Pattern Recognition 183 (2027) 114950



Contents lists available at ScienceDirect

Pattern Recognition

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





Pose2Point: Bridging human poses and point clouds for unified action recognition

Jiaying Lin [a,b], Jiajun Wen [c], Weiwei Wan [d], Mengyuan Liu [a,b,*]

[a] State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School, China
[b] Guangdong Provincial Key Laboratory of Ultra High Definition Immersive Media Technology, Peking University, Shenzhen Graduate School, China
[c] Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
[d] Graduate School of Engineering Science, Osaka University, Toyonaka, Osaka, Japan

ARTICLE INFO

Keywords:
Action recognition
Human pose
Point cloud

ABSTRACT

Pose and point cloud representations are inherently complementary for human action recognition: pose provides compact structural priors, while point clouds offer flexible geometric modeling. Although multimodal approaches attempt to combine these advantages, existing dual-input architectures often suffer from a strong reliance on paired data and complex cross-modal alignment. To address these issues, we propose the Pose2Point (P2P) framework, which reformulates pose sequences into a Pose Point Cloud (PPC) representation. By achieving data-level unification at the source, P2P deeply integrates pose structural priors with point cloud geometric flexibility within a single-modality paradigm. The framework introduces a Transfer2PointCloud module to convert pose sequences into point clouds encoding 3D spatial coordinates and spatiotemporal topology, enabling the joint preservation of pose semantics and motion dynamics. By selecting points based on joint distribution, the proposed Uniform Mixed Sampling (UMS) strategy circumvents the structural and temporal destruction inherent in standard downsampling, ensuring the preservation of pose topology and motion continuity. On the resulting PPCs, hierarchical neighborhood aggregation is employed to capture fine-grained local interactions and high-level action semantics. In addition, a Multi-CLS loss and PPC-Specific Training Strategy strategies are incorporated to enhance optimization and robustness. Extensive experiments on NTU RGB+D, NTU RGB+D 120, and NW-UCLA demonstrate that Pose2Point achieves state-of-the-art performance, validating the effectiveness of the proposed unified point cloud representation for pose-based action recognition. Code is publicly available at: https://github.com/jolin830/Pose2Point.

1. Introduction

Human action recognition (HAR) is a crucial task in video understanding and human motion understanding, aiming to accurately interpret and classify human activities by analyzing the dynamic actions and motion patterns [1–4]. To achieve precise recognition, various video modalities have been explored, such as pose sequences [5–7], RGB images [8,9], and point clouds [10,11].

Pose-based HAR methods have received extensive attention due to their efficiency, lightweight representation, and robustness to background variations. Since human poses can naturally be represented as graphs, most existing methods model pose data based on a predefined topological structure, as shown in Fig. 1(a), where joints are treated as nodes and bones as edges, and feature learning is performed via information propagation on the graph. By explicitly modeling human structural priors, these methods have achieved strong performance on multiple benchmark datasets. However, the reliance on fixed topology also imposes a fundamental limitation on modeling flexibility. Human actions are essentially continuous motions of joints in space, and many motion patterns involve cross-topology joint interactions, such as hand–hand coordination (e.g., clapping), hand–head interaction (e.g., wear on glasses), and hand–foot interaction (e.g., wear a shoe). In these cases, the involved joints are physically close in space but far apart in the predefined topology, or even directly disconnected. In topology-based methods, information exchange between such non-adjacent joints must be established through multi-hop message passing, which not only increases the length of the propagation path but may also lead to gradual attenuation of critical features, thereby impairing the modeling of local coordinated motions.

In contrast to pose representations, point clouds are unordered geometric representations whose core modeling paradigm is based on local aggregation within spatial neighborhoods rather than fixed structure

* Corresponding author.
E-mail address: liumengyuan@pku.edu.cn (M. Liu).

https://doi.org/10.1016/j.patcog.2026.114950
Received 30 April 2026; Received in revised form 21 August 2026; Accepted 15 September 2026
Available online 22 September 2026
0031-3203/© 2026 Published by Elsevier Ltd.

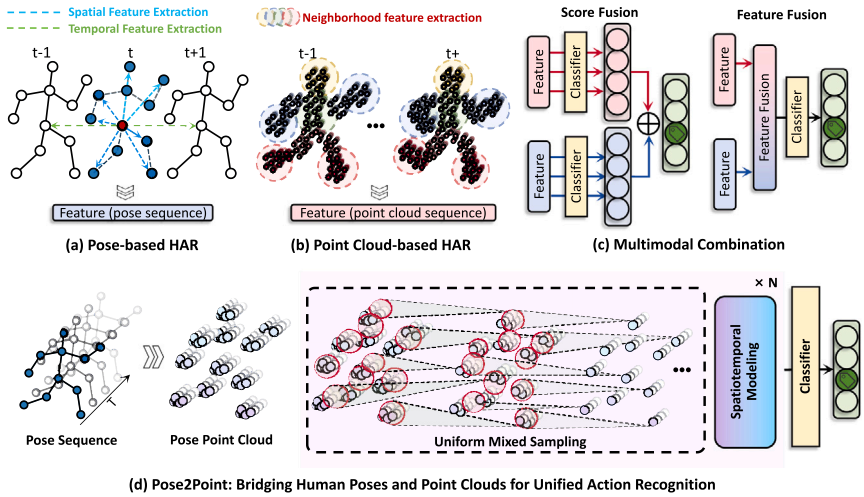


Fig. 1. A comparison between our Pose2Point and previous action recognition approaches. (a) and (b) illustrate the feature extraction methods for pose-based and point cloud-based modalities, respectively, while (c) process pose and point cloud sequences as separate branches. In contrast, we propose Pose2Point (d), which unifies pose sequences within a point cloud processing framework.

as in Fig. 1(b). Point cloud deep learning methods are naturally effective at capturing local geometric relationships. Point cloud methods model local interactions based on spatial proximity, enabling effective aggregation of neighboring points regardless of their connectivity in the original topology. This distance-driven paradigm compensates for the limitations of fixed topology in capturing physically close yet topologically distant joint interactions. However, point cloud-based HAR remains challenging due to high computational cost caused by large-scale, unordered data, as well as the lack of explicit structural priors, which hinders the modeling of coordinated joint interactions and structured motion patterns. In contrast, pose representations are compact and semantically explicit, providing strong priors for human motion modeling.

Consequently, pose and point cloud representations exhibit an inherent complementarity for human action recognition: while point clouds excel at flexible geometric modeling, poses provide efficient and structured semantic priors. Most prevailing approaches leverage multimodal fusion to combine these respective strengths, as illustrated in Fig. 1(c). However, such dual-input architectures encounter a significant representation bottleneck. They primarily suffer from a heavy reliance on paired datasets—a requirement that many existing skeleton benchmarks fail to meet. Furthermore, fusion at the feature level often demands intricate alignment, leading to increased training difficulty and overhead. Fundamentally, existing multimodal methods continue to treat skeletons and point clouds as isolated entities, focusing on feature-level manipulation rather than data-level unification. This modality gap constrains the capacity to deeply explore the latent relationships between human joints and spatial geometry. Therefore, a natural question arises: *Can we deeply integrate the complementary advantages of both representations within a unified, single-modality framework?*Building upon these insights, we propose Pose2Point (P2P), a unified framework for pose-based human action recognition based on point cloud representation learning as shown in Fig. 1(d). By reformulating pose sequences into a point cloud format, *P2P achieves data-level unification at the source, allowing joint structural priors to seamlessly guide geometric point-wise modeling within a single modality*. Meanwhile, it leverages the flexible neighborhood aggregation inherent in point clouds to effectively capture long-range cross-joint interactions. By synergizing skeleton structural priors with geometric flexibility, P2P establishes a simplified yet effective paradigm that circumvents the pitfalls of multimodal alignment, yielding a more cohesive and efficient end-to-end feature extraction scheme. Pose2Point first transforms pose sequences into dense Pose Point Clouds (PPCs) through the Transfer2PointCloud module, which encodes both 3D positions and spatiotemporal dynamics. To preserve pose topology and motion continuity while selecting diverse and informative points, we apply a Uniform Mixed Sampling (UMS) strategy that supports effective downstream feature learning. The downsampled point set is then processed using hierarchical point-wise local neighborhood aggregation and global feature abstraction to capture both fine-grained local joint interactions and holistic action semantics. We further enhance optimization by incorporating a Multi-CLS loss, which strengthens gradient propagation and provides intermediate supervision, and complement this with a PPC-specific data augmentation strategy to boost performance. Extensive experiments were conducted on benchmark datasets, including NTU RGB+D, NTU RGB+D 120, and NW-UCLA. The results demonstrate that Pose2Point achieves state-of-the-art performance comparable to advanced human pose models, thereby validating its effectiveness.

Our main contributions can be summarized as follows:

- We propose Pose2Point, a unified framework that reformulates HAR through a Pose Point Cloud representation. It seamlessly integrates pose structural semantics with geometric point cloud modeling within a cohesive single-modality paradigm.
- We introduce a Transfer2PointCloud module to encode pose sequences into PPC with explicit spatiotemporal dynamics, enabling point cloud networks to leverage structural priors for modeling action evolution.
- We develop a Uniform Mixed Sampling strategy that preserves pose topology and motion continuity, circumventing the structural destruction typical of conventional point cloud downsampling.
- Extensive Experiments on NTU-RGB+D, NTU-RGB+D 120, and NW-UCLA demonstrate state-of-the-art performance, establishing P2P as a simple and efficient paradigm for pose-based action recognition.

2. Related work

2.1. Pose-based action recognition

Deep learning has been widely adopted for pose-based action recognition, with representative approaches based on CNNs, GCNs, and Transformers [12–15]. CNN-based methods typically transform pose sequences into pseudo-images to capture spatial–temporal patterns [12]. GCN-based approaches represent human poses as skeletal graphs and

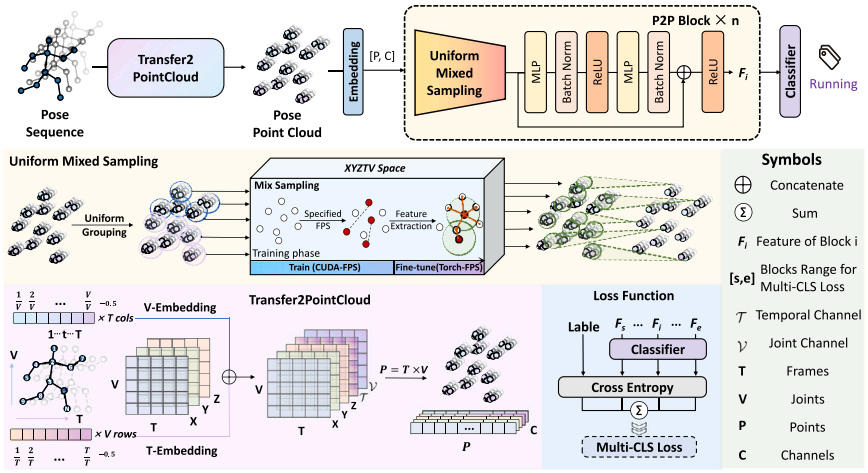


Fig. 2. Framework of the proposed Pose2Point Network (P2P). The input pose sequence is transformed into a pose point cloud via the Transfer2PointCloud module, encoding both joint and temporal information into a unified 3D representation. Each P2P block adopts Uniform Mixed Sampling to ensure structured and uniform sampling over spatiotemporal dimensions, enabling effective global and local modeling. Furthermore, Multi-CLS loss is introduced to guide multi-level feature learning and improve overall representation quality.

exploit graph convolutions to model the predefined topological relationships among joints [13]. More recently, Transformer-based methods have been introduced to capture global dependencies across joints and frames [14]. Despite their effectiveness, most existing methods still rely on predefined skeletal topology, which may limit their ability to model interactions between physically close but topologically distant joints. In contrast, Pose2Point converts pose sequences into PPCs, enabling flexible geometric modeling of cross-joint interactions within a unified point cloud framework.

2.2. Point cloud based on deep learning

Point cloud can be categorized into static and dynamic point clouds. Static point clouds represent 3D points sampled at a single moment and are widely used for object and scene understanding [16–19], with recent MLP-based and Transformer-based models such as PointMLP [17] showing strong classification performance. Point cloud processing has also been extended to geometric completion and LiDAR upsampling [20–22]. Dynamic point clouds model temporal sequences and have been explored for action recognition [10,11,23–25]. However, existing point cloud-based action recognition methods are mainly designed for dense point clouds and lack explicit pose structural priors. Pose2Point bridges this gap by converting pose sequences into PPCs, combining pose semantics with point cloud geometric modeling in a unified single-modality framework.

2.3. Multimodal combination of human pose and point cloud

Pose and point cloud data offer complementary information for action recognition [26]. Early dual-branch methods process the two modalities independently but require high-quality paired data and substantial computation [27]. More efficient approaches convert pose sequences into 3D point clouds [28,29]. However, representing an entire sequence as a static point cloud discards motion dynamics [28], while spatiotemporal color encoding lacks continuous temporal modeling and topological awareness [29]. Moreover, existing point cloud paradigms rarely account for the structural characteristics of poses. In contrast, our unified framework preserves spatiotemporal topology while leveraging local geometric modeling, enabling efficient and robust end-to-end representation.

3. Method

The architecture of P2P is illustrated in Fig. 2. Following the problem formulation (Section 3.1), P2P first transforms a pose sequence into a Pose Point Cloud via the Transfer2PointCloud module (Section 3.2). This PPC is then processed through a series of P2P blocks, incorporating Uniform Mixed Sampling to preserve joint consistency and temporal continuity (Section 3.3). Finally, the refined features are extracted by a backbone network, optimized with a Multi-CLS loss (Section 3.4) and specialized training strategies (Section 3.5) to enhance recognition performance.

3.1. Problem setting

Pose data offers a compact and structured representation of human motion, capturing key joint positions and their dynamics. For a pose sequence S comprising T frames, the pose at frame t is represented as $S_t = \{j_1^t, \dots, j_i^t, \dots, j_V^t\} \in \mathbb{R}^{V \times C}$, where j_i^t denotes the ith joint at frame t, V is the number of joints, and C denotes the number of channels per joint. Consequently, the entire pose sequence can be uniformly represented as $S = (S_1, \dots, S_t, \dots, S_T) \in \mathbb{R}^{T \times V \times C}$. In this work, we use 3D pose data, where each joint is represented by its (x, y, z) coordinates, so that a pose sequence is given by $S \in \mathbb{R}^{T \times V \times 3}$.

Point cloud data serves as a crucial representation for describing the geometric structure of objects. For a point cloud $P \in \mathbb{R}^{N \times C}$, it can be defined as: $P = \{p_1, \dots, p_i, \dots, p_N\}, p_i \in \mathbb{R}^C$, where N denotes the number of points, and each point p_i has C channels. Typically, these channels include the (x, y, z) coordinates in 3D space, and may also contain other features such as RGB.

3.2. Transfer2PointCloud

To accurately convert pose data into a Pose Point Cloud without losing critical information, it is essential to fully consider the characteristics of both pose representations and point clouds. If a single-frame pose S_t is directly treated as a point cloud, the limited number of joints leads to severe information loss during point cloud learning. A natural solution is to stack the entire pose sequence along the temporal axis to form a denser point cloud. However, such direct stacking destroys the temporal order and topological structure of the pose sequence, making it difficult to capture motion continuity and dynamic evolution. To address this, Pose2Point first performs Temporal-Vertex Embedding (TV-Embedding) on each joint in the pose sequence. By incorporating

spatio-temporal structural prior to stacking, the resulting Pose Point Cloud becomes both dense and structurally consistent, while preserving temporal coherence.

Before merging joints across frames, Pose2Point applies TV-Embedding to each 3D joint j_i^t in $S \in \mathbb{R}^{T\times V\times 3}$, augmenting it with a temporal channel $\mathcal{T}$ and a spatial channel $\mathcal{V}$ to obtain the enhanced 5D joint $\tilde{j}_i^t$:

$$\tilde{j}_i^t = \left[j_i^t;\ \frac{t}{T} - 0.5;\ \frac{i}{V} - 0.5 \right] \in \mathbb{R}^5, \tag{1}$$

where $[\cdot;\cdot]$ denotes concatenation, the temporal channel $\mathcal{T}$ corresponds to the normalized frame index $\frac{t}{T} - 0.5$, and the spatial channel $\mathcal{V}$ corresponds to the normalized joint index $\frac{i}{V} - 0.5$. Applying TV-Embedding to the input sequence $S \in \mathbb{R}^{T\times V\times 3}$ yields the enhanced representation $S \in \mathbb{R}^{T\times V\times 5}$, where the embedding is applied per-subject for multi-person sequences. All enhanced joints across all frames are reshaped along the temporal dimension to form a Pose Point Cloud $\mathcal{P}$:

$$S \in \mathbb{R}^{T\times V\times 5} \xrightarrow{P=T\times V} \mathcal{P} \in \mathbb{R}^{P\times 5}, \tag{2}$$

where $P = T \times V$ denotes the total number of points in the Pose Point Cloud. Each point is a 5D representation consisting of 3D spatial coordinates (x, y, z), a temporal channel, and a spatial channel. Subsequently, a linear layer is applied to $\mathcal{P}$ for feature embedding, projecting it to a C-dimensional point cloud $\mathcal{P} \in \mathbb{R}^{P\times C}$. Through the Transfer2PointCloud, the pose sequence is transformed into a dense Pose Point Cloud that integrates positional and spatio-temporal information, enabling unified modeling within a point cloud framework. It not only supports effective local–global feature aggregation but also preserves motion continuity, resulting in more expressive representations for action recognition.

3.3. Uniform mixed sampling

The main goal of sampling in point cloud tasks is to reduce the computational cost of analyzing large-scale point clouds while preserving their overall contour. Farthest Point Sampling (FPS) [16] was first proposed and widely used for point cloud grouping. FPS relies on spatial distances to cover the entire point cloud, capturing its maximal contour. This approach is highly effective for dense, widely distributed point clouds with many redundant points. However, pose data differ fundamentally from dense point clouds, making the direct use of FPS inappropriate. Poses consist of a limited set of key joints, each carrying explicit semantic and functional information, forming an inherently structured representation that contrasts with unordered point clouds. FPS selects points based solely on Euclidean distance and thus favors farthest contour points. When applied to PPC, this leads to uneven sampling: distal joints are oversampled, while proximal joints critical for motion representation are under-sampled. This bias undermines human topology and motion continuity, resulting in significant information loss during feature extraction.

To address this, Pose2Point proposes a Uniform Mixed Sampling (UMS) strategy that enforces both sampling uniformity and diversity. UMS ensures uniform coverage over the spatio-temporal domain of the Pose Point Cloud, preserving both pose topology and motion continuity. Furthermore, a mixed sampling strategy combining two variants of FPS is employed to enhance sampling diversity and mitigate the bias of conventional single FPS. Therefore, UMS yields a structurally consistent and information-rich subset for downstream feature learning.

3.3.1. Uniform sampling

Define the input PPC at layer l as $\mathcal{P}^l \in \mathbb{R}^{P^l\times C}$, where P^l denotes the total number of points at layer l. At the initial layer ($l = 0$), $\mathcal{P}^0 \in \mathbb{R}^{(TV)\times C}$, where $P^0 = T \times V$. The input $\mathcal{P}^l$ is partitioned into V disjoint groups according to the joint index encoded in the spatial channel $\mathcal{V}$:

$$\mathcal{P}_v^l = \{\, p_i^l \mid \mathcal{V}(p_i^l) = v \},\quad P_v^l = \frac{P^l}{V},\quad v = 1, 2, \ldots, V, \tag{3}$$

where $\mathcal{P}_v^l$ denotes the set of points corresponding to the vth joint, with $P_v^0 = T$ in particular, and $\mathcal{V}(\cdot)$ extracts the joint index. Each group aggregates all points corresponding to the same joint across different frames, thereby preserving joint-wise structural consistency. Next, each group is independently downsampled with a fixed ratio ρ using the proposed mixed sampling strategy:

$$\hat{\mathcal{P}}_v^l = \mathrm{MixedSampling}\left(\mathcal{P}_v^l,\ \lfloor \rho\, P_v^l \rfloor\right),\quad \rho = 0.5 \tag{4}$$

where $\hat{\mathcal{P}}_v^l$ is the sampled output of the vth joint group at layer l, MixedSampling refers to the mixed sampling method described in Section 3.3.2, and $\rho \in (0, 1)$ is the sampling ratio, set to 0.5 by default. By recursion, the number of sampled output points in each group at layer l can be approximated as:

$$\hat{P}_v^l = \lfloor \rho^l T \rfloor, \tag{5}$$

with $\hat{P}_v^0 = \lfloor \rho T \rfloor$ in particular, ensuring that all joint groups are sampled with the same ratio and avoiding sampling bias across joints. After downsampling, the sampled groups are merged and fed into the point cloud backbone:

$$\hat{\mathcal{P}}^l = \bigcup_{v=1}^{V} \hat{\mathcal{P}}_v^l,\quad \hat{P}^l = \sum_{v=1}^{V} \left|\hat{\mathcal{P}}_v^l\right| = \rho^l P, \tag{6}$$

ensuring a consistent and balanced hierarchical sampling across all joint groups.

3.3.2. Mixed sampling

In point cloud processing, FPS has two common implementations: a PyTorch-based version, referred to as Torch-FPS, and a CUDA-accelerated version, referred to as CUDA-FPS. Typically, models adopt only one of these methods during training. However, each FPS implementation differs in efficiency and exhibits its own selection bias, resulting in different sampling outcomes for the same point cloud. To alleviate this bias and enhance the diversity of sampled points, Pose2Point employs a Mixed Sampling strategy that leverages the strengths of both Torch-FPS and CUDA-FPS by using them in phases during training.

In terms of **computational efficiency**, CUDA-FPS, implemented in CUDA and C++ with full GPU parallelization, runs significantly faster than the pure PyTorch implementation, Torch-FPS. In terms of **sampling strategy**, the differences mainly stem from variations in the initial point selection and the distance update mechanism. For the initial point, Torch-FPS selects a random starting point, introducing stochasticity, whereas CUDA-FPS deterministically uses the first point (index 0). Regarding the update strategy, CUDA-FPS ignores points that are very close to the current reference point during distance updates, which improves efficiency but may alter the subsequent sampling trajectory. In contrast, Torch-FPS updates distances using all points, resulting in a more exhaustive and consistent sampling process. These implementation-level differences introduce a trade-off between efficiency and sampling quality. Empirically, when applied to PPC downsampling, CUDA-FPS achieves faster runtime but tends to produce inferior sampling results, whereas Torch-FPS is slower but yields more reliable and effective samples.

To leverage the complementary advantages of the two FPS implementations, Mixed Sampling adopts a staged training strategy as shown in Algorithm 1. During the main training phase, CUDA-FPS is employed on the PPC to enable fast and efficient sampling across most epochs. In the subsequent fine-tuning phase, Torch-FPS is used to refine the sampling results, enhancing precision and mitigating the selection bias introduced by CUDA-FPS. This staged design not only preserves the computational advantage of CUDA-FPS, but also improves the stability and quality of the final sampled point sets. As a result, Mixed Sampling provides a practical balance between efficiency and sampling quality, yielding more representative point subsets and ultimately boosting downstream performance.

Algorithm 1: Mixed Sampling Strategy in Pose2Point Training

Input: Pose Point Cloud $\mathcal{P}$, total epochs E, main training epochs E_m, fine-tune epochs E_f
Output: Trained network with representative point subsets

```
for e = 1 to E do
    if e ≤ E_m then
        Sample points from P using CUDA-FPS;
    else
        Sample points from P using Torch-FPS;
    Forward sampled points through the P2P network;
```

Table 1
Comparison of accuracy (%) with state-of-the-art **pose-based** methods on the large-scale **NTU-RGB+D 120** dataset (joint stream only). Bold indicates the best performance.

Method	Source	NTU-RGB+D 120	
		X-Sub (%)	X-Set (%)
2S-AGCN [30]	*CVPR'19*	84.00	85.30
Shift-GCN [31]	*CVPR'20*	80.90	83.20
MS-G3D [12]	*CVPR'20*	82.35	84.14
CTR-GCN [32]	*ICCV'21*	84.95	86.68
MSTGCN [33]	*AAAI'21*	82.80	84.50
Info-GCN [34]	*CVPR'22*	85.10	86.30
FR-Head [35]	*CVPR'23*	85.51	87.32
HD-GCN [36]	*ICCV'23*	85.70	87.30
SkeletonGCL [37]	*ICLR'23*	85.20	87.20
SkeMixFormer [38]	*ACMMM'23*	85.82	87.64
DSDC-GCN [39]	*TCSVT'24*	84.00	87.00
Mss-AGCN [40]	*Neur.'24*	85.90	87.30
BlockGCN [13]	*CVPR'24*	86.90	88.20
EPP-Net [26]	*TRIT'24*	85.00	86.70
HSIC-based [41]	*AAAI'25*	86.00	87.40
VA-AR [42]	*AAAI'25*	84.60	87.10
LA-GCN [43]	*TMM'25*	86.50	88.00
Gamba [44]	*CVPR'26*	86.30	–
DSTAFormer [14]	*TCSVT'26*	87.20	87.10
Pose2Point (Ours)	–	**87.34**	**88.22**

Table 2
Comparison of accuracy (%) with state-of-the-art **pose-based** methods on NTU-RGB+D and NW-UCLA datasets (joint stream only). Bold indicates the best performance.

Method	Source	NTU-RGB+D (%)		NW-UCLA
		X-Sub	X-View	(%)
2S-AGCN [30]	*CVPR'19*	88.90	94.50	92.00
Shift-GCN [31]	*CVPR'20*	87.80	95.10	92.50
MS-G3D [12]	*CVPR'20*	88.77	94.88	–
CTR-GCN [32]	*ICCV'21*	88.95	90.23	93.10
MSTGCN [33]	*AAAI'21*	89.00	95.10	–
Info-GCN [34]	*CVPR'22*	89.80	95.20	93.80
FR-Head [35]	*CVPR'23*	90.33	95.26	–
HD-GCN [36]	*ICCV'23*	90.60	95.70	–
SkeletonGCL [37]	*ICLR'23*	90.10	95.50	94.80
TD-GCN [45]	*TMM'23*	90.20	95.00	94.82
SkeMixFormer [38]	*ACMMM'23*	90.73	95.76	–
DSDC-GCN [39]	*TCSVT'24*	90.50	95.60	–
Mss-AGCN [40]	*Neur.'24*	90.40	95.50	95.50
BlockGCN [13]	*CVPR'24*	90.90	95.40	95.50
EPP-Net [26]	*TRIT'24*	90.20	95.00	–
VA-AR [42]	*AAAI'25*	88.70	93.50	–
DSTAFormer [14]	*TCSVT'26*	90.70	95.20	–
ST-VA-AR [46]	*PR'26*	90.30	95.10	–
Pose2Point (Ours)	–	**91.13**	**96.07**	**95.64**

Table 3
Comparison with state-of-the-art **point cloud-based** methods on NTU RGB+D and NTU RGB+D 120 X-Sub benchmarks.

Method	Source	NTU60 (%)	NTU120 (%)
P4Transformer [25]	*CVPR'21*	90.2	86.4
PSTNet [24]	*ICLR'21*	90.5	87.0
HyperPointNet [47]	*ICME'22*	90.2	83.2
SequentialPointNet [48]	*TII'22*	90.3	83.5
CPR [49]	*AAAI'23*	91.0	–
3DinAction [11]	*CVPR'24*	89.3	–
PRG-Net [50]	*Neur.'25*	91.0	85.4
PvNeXt [51]	*ICLR'25*	89.2	–
EchoNet [52]	*KBS'26*	91.0	84.3
Pose2Point (Ours)	–	**91.13**	**87.34**

3.4. Enhanced backbone with Multi-CLS loss

Pose2Point consists of n P2P blocks, each comprising a UMS module followed by a point cloud backbone. In the lth P2P block, UMS first downsamples the input PPC features to obtain $\hat{\mathcal{P}}^l$, which is then fed into the point cloud backbone for spatiotemporal modeling of PPC. Features extracted from the final block are passed to the classification head to produce the prediction. Most existing point cloud classification methods rely solely on a single classification loss computed at the final output as the supervisory signal. However, using only a single classification loss may limit gradient propagation: in a network with n blocks, gradients from the final loss tend to weaken when reaching earlier blocks, which must process larger and denser PPCs.

To address this, Pose2Point introduce independent heads for the last m blocks. Each head computes a loss from its block's output and back-propagates, enhancing gradient flow and providing direct supervision to intermediate layers. The overall Multi-CLS loss is:

$$\mathcal{L} = \sum_{i=n-m+1}^{n} \mathcal{L}_i, \tag{7}$$

where $\mathcal{L}_i$ denote the loss at block i for $i = n-m+1, \ldots, n$. This strategy significantly improves backbone performance by ensuring that all blocks receive strong and direct training signals.

3.5. PPC-specific training strategy

Training strategies are crucial, and the same applies to our PPC modality, which requires an appropriate training strategy. Common data augmentation techniques in both point cloud–based and pose-based representations include scaling, translation, rotation, and mixup augmentation. Specifically, scaling is used to adjust the overall size of the input, translation shifts the overall structure within a specified range, rotation transforms the data using fixed or random angles, and mixup augmentation generates new data by interpolating between different samples to improve the model's generalization ability. Through a series of ablation experiments, we propose a **PPC data augmentation** method, where random rotation and mixup augmentation are found to be the most effective enhancement techniques for PPC. In addition, since the PPC is generated by randomly selecting subsets from the entire point set, each sampling process introduces a degree of randomness. In order to mitigate the impact of this randomness on the final outcomes and enhance the model's stability, we introduce a **PPC-Ensemble** strategy: we perform multiple inferences to obtain results from different Pose Point Clouds, and then average these results to produce the final output.

4. Experiments

4.1. Datasets

We gauge the performance of our Pose2Point on three widely-used benchmark datasets for pose-based human action recognition: NTU RGB+D [53], NTU RGB+D 120 [54], and Northwestern-UCLA [55].

NTU-RGB+D [53] is a large-scale human action recognition dataset collected in indoor environments, comprising 56,880 pose action sequences. The action samples were performed by 40 volunteers and

Table 4
Ablation results of Pose2Point modules on NTU-RGB+D 120 X-Sub. The full model is compared against versions with individual components removed. ↓ denotes the performance degradation.

Method	Acc (%)	Δ (%)
Pose2Point (Full Model)	**87.34**	–
w/o Transfer2PointCloud	79.37	−7.97
w/o Uniform Mixed Sampling	84.92	−2.42
w/o Multi-CLS Loss	84.45	−2.89
w/o PPC-Specific Strategy	84.81	−2.53
Baseline	74.74	−12.60

Table 5
Performance of Uniform Sampling under different FPS implementations.

FPS	Acc (%)	Latency (ms/sample)	Throughput (samples/s)
CUDA-FPS	86.06	5.47	182.96
Torch-FPS	87.09	6.48	154.21

Table 6
Ablation study for different sampling strategies.

Method	Components			Acc (%)
	Uniform	Mixed	FPS	
FPS	✗	✗	✓	85.31
Uniform Sampling	✓	✗	✓	86.06$^{\uparrow 0.75}$
Mixed Sampling	✗	✓	✓	86.92$^{\uparrow 1.61}$
Uniform Mixed Sampling	✓	✓	✓	**87.34**$^{\uparrow 2.03}$

Table 7
Sensitivity analysis of the sampling ratio ρ in UMS on NTU-RGB+D X-Sub.

Sampling Ratio ρ	Acc (%)
0.30	88.97
0.50	**91.13**
0.70	90.24

Table 8
Ablation study on the number of Multi-CLS loss heads (m). Results are reported under uniform sampling without PPC-Ensemble. Total Params denotes the total number of network parameters.

Count (m)	Acc (%)	Total Params (M)
1	84.38	13.23
2	85.17$^{\uparrow 0.79}$	13.39$^{\uparrow 0.16M}$
3	**86.06**$^{\uparrow 1.68}$	**13.44**$^{\uparrow 0.21M}$
4	85.87$^{\uparrow 1.49}$	13.47$^{\uparrow 0.24M}$

categorized into 60 classes. Each sample contains a single action and at most two subjects, captured simultaneously from three Microsoft Kinect v2 cameras from different views. Two benchmarks are defined: (1) Cross-View (X-View): Training data are collected from cameras positioned at 0° (View 2) and 45° (View 3), while testing data are obtained from a camera at −45° (View 1). (2) Cross-Subject (X-Sub): Data from 20 subjects are used for training, and data from the remaining 20 subjects are reserved for testing.

NTU-RGB+D 120 [54], an extension of NTU-RGB+D, is the largest top-view dataset for action recognition, featuring 114,480 samples across 120 classes and 106 subjects. It introduces 32 distinct setups with diverse locations and backgrounds. Following the official recommendation, we adopt two benchmarks: (1) Cross-Subject (X-Sub), involving 53 subjects for training and 53 for testing; and (2) Cross-Setup (X-Set), where training data are from even-numbered setups and testing data from odd-numbered ones.

Table 9
Ablation study results for Pose Point Cloud data augmentation. The results shown are without PPC-Ensemble.

Data Augmentation Strategies			Acc (%)
Rotation	Mixup	Translation & Scaling	
✗	✗	✗	84.81
✓	✗	✗	85.13$^{\uparrow 0.32}$
✗	✓	✗	85.46$^{\uparrow 0.65}$
✗	✗	✓	84.85$^{\uparrow 0.04}$
✓	✓	✗	**87.07**$^{\uparrow 2.26}$
✓	✓	✓	85.46$^{\uparrow 0.65}$

Northwestern-UCLA (NW-UCLA) [55] contains 1494 video clips captured by three Kinect cameras, covering 10 action classes performed by 10 subjects from multiple viewpoints. Following [55], we train on data from two cameras and test on the remaining one.

4.2. Implementation details

All experiments are conducted on four NVIDIA GeForce RTX 3070 Ti GPUs. The model comprises four P2P blocks. In the UMS module, the number of sampled points is initially set to 1024, and at each subsequent stage, the number of points is reduced by half. The neighborhood size for the KNN operation is fixed at 32. The Multi-CLS loss is applied to the last three blocks. Training is performed using stochastic gradient descent (SGD) with a batch size of 48, an initial learning rate of 0.025, and a weight decay of 1×10^{-4}. For the PPC-specific training strategy, random rotation is applied with a probability of 1 using rotation angles sampled from ([−0.3, 0.3]) radians for each axis, and Mixup is applied with a probability of 0.5. During inference, PPC-Ensemble performs four independent runs and averages their scores to produce the final output. The models are trained for 256 epochs on NTU-RGB+D and NTU-RGB+D 120, and for 128 epochs on NW-UCLA using CUDA-based FPS. This is followed by an additional 5–10 epochs of fine-tuning using Torch-based FPS.

4.3. Comparison with state-of-the-art methods

We compare the proposed P2P method with state-of-the-art approaches, and the comparative results are presented in Tables 1 and 2. Notably, the comparison is performed using **only the *Joint* stream**. This is because the *Joint* stream contains the same 3D coordinates as point clouds, allowing it to be directly converted into a Pose Point Cloud (PPC) representation. In contrast, point cloud-based methods cannot naturally accommodate modalities such as *Bone*, *Joint Motion*, and *Bone Motion*. Therefore, to ensure a fair comparison, we report results using only the *Joint* stream rather than those obtained from multi-stream fusion. It can be observed that our method achieves state-of-the-art performance across all benchmarks of NTU-RGB+D, NTU-RGB+D 120, and NW-UCLA. Notably, *as a pure point cloud-based approach, our method achieves state-of-the-art performance compared with pose-based methods*, demonstrating the feasibility and potential of converting pose sequences into point clouds and processing them within a unified Pose Point Cloud framework.

In addition, we compare our method with recent point cloud-based approaches, as shown in Table 3. Notably, conventional point cloud methods typically take sequential point clouds as input, where each frame is dense and rich in surface geometry.

In contrast, PPC aggregates the joint coordinates of an entire pose sequence into a point cloud, implicitly preserving temporal information while being relatively sparse and lacking surface details compared to conventional dense point clouds. *Despite this lower information density, Pose2Point still outperforms all dense point cloud-based baselines.* This result demonstrates that Pose2Point effectively bridges the gap between pose data and point cloud modeling, further validating the effectiveness of the unified modality.

Table 10
Performance improvement of PPC-Ensemble on different benchmarks. *Baseline* refers to the model without PPC-Ensemble.

Dataset	Benchmark	Acc(%)	
		Baseline	PPC-Ensemble
NTU-RGB+D	X-Sub	90.76	91.13$^{\uparrow 0.37}$
	X-View	95.42	96.07$^{\uparrow 0.65}$
NTU-RGB+D 120	X-Sub	87.07	87.34$^{\uparrow 0.27}$
	X-Set	87.44	88.22$^{\uparrow 0.78}$
NW-UCLA	–	95.10	95.64$^{\uparrow 0.54}$

Table 11
Comparison of parameters and accuracy with point cloud-based HAR methods on NTU-RGB+D X-Sub.

Method	Params (M)	Acc (%)
MeteorNet [10]	17.6	–
P4Transformer [25]	44.1	90.20
PSTNet [24]	20.5	90.50
PST-Transformer [23]	44.2	91.00
Pose2Point	13.4	**91.13**

Table 12
Complexity and accuracy analysis of Pose2Point and its components.

Method	Params (M)	FLOPs (G)	Acc (%)
PointMLP	13.40	15.67	74.74
Pose2Point	13.44$^{\uparrow 0.04}$	15.68$^{\uparrow 0.01}$	87.34$^{\uparrow 12.60}$
- Transfer2PointCloud	0.00	0.00	–
- UMS	0.00	0.01	–
- Feature Extraction	12.57	15.67	–
- Classifier	0.86	0.86	–

4.4. Ablation studies

In this section, ablation studies are conducted to demonstrate the effectiveness of each module in the proposed Pose2Point framework. Unless otherwise specified, experiments are performed on the most challenging benchmark, NTU-RGB+D 120 X-Sub.

Module ablation. Table 4 summarizes the ablation experiment results of each module of the proposed Pose2Point model. The results show that removing the modality transformation module Transfer2PointCloud will reduce the accuracy of NTU-RGB+D 120 X-Sub by 7.97%; removing UMS will cause the accuracy to drop by 2.42%; removing Multi-CLS loss in the backbone network will reduce the accuracy by about 2.89%; removing the PPC-Specific Training Strategy will reduce the accuracy by about 2.53%. Compared with the baseline method, where the input pose data is modeled using PointMLP, the overall accuracy of the P2P model is improved by about 12.60%. These results fully demonstrate the importance of each module we proposed to the overall performance of the model and significantly improve the recognition accuracy.

Sampling Method. We propose a UMS strategy, which consists of two components: uniform sampling and mixed sampling. Uniform sampling is performed on joints along both the temporal and spatial dimensions to preserve the structural integrity of human semantics and capture dynamic variations in motion sequences. Mixed Sampling adopts a staged strategy that combines the efficiency of CUDA-FPS with the sampling stability of Torch-FPS, thereby alleviating the sampling bias introduced by different FPS implementations. As shown in Table 5, different FPS implementations exhibit a clear efficiency–performance trade-off: CUDA-FPS achieves lower inference latency and higher throughput, whereas Torch-FPS provides slightly better recognition accuracy. Furthermore, we compare UMS with other sampling strategies in Table 6, including uniform sampling, mixed sampling, and farthest point sampling. The results show that using FPS alone leads to a 1.73% drop in accuracy compared with UMS, indicating a significant performance degradation. In addition, neither uniform sampling nor mixed sampling alone can achieve the performance obtained by combining both strategies, demonstrating their complementary benefits and validating the effectiveness of the proposed UMS framework. We further analyze the sensitivity of UMS to the sampling ratio ρ. As in Table 7, $\rho = 0.5$ achieves the best accuracy among the evaluated settings, indicating a better balance between topology preservation and sampling efficiency. Therefore, we use $\rho = 0.5$ as the default setting.

Multi-CLS Loss. Table 8 analyzes the effect of the number of Multi-CLS loss heads m. Compared with the single-head setting ($m = 1$), adding auxiliary classification heads consistently improves recognition accuracy. Specifically, increasing m from 1 to 2 improves the accuracy by 0.79%, with only 0.16M additional parameters. The best performance is achieved when $m = 3$, reaching 86.06% and yielding a 1.68% improvement over the baseline, while the total model size only increases from 13.23M to 13.44M. This result indicates that intermediate supervision can effectively enhance feature learning by providing stronger gradient signals to earlier network stages. The auxiliary heads introduce only marginal overhead, adding 0.21M parameters when $m = 3$, and they are disabled during inference so that only the final classification head is used. When m is further increased to 4, the total parameters increase to 13.47M, but the accuracy drops to 85.87%. This suggests that excessive auxiliary supervision may not bring further benefits and could introduce additional optimization constraints. Therefore, we adopt $m = 3$ as the default setting, which provides the best trade-off between performance and computational overhead.

PPC Data Augmentation. An ablation study on PPC data augmentation was conducted to evaluate the effects of three common augmentation techniques: Rotation, Mixup, and Translation & Scaling. As shown in Table 9, each augmentation method contributed to an improvement in accuracy relative to the baseline, but the gain from Translation & Scaling was significantly lower than that from Rotation and Mixup. For combined augmentation strategies, the data augmentation strategy we used, the combination of Rotation and Mixup, led to the best performance, with a gain of +2.26%. It is worth noting that simply stacking all strategies does not yield better results. The accuracy improvement is only +0.65% when all three strategies are combined, which is identical to the effect achieved using only mixup. This indicates that stacking multiple data augmentation strategies does not automatically result in better performance and further confirms the effectiveness of our PPC data augmentation.

PPC-Ensemble. In addition to data augmentation, we also introduced the PPC-Ensemble strategy. Since PPC is generated by randomly selecting a subset from the entire point set, each sampling process will introduce a certain degree of randomness, which may lead to unstable results. In order to mitigate the impact of this randomness and enhance the robustness of the model, we designed the PPC-Ensemble strategy. Table 10 shows that performance improvements were achieved on different benchmarks after adopting the PPC-Ensemble strategy. It can be seen that PPC-Ensemble has achieved significant performance gains on various benchmarks, which proves that the strategy can effectively reduce the negative impact of sampling randomness, and thereby improve the generalization ability and robustness of the entire system in different scenarios.

Efficiency. As shown in Table 11, Pose2Point achieves a better trade-off between parameter efficiency and performance. Table 12 further analyzes the contributions of individual components: compared with the baseline PointMLP model, Pose2Point improves accuracy by 12.60% while incurring negligible additional overhead. Notably, the Transfer2PointCloud and UMS modules introduce no trainable parameters, with most parameters stemming from the baseline's original feature extraction. This confirms that the performance gain mainly comes from the modules proposed in Pose2Point, such as Transfer2PointCloud and UMS. All results are obtained using 64-clip inputs, averaged over 50 independent runs.

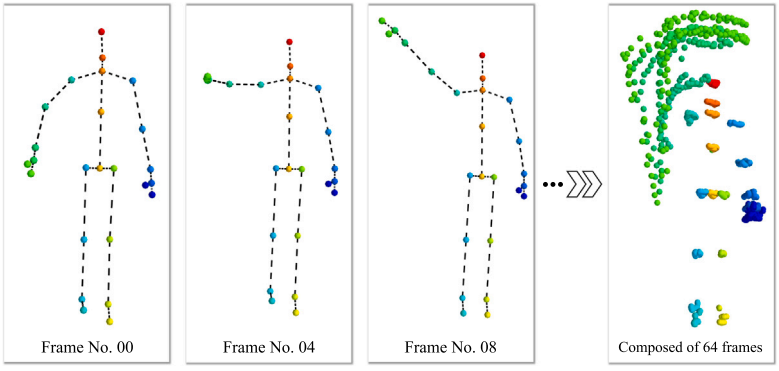


Fig. 3. Pose sequence represented as a static point cloud in 3D space, illustrated with the "hand waving" action.

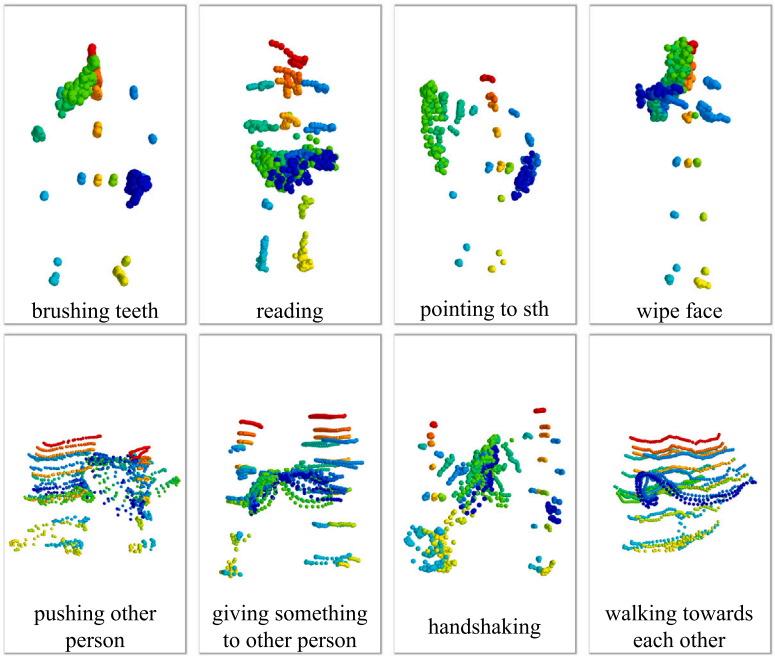


Fig. 4. Point cloud visualization for different actions, where different colors represent different joints.

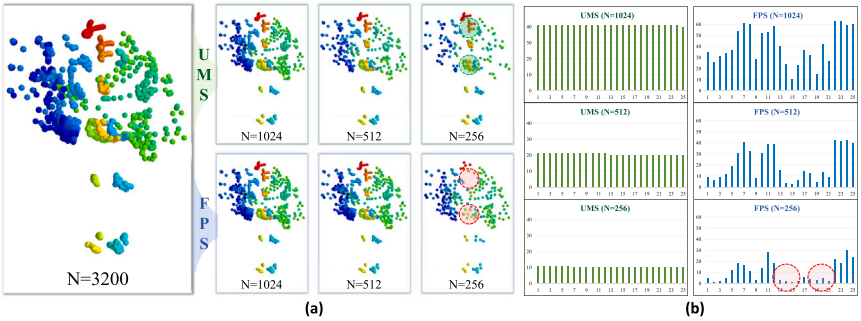


Fig. 5. Comparison of Uniform Manifold Sampling (UMS) and Farthest Point Sampling (FPS) on the "taking off a jacket" action. (a) Visualization of progressive downsampling across stages. (b) Distribution of sampling points over 25 keypoints at each stage. N denotes the total number of points.

4.5. Visualization

Transfer2PointCloud. The original pose data retains 3D spatial coordinate information, and the pose sequence encapsulates temporal information. During the conversion of the pose sequence into a point cloud, the 3D coordinates of each joint are preserved while temporal and joint-specific information are embedded into the feature channels. This allows for an intuitive visualization of the action point cloud based solely on 3D spatial coordinates. In Fig. 3, uses the "hand waving" action as an example to illustrate the transformation from a pose sequence to a point cloud in the 3D space, and Fig. 4 presents representative visualizations of action point clouds for both single-person and two-person actions. The local geometric details of these action point clouds reveal rich motion characteristics.

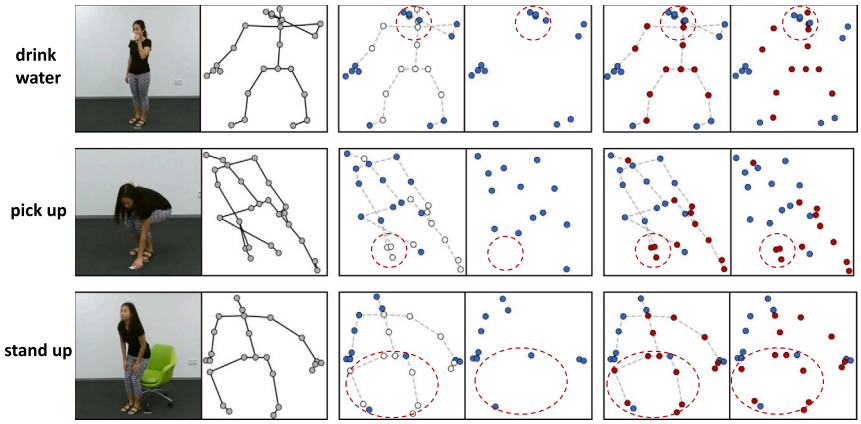


Fig. 6. Visualization comparison of joint retention during point cloud modeling between P2P and the baseline. The first group shows the RGB image and its corresponding pose. The second and third groups present the joint retention results of the baseline and P2P, respectively, where the left images retain the pose topology to clearly illustrate the relationships between joints. Blue indicates joints retained by the baseline, white indicates joints discarded by the baseline, and red indicates the additional joints preserved by P2P compared to the baseline. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

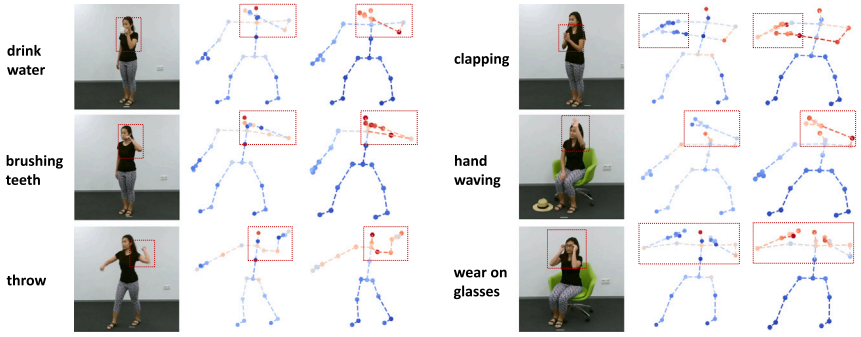


Fig. 7. Attention visualization comparison between topology-based pose modeling and geometry-based point cloud modeling. For each sample, we show the RGB frame, followed by the attention maps produced by the pose-based method (CTR-GCN [32]) and the point cloud-based method, respectively. Color intensity from blue to red indicates increasing importance. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Uniform Mixed Sampling. In hierarchical point cloud feature extraction, the downsampling strategy is critical for preserving semantic information. As illustrated in Fig. 5, although FPS ensures spatial coverage, its farthest-distance criterion may discard semantically important body regions, as highlighted by the red circles in Fig. 5(a) and (b). In contrast, our UMS incorporates pose priors to distribute sampling points evenly across the 25 keypoints, thereby preserving human topological integrity. As shown in Fig. 6, the FPS-based baseline loses hand-related points for "pick up" and lower-limb points for "stand up", weakening essential motion cues and structural representations. Given the inherent sparsity of human joints, such structural disruption hinders effective spatiotemporal modeling. Pose2Point addresses this limitation by retaining critical joints (highlighted in red), enabling more robust representations of fine-grained actions.

Topology-based vs. Geometry-based Attention Based on Fig. 7, we compare attention distributions of a topology-based pose method and a geometry-based point cloud method. The pose method shows scattered attention and fails to focus on key joints, especially in cross-topology actions like clapping or wearing on glasses. In these cases, its responses are confined to local pose chains and miss physically close but topologically distant joints. This occurs because pose methods rely on a fixed topology, requiring multi-hop propagation that dilutes critical features. In contrast, point cloud methods directly capture spatially adjacent joints via neighborhood modeling, leading to more accurate attention. This comparison confirms the advantage of geometric modeling for cross-joint interactions and supports converting pose sequences to point clouds.

Feature Analysis. In Fig. 8, we visualize the feature distributions before the classification layer using t-SNE for four configurations: (a) the baseline (PointMLP), (b) the baseline with the Transfer2PointCloud module, (c) the P2P method without the Transfer2PointCloud module, and (d) our complete P2P method. Under identical experimental settings, the full P2P method (d) yields the most compact intra-class clusters and clearest inter-class separation, with minimal overlap and well-defined class boundaries, demonstrating the strongest discriminative power. In contrast, the baseline (a) exhibits dispersed features with substantial inter-class overlap, highlighting the limitations of directly treating raw pose sequences as point clouds. Introducing the Transfer2PointCloud module (b) improves class separability, while the ablated P2P variant (c) also outperforms the baseline, yet both remain inferior to the complete P2P pipeline (d). Furthermore, the comparison between (b) and (d) shows that our complete P2P method produces more discriminative feature representations, demonstrating the positive impact of the PPC-specific modules and strategy after converting pose sequences into Pose Point Clouds. Overall, these results validate the effectiveness and synergy of the proposed components.

Per-Action Performance. Fig. 9 visually illustrates the accuracy differences across 120 actions in the NTU-RGB+D 120 between our method and the baseline (PointMLP). It is evident that our method outperforms the baseline in every category, demonstrating its superior performance. Notably, for action categories involving complex spatiotemporal dynamics, such as those with subtle hand or body movements, our approach shows significant gains. This indicates that

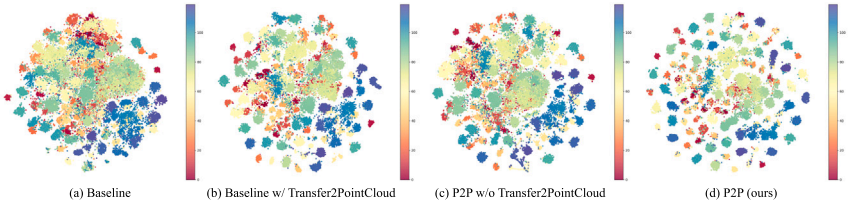


Fig. 8. T-distributed Stochastic Neighbor Embedding (t-SNE) visualization of features. Visualizations are provided for: (a) the baseline model (PointMLP), (b) the baseline model with the Transfer2PointCloud module, (c) the P2P method without the Transfer2PointCloud module, and (d) our proposed P2P method.

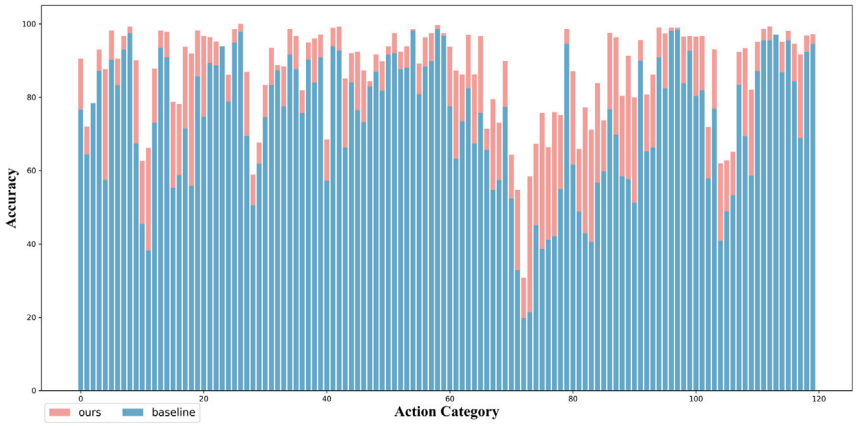


Fig. 9. A bar chart comparing the per-action accuracy of our method and the baseline on the NTU-RGB+D 120 dataset. Our method outperforms the baseline (PointMLP) for every action.

by converting pose sequences into Pose Point Clouds and incorporating our proposed sampling, modeling, and related techniques, P2P can capture fine-grained action features more accurately, thereby substantially improving recognition accuracy.

5. Limitations and future work

Although Pose2Point achieves strong performance by converting pose sequences into PPCs, it still has several limitations. First, PPC quality depends on the accuracy of input poses, and severe occlusion, missing joints, or noisy pose estimation may introduce incomplete or inaccurate structural information. Second, the current framework mainly focuses on the joint stream and has not yet fully exploited complementary modalities such as bone, motion, RGB appearance, and depth cues. In future work, we will extend Pose2Point toward a more general unified multimodal framework. Specifically, we plan to explore converting more modalities into point cloud-like representations and using point clouds as a unified modality for joint modeling. This may enable different sources of information to be represented, aligned, and learned within a single point cloud-based paradigm.

6. Conclusion

We propose a novel data-driven cross-modal unified framework, Pose2Point, which achieves an end-to-end structured conversion from pose sequences to point clouds and constructs a Pose Point Cloud modality that integrates both temporal dynamics and human topology characteristics. Our approach circumvents the challenges of multi-modal feature alignment through the Transfer2PointCloud module, and leverages prior knowledge of human joints by designing a Uniform Mixed Sampling strategy, combined with Multi-CLS loss and a PPC-Specific Training Strategy, significantly enhancing the feature extraction capability of the Pose Point Cloud. The effectiveness of the proposed Pose2Point is validated on the NTU RGB+D, NTU RGB+D 120, and NW-UCLA datasets, demonstrating that a pure point cloud approach can achieve state-of-the-art performance comparable to advanced pose-based models. This work provides a simple and efficient solution for cross-modal research on poses and point clouds, substantially improving the system's generalization and robustness in complex scenarios.

CRediT authorship contribution statement

Jiaying Lin: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Jiajun Wen:** Writing – original draft, Validation, Investigation, Formal analysis. **Weiwei Wan:** Writing – review & editing. **Mengyuan Liu:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

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.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 62473007), Guangdong Provincial Key Laboratory of Ultra High Definition Immersive Media Technology (No. 2024B1212010006), Shenzhen Innovation in Science and Technology Foundation for The Excellent Youth Scholars (No. RCYX20231211090248064).

Data availability

Data will be made available on request.

References

[1] H. Qiu, B. Hou, Multi-grained clip focus for skeleton-based action recognition, Pattern Recognit. 148 (2024) 110188.

[2] M. Liu, H. Liu, C. Chen, Enhanced skeleton visualization for view invariant human action recognition, Pattern Recognit. 68 (2017) 346–362.

[3] F. Yu, Y. Liu, Z. Wu, M. Tan, J. Yu, Adaptive gait training of a lower limb rehabilitation robot based on human–robot interaction force measurement, Cyborg Bionic Syst. 5 (2024) 0115.

[4] Y. Yu, Z. Zhou, Y. Xu, C. Chen, W. Guo, X. Sheng, Toward hand gesture recognition using a channel-wise cumulative spike train image-driven model, Cyborg Bionic Syst. 6 (2025) 0219.

[5] Y. Li, R. Xia, X. Liu, Learning shape and motion representations for view invariant skeleton-based action recognition, Pattern Recognit. 103 (2020) 107293.

[6] Z. Wu, Y. Ding, L. Wan, T. Li, F. Nian, Local and global self-attention enhanced graph convolutional network for skeleton-based action recognition, Pattern Recognit. 159 (2025) 111106.

[7] Z. Li, Y. Wang, X. Du, C. Wang, R. Koch, M. Liu, Asmnet: Action and style-conditioned motion generative network for 3D human motion generation, Cyborg Bionic Syst. 5 (2024) 0090.

[8] Y. Zhang, Z. Chen, T. Xu, J. Zhao, S. Mi, X. Geng, M.-L. Zhang, Temporal segment dropout for human action video recognition, Pattern Recognit. 146 (2024) 109985.

[9] J. Zhang, W. Li, P.O. Ogunbona, P. Wang, C. Tang, RGB-d-based action recognition datasets: A survey, Pattern Recognit. 60 (2016) 86–105.

[10] X. Liu, M. Yan, J. Bohg, Meteornet: Deep learning on dynamic 3d point cloud sequences, in: IEEE/CVF International Conference on Computer Vision, 2019, pp. 9246–9255.

[11] Y. Ben-Shabat, O. Shrout, S. Gould, 3Dinaction: Understanding human actions in 3d point clouds, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 19978–19987.

[12] L. Z., Z. H., C. Z., W. Z., O. W., Disentangling and unifying graph convolutions for skeleton-based action recognition, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 143–152.

[13] Y. Zhou, X. Yan, Z.-Q. Cheng, Y. Yan, Q. Dai, X.-S. Hua, Blockgcn: Redefine topology awareness for skeleton-based action recognition, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 2049–2058.

[14] Y. Zhu, G. Liu, Dynamic prompting spatial temporal actor transformer for fine-grained skeleton-based action recognition, IEEE Trans. Circuits Syst. Video Technol. (2026).

[15] B. Ren, M. Liu, R. Ding, H. Liu, A survey on 3d skeleton-based action recognition using learning method, Cyborg Bionic Syst. 5 (2024) 0100.

[16] C.R. Qi, H. Su, K. Mo, L.J. Guibas, Pointnet: Deep learning on point sets for 3d classification and segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.

[17] X. Ma, C. Qin, H. You, H. Ran, Y. Fu, Rethinking network design and local geometry in point cloud: A simple residual MLP framework, Int. Conf. Learn. Represent. (2022).

[18] J. Liu, Z. Jiang, X. Xu, W. Li, M. Liu, H. Liu, Multi-robot collaborative complex indoor scene segmentation via multiplex interactive learning, CAAI Trans. Intell. Technol. 10 (6) (2025) 1646–1660.

[19] J. Lin, S. Wu, J. Liu, C. Wang, M. Liu, DAP: Doppler-aware point network for heterogeneous mmwave action recognition, in: Proceedings of the European Conference on Computer Vision, 2026.

[20] L. Wu, J. Hou, L. Song, Y. Xu, 3D shape completion on unseen categories: A weakly-supervised approach, IEEE Trans. Vis. Comput. Graphics 31 (9) (2024) 5547–5561.

[21] L. Wu, X. Cheng, Y. Xu, H. Zeng, J. Hou, Unsupervised 3d point cloud completion via multi-view adversarial learning, IEEE Trans. Vis. Comput. Graphics (2025).

[22] X. Cheng, L. Wu, Z. Wang, J. Hou, J. Wen, Y. Xu, Pvnet: Point-voxel interaction lidar scene upsampling via diffusion models, IEEE Trans. Image Process. (2025).

[23] H. Fan, Y. Yang, M. Kankanhalli, Point spatio-temporal transformer networks for point cloud video modeling, IEEE Trans. Pattern Anal. Mach. Intell. 45 (2) (2022) 2181–2192.

[24] H. Fan, X. Yu, Y. Ding, Y. Yang, M. Kankanhalli, PSTNet: Point spatio-temporal convolution on point cloud sequences, in: International Conference on Learning Representations, 2022.

[25] H. Fan, Y. Yang, M. Kankanhalli, Point 4d transformer networks for spatio-temporal modeling in point cloud videos, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 14204–14213.

[26] J. Liu, R. Ding, Y. Wen, N. Dai, F. Meng, F.-L. Zhang, S. Zhao, M. Liu, Explore human parsing modality for action recognition, CAAI Trans. Intell. Technol. 9 (6) (2024) 1623–1633.

[27] B. Yin, J. Lin, J. Wen, Y. Li, J. Liu, Y. Wang, M. Liu, Recognizing skeleton-based actions as points, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2025, pp. 21582–21589.

[28] Y. Su, G. Lin, J. Zhu, Q. Wu, Human interaction learning on 3d skeleton point clouds for video violence recognition, in: European Conference on Computer Vision, Springer, 2020, pp. 74–90.

[29] S. Yang, J. Liu, S. Lu, E.M. Hwa, Y. Hu, A.C. Kot, Self-supervised 3D action representation learning with skeleton cloud colorization, IEEE Trans. Pattern Anal. Mach. Intell. 46 (1) (2023) 509–524.

[30] L. Shi, Y. Zhang, J. Cheng, H. Lu, Two-stream adaptive graph convolutional networks for skeleton-based action recognition, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 12026–12035.

[31] C. K., Z. Y., H. X., C. W., C. J., L. H., Skeleton-based action recognition with shift graph convolutional network, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 183–192.

[32] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, W. Hu, Channel-wise topology refinement graph convolution for skeleton-based action recognition, in: IEEE International Conference on Computer Vision, 2021, pp. 13359–13368.

[33] C. Z., L. S., Y. B., L. Q., L. H., Multi-scale spatial temporal graph convolutional network for skeleton-based action recognition, in: AAAI Conference on Artificial Intelligence, 2021.

[34] C. H., H. M., C. S., L. S., H. Q., R. K., Infogcn: Representation learning for human skeleton-based action recognition, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 20186–20196.

[35] H. Zhou, Q. Liu, Y. Wang, Learning discriminative representations for skeleton based action recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Vancouver, Canada, 2023, pp. 10608–10617.

[36] J. Lee, M. Lee, D. Lee, S. Lee, Hierarchically decomposed graph convolutional networks for skeleton-based action recognition, in: IEEE International Conference on Computer Vision, 2023, pp. 10444–10453.

[37] X. Huang, H. Zhou, J. Wang, H. Feng, J. Han, E. Ding, J. Wang, X. Wang, W. Liu, B. Feng, Graph contrastive learning for skeleton-based action recognition, in: International Conference on Learning Representations, 2023.

[38] W. Xin, Q. Miao, Y. Liu, R. Liu, C.-M. Pun, C. Shi, Skeleton mixformer: Multivariate topology representation for skeleton-based action recognition, in: ACM International Conference on Multimedia, 2023, pp. 2211–2220.

[39] T. Zhuang, Z. Qin, Y. Ding, Z. Qin, J. Geng, Y. Liu, K.-K.R. Choo, DSDC-GCN: Decoupled static-dynamic co-occurrence graph convolutional networks for skeleton-based action recognition, IEEE Trans. Circuits Syst. Video Technol. (2024).

[40] H. Tian, Y. Zhang, H. Wu, X. Ma, Y. Li, Multi-scale sampling attention graph convolutional networks for skeleton-based action recognition, Neurocomputing 597 (2024) 128086.

[41] H. Chen, Y. Yang, Y. Lyu, Skeleton-based action recognition with non-linear dependency modeling and hilbert-schmidt independence criterion, in: AAAI Conference on Artificial Intelligence, vol. 39, (2) 2025, pp. 2043–2051.

[42] J. Wei, L. Qin, B. Yu, T. Zou, C. Yan, D. Xiao, Y. Yu, L. Yang, K. Li, J. Liu, Va-ar: Learning velocity-aware action representations with mixture of window attention, in: AAAI Conference on Artificial Intelligence, vol. 39, (8) 2025, pp. 8286–8294.

[43] H. Xu, Y. Gao, Z. Hui, J. Li, X. Gao, Language knowledge-assisted representation learning for skeleton-based action recognition, IEEE Trans. Multimed. (2025).

[44] R. Zhou, Y. Wu, J. Wen, C. Gao, F. Liu, Z. Lai, L. Shen, Gamba: Mamba-based graph convolutional network with dynamic graph topology learning for action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2026, pp. 6065–6074.

[45] J. Liu, X. Wang, C. Wang, Y. Gao, M. Liu, Temporal decoupling graph convolutional network for skeleton-based gesture recognition, IEEE Trans. Multimed. 26 (2023) 811–823.

[46] J. Wei, B. Yu, K. Li, L. Yang, D. Xiao, J. Liu, ST-va-AR: Learning velocity-aware action representations with mixture of spatiotemporal attention, Pattern Recognit. (2026) 113200.

[47] X. Li, Q. Huang, T. Yang, Q. Wu, Hyperpointnet for point cloud sequence-based 3D human action recognition, in: IEEE International Conference on Multimedia and Expo, IEEE, 2022, pp. 1–6.

[48] X. Li, Q. Huang, Z. Wang, T. Yang, Z. Hou, Z. Miao, Real-time 3-d human action recognition based on hyperpoint sequence, IEEE Trans. Ind. Informatics 19 (8) (2022) 8933–8942.

[49] X. Sheng, Z. Shen, G. Xiao, Contrastive predictive autoencoders for dynamic point cloud self-supervised learning, in: AAAI Conference on Artificial Intelligence, vol. 37, (8) 2023, pp. 9802–9810.

[50] Y. Du, Z. Hou, E. Lin, X. Li, J. Liang, X. Zhou, PRG-net: Point relationship-guided network for 3D human action recognition, Neurocomputing 635 (2025) 130015.

[51] J. Wang, T. Xu, L. Ding, X. Zhang, L. Bai, J. Li, Pvnext: Rethinking network design and temporal motion for point cloud video recognition, in: International Conference on Learning Representations, 2025, pp. 102538–102554.

[52] G. Huang, Z. Hou, X. Li, J. Liang, X. Zhou, EchoNet: A hierarchical collaborative network for point cloud-based 3D action recognition, Knowl.-Based Syst. (2026) 115257.

[53] A. Shahroudy, J. Liu, T.-T. Ng, G. Wang, Ntu rgb+ d: A large scale dataset for 3d human activity analysis, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016, pp. 1010–1019.

[54] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, A.C. Kot, Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding, IEEE Trans. Pattern Anal. Mach. Intell. 42 (10) (2019) 2684–2701.

[55] J. Wang, X. Nie, Y. Xia, Y. Wu, S.-C. Zhu, Cross-view action modeling, learning and recognition, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2014, pp. 2649–2656.