analysis of the computational efficiency of internal modules, we also report the performance of our model under two configurations: without augmentation and with only structural augmentation. All experiments were conducted on a single NVIDIA RTX 4090 GPU. It should be noted that when using the same batch size as other baseline methods, the GPU memory usage of PoSynDa during training exceeded the capacity of a single GPU. Therefore, its training-phase memory usage was measured in a multi-GPU environment. The specific results are shown in Table 8. As can be observed, the FLOPs and memory usage of our method during training are significantly lower than those of PoSynDa, which is also a data augmentation-based approach. While diffusion-based methods D3DP and KTPFormer exhibit low overhead during training, their inference efficiency is poor due to the complexity of the diffusion-based inference mechanism. In summary, our method achieves notable performance improvements with reasonable training costs while maintaining efficient inference.

Table 8
Computational efficiency. "w/o Aug.": without augmentation; "1 Aug.": structural augmentation only. MPJPE is reported on Human3.6M (S1,GT).

	Training			Inference				
	FLOPs (G)	Memory (GB)	Time (Hours)	FLOPs (G)	Memory (GB)	Speed (FPS)	Params (M)	MPJPE
MixSTE [7]	0.57	3.5	15	0.57	3.4	2375	33.7	48.3
D3DP [36]	0.57	3.3	36	228.2	3.7	99	35.0	46.0
KTPFormer [8]	0.57	4.1	19	228.4	3.8	92	34.1	43.8
PoSynDa [17]	3.44	160.8	28	0.57	3.4	2375	34.8	43.1
Ours (w/o Aug.)	0.59	3.6	15	0.59	3.6	2133	34.1	51.4
Ours (1 Aug.)	1.62	5.4	20	0.59	3.6	2133	34.1	44.5
Ours	2.61	14.2	23	0.59	3.6	2133	34.1	42.4

5. Conclusion

In this paper, we propose DAK-Pose, an innovative framework designed to enhance the generalization capability of video-based 3D human pose estimation by augmenting representations in the feature space. We disentangle motion into structural and dynamic features, and employ two complementary motion augmentors to extract their respective strengths. The disentangled features from synthetic motions are then aligned to the pose estimation network through adversarial alignment, effectively mitigating the negative impact of inherent artifacts in synthetic data when used in the observation space. Extensive experiments on multiple datasets demonstrate that DAK-Pose achieves state-of-the-art cross-dataset performance.

CRediT authorship contribution statement

Yachuan Wang: Writing – original draft, Validation, Methodology, Formal analysis, Data curation, Conceptualization; **Bin Zhang:** Supervision, Software, Resources, Project administration, Investigation; **Hao Yuan:** Writing – review & editing, Visualization.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Yuan, Y., Song, J., Iqbal, U., Vahdat, A., Kautz, J., 2023. Physdiff: physics-guided human motion diffusion model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16010–16021.

[2] Zhang, M., Cai, Z., Pan, L., Hong, F., Guo, X., Yang, L., Liu, Z., 2024a. Motiondiffuse: text-driven human motion generation with diffusion model. IEEE Trans. Pattern Anal. Mach. Intell. 46, 4115–4128.

[3] Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J., 2018. Cascaded pyramid network for multi-person pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7103–7112.

[4] Zhang, T., Li, Q., Wen, J., Chen, C.P., 2024b. Enhancement and optimisation of human pose estimation with multi-scale spatial attention and adversarial data augmentation. Inf. Fusion 111, 102522.

[5] Martinez, J., Hossain, R., Romero, J., Little, J.J., 2017. A simple yet effective baseline for 3D human pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2640–2649.

[6] Liu, K., Ding, R., Zou, Z., Wang, L., Tang, W., 2020. A comprehensive study of weight sharing in graph networks for 3D human pose estimation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16, pp. 318–334.

[7] Zhang, J., Tu, Z., Yang, J., Chen, Y., Yuan, J., 2022. 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.

[8] Peng, J., Zhou, Y., Mok, P., 2024a. 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.

[9] Gong, K., Li, B., Zhang, J., Wang, T., Huang, J., Mi, M.B., Feng, J., Wang, X., 2022. Posetriplet: co-evolving 3D human pose estimation, imitation, and hallucination under self-supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11017–11027.

[10] Kim, H.W., Lee, G.H., Nam, W.J., Jin, K.M., Kang, T.K., Yang, G.J., Lee, S.W., 2024. Mhcanonnet: multi-hypothesis canonical lifting network for self-supervised 3D human pose estimation in the wild video. Pattern Recognit. 145, 109908.

[11] Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M., 2019. 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.

[12] Song, X., Li, Z., Chen, S., Demachi, K., 2024. Quater-gcn: enhancing 3D human pose estimation with orientation and semi-supervised training. In: ECAI 2024. IOS Press, pp. 121–128.

[13] Gong, K., Zhang, J., Feng, J., 2021. Poseaug: a differentiable pose augmentation framework for 3D human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8575–8584.

[14] Zhang, J., Gong, K., Wang, X., Feng, J., 2023. Learning to augment poses for 3D human pose estimation in images and videos. IEEE Trans. Pattern Anal. Mach. Intell. 45, 10012–10026.

[15] Peng, Q., Zheng, C., Chen, C., 2024b. A dual-augmentor framework for domain generalization in 3D human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2240–2249.

[16] Li, H., Pun, C.M., 2023. Cee-net: complementary end-to-end network for 3D human pose generation and estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 1305–1313.

[17] Liu, H., He, J.Y., Cheng, Z.Q., Xiang, W., Yang, Q., Chai, W., Wang, G., Bao, X., Luo, B., Geng, Y., et al., 2023. Posynda: multi-hypothesis pose synthesis domain adaptation for robust 3D human pose estimation. In: Proceedings of the 31st ACM International Conference on Multimedia, pp. 5542–5551.

[18] Kim, J.H., Lee, S.W., 2024. Toward approaches to scalability in 3D human pose estimation. Adv. Neural Inf. Process. Syst. 37, 105476–105502.

[19] Denton, E.L., et al., 2017. Unsupervised learning of disentangled representations from video. Adv. Neural Inf. Proc. Syst. 30, 4417–4426.

[20] Wei, Z., Chen, L., Tu, T., Ling, P., Chen, H., Jin, Y., 2023. Disentangle then parse: night-time semantic segmentation with illumination disentanglement. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 21593–21603.

[21] Sun, X., Feng, Q., Li, X., Zhang, J., Lai, Y.K., Yang, J., Li, K., 2023. Learning semantic-aware disentangled representation for flexible 3D human body editing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16985–16994.

[22] Aumentado-Armstrong, T., Tsogkas, S., Dickinson, S., Jepson, A., 2023. Disentangling geometric deformation spaces in generative latent shape models. Int. J. Comput. Vis. 131, 1611–1641.

[23] Yang, Z., Zhu, W., Wu, W., Qian, C., Zhou, Q., Zhou, B., Loy, C.C., 2020. Transmomo: invariance-driven unsupervised video motion retargeting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5306–5315.

[24] Zhu, W., Yang, Z., Di, Z., Wu, W., Wang, Y., Loy, C.C., 2022. Mocanet: motion retargeting in-the-wild via canonicalization networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 3617–3625.

[25] Remelli, E., Han, S., Honari, S., Fua, P., Wang, R., 2020. Lightweight multi-view 3D pose estimation through camera-disentangled representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6040–6049.

[26] Honari, S., Constantin, V., Rhodin, H., Salzmann, M., Fua, P., 2022. Temporal representation learning on monocular videos for 3D human pose estimation. IEEE Trans. Pattern Anal. Mach. Intell. 45, 6415–6427.

[27] He, J., Yang, W., 2024. Video-based human pose regression via decoupled space-time aggregation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1022–1031.

[28] Xu, G., Tao, J., Li, W., Duan, L., 2024. Learning semantic latent directions for accurate and controllable human motion prediction. In: European Conference on Computer Vision, pp. 56–73.

[29] Wang, X., Zhang, W., Wang, C., Gao, Y., Liu, M., 2023. Dynamic dense graph convolutional network for skeleton-based human motion prediction. IEEE Trans. Image Process. 33, 1–15.
[30] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., Lempitsky, V., 2016. Domain-adversarial training of neural networks. J. Mach. Learn. Res. 17, 1–35.
[31] Tzeng, E., Hoffman, J., Saenko, K., Darrell, T., 2017. Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7167–7176.
[32] Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C., 2013. Human3. 6m: large scale datasets and predictive methods for 3D human sensing in natural environments. IEEE Trans. Pattern Anal. Mach. Intell. 36, 1325–1339.
[33] Gholami, M., Wandt, B., Rhodin, H., Ward, R., Wang, Z.J., 2022. Adaptpose: cross-dataset adaptation for 3D human pose estimation by learnable motion generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13075–13085.
[34] Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.P., Xu, W., Casas, D., Theobalt, C., 2017. Vnect: real-time 3D human pose estimation with a single RGB camera. ACM Trans. Graph. (TOG) 36, 1–14.
[35] Von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G., 2018. Recovering accurate 3D human pose in the wild using imus and a moving camera. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 601–617.
[36] Shan, W., Liu, Z., Zhang, X., Wang, Z., Han, K., Wang, S., Ma, S., Gao, W., 2023. Diffusion-based 3D human pose estimation with multi-hypothesis aggregation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 14761–14771.
[37] Cai, Q., Hu, X., Hou, S., Yao, L., Huang, Y., 2024. Disentangled diffusion-based 3D human pose estimation with hierarchical spatial and temporal denoiser. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 882–890.
[38] Chai, W., Jiang, Z., Hwang, J.N., Wang, G., 2023. Global adaptation meets local generalization: unsupervised domain adaptation for 3D human pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 14655–14665.
[39] Huang, L., Liang, J., Deng, W., 2022. DH-AUG: DH forward kinematics model driven augmentation for 3D human pose estimation. In: European Conference on Computer Vision, pp. 436–453.
[40] Kocabas, M., Athanasiou, N., Black, M.J., 2020. Vibe: video inference for human body pose and shape estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5253–5263.
[41] Guan, S., Xu, J., Wang, Y., Ni, B., Yang, X., 2021. Bilevel online adaptation for out-of-domain human mesh reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10472–10481.