2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Rethinking 3D Geometric Object Features for Enhancing Skeleton-based Action Recognition

Yuankai Wu[1], Chi Wang[1], Driton Salihu[1], Constantin Patsch[1], Marsil Zakour[1], Eckehard Steinbach[1]

***Abstract*— Human action recognition is crucial for intelligent robots, especially in the realm of human-robot collaboration research. Recent advancements in human pose estimation algorithms have shifted the focus of action recognition towards skeleton-based models, which exhibit robustness to changes in background and illumination. However, many state-of-the-art action recognition models rely on 2D skeleton data, neglecting object features. This limitation becomes obvious in complex scenarios where human interactions with objects are crucial, potentially compromising the reliability of assistive robots in understanding human behavior in their environment. To address this issue, we propose a method that effectively integrates 3D geometric object features into skeleton data using graph convolutional neural networks (GCNs). In addition to analyzing the effectiveness of information from different dimensions such as object center position, category, translation, and rotation, we explore various adjacency matrix designs for graph networks. Our model performance is evaluated on two challenging datasets: IKEA ASM and Bimanual Actions. The results demonstrate a significant improvement in action recognition by integrating object features into skeleton-based models. Specifically, on the IKEA-ASM dataset, our approach achieves a frame-wise Top-1 score improvement of 10.8% and an average F1@k improvement of 13.3%, while on the Bimanual Actions dataset, it achieves a frame-wise Top-1 score improvement of 11.4% and an average F1@k improvement of 5.3%, with negligible increases in model complexity.**

I. INTRODUCTION

Human action recognition is a field within computer vision, dedicated to the creation of algorithms and systems aimed at comprehending human activities from diverse forms of data [1], [2]. This field has notable significance in various domains, such as video surveillance [3], healthcare [4], and assistive robotics [5], where understanding human actions can be critical to effective operation. Unlike recognizing and analyzing human activities exclusively in two-dimensional dimensions as [6], [7], a robotic perception system is usually built in three-dimensional space. It can thus lead the way in providing solutions for robot perception systems in the field of assistive robotics [8], [9].

From a modeling perspective, human action recognition methods typically fall into two categories: One approach involves directly processing the image and classifying the action [10], [11]. The other approach entails extracting human pose information using a skeleton extractor and then utilizing this information to classify the action [12], [13], [14]. Additionally, multiple data modalities such as depth images and optical flow are fused in action recognition alongside RGB images [15]. However, these approaches require a large volume of data and face challenges in extracting meaningful action-related features [16]. Furthermore, when trained on 2D images, models tend to overfit and become highly sensitive to changes in background or illumination [17]. In response to the limitations of methods using images as input, there has been a shift towards algorithms utilizing the 2D skeleton as input. Compared to 2D images, the skeleton-based approach offers two significant advantages: robustness and compactness. Skeleton data, derived from 2D images, describe the positions of human joints and do not contain background information, making them resilient to changes in background and illumination [18]. Moreover, skeleton data are compact and concise, requiring significantly less data during training compared to methods using 2D images, thus leading to substantial savings in computational resources.

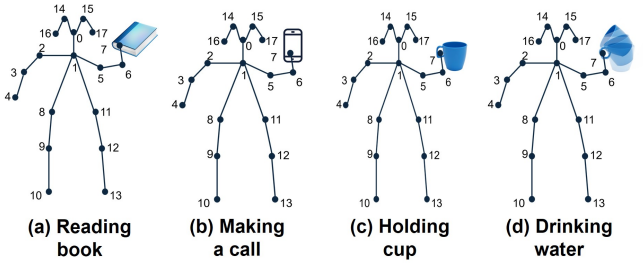


Fig. 1: Different actions with the same skeleton data. In the case of having the same skeleton information, different actions are generated due to the different interactive objects (a,b,c). While interacting with the same object, different actions can also be characterized due to the features of object rotation or translation (c,d).

Recent advancements in skeleton-based action recognition models have achieved remarkable success [19], [20]. Much research effort has been devoted to extracting unique spatial and temporal features from skeleton data [21], [22], [23]. However, a significant limitation of this research has been the fixation skeleton data alone, disregarding the potential contributions of object features to action recognition. As illustrated in Figure 1, scenarios with identical skeleton data may exhibit distinct action categories. Moreover, when an individual interacts with the same object, the movement details of the object can furnish crucial features for distinguishing between different action classifications.

[1]Authors are with Technical University of Munich / School of Computation, Information and Technology / Chair of Media Technology and Munich Institute of Robotics and Machine Intelligence, Munich, Germany. {yuankai.wu, chi.wang, driton.salihu, constantin.patsch, marsil.zakour, eckehard.steinbach}@tum.de

The approach described in this paper addresses the following two main challenges: object feature extraction and feature fusion. For object feature extraction, we utilize the derived object detection results from the dataset to obtain the object category and position. In addition to this, we use object registration to obtain information about the movement of objects in 3D space. In order to improve the generalization of object registration in our method, we do not use current deep learning methods. Instead, we construct a three-stage generic approach of detecting objects, extracting object point clouds, and performing point cloud registration. For geometric object feature fusion, we proposed two strategies: 1) concatenation of skeleton and object features and 2) addition of extra object features nodes. And we evaluate our proposed approach on two widely used challenging datasets. The contributions of our work are summarized as follows:

1) Our work introduces techniques for integrating geometric object information with skeleton-based action recognition, offering a universally adaptable approach compatible with the current state-of-the-art (SOTA) skeleton-based action recognition methods.

2) Our method leverages both 2D detection and depth stream to acquire 3D geometric object information. We propose a generalized methodology for obtaining object registration details based on 2D object detection. Additionally, we present various approaches for merging geometric object information with skeleton information. Furthermore, we conduct a comprehensive ablation study to analyze and evaluate these methods.

3) We assess the performance of our approach on two challenging and intricate datasets: IKEA Assembly [24] and Bimanual Actions [25]. Experimental results demonstrate that incorporating geometric object features leads to significant improvements in frame-wise Top-1 score and F1@k score on both datasets. Moreover, the increase in Floating Point Operations Per Second (FLOPS) due to model complexity is negligible.

II. Related Work

A. Skeleton-based action recognition

Skeleton-based action recognition, utilizing human skeleton information stands as the most popular method in the field. Early works employed 3D convolutional neural networks [26] or LSTM-based [27] methods to process skeleton data. Tran et al. [28] presented the C3D model, which applied 3D convolutions to video data and yielded promising results on benchmark datasets. Zhang et al. [29] proposed an LSTM-based framework integrating spatial and temporal information from skeleton sequences. Graph convolutional networks (GCNs) [30] have recently gained popularity for analyzing skeleton data due to their ability to exploit the inherent structure and relational information. Yan et al. [12] pioneered the integration of GCNs into action recognition with the Spatial-Temporal Graph Convolutional Network (ST-GCN), incorporating 2D spatial information and temporal dynamics of skeleton sequences. Shi et al. [13] introduced a two-stream adaptive GCN, including a joint stream and a bone stream, enhancing recognition accuracy. Data-driven attention mechanisms are integrated to adapt to different graph structures and data samples. Moreover, existing skeleton-based GCNs often suffer from complexity, leading to inefficiencies in training and inference, especially with large-scale datasets. To address this, Song et al. [14] devised a compound scaling strategy, expanding model width and depth simultaneously, resulting in efficient GCN baselines with high accuracies and fewer parameters. However, none of the aforementioned methods account for the contributions of object features within the scene. In our proposed approach, we leverage the geometric information of objects in the 3D scene to augment the performance of existing state-of-the-art (SOTA) methods.

B. Object registration

Object registration is a fundamental task in computer vision, involving the alignment of multiple instances of a scene captured at different times or by different sensors. This process integrates diverse datasets into a unified system, beneficial for tasks such as image stitching, object recognition, and point cloud registration. Previous works have explored various approaches in this domain. In previous work, Guo et al. introduced a shape-growing-based multi-view registration algorithm, initialized with a selected range image, and iteratively updated by conducting pairwise registration between itself and the input range images[31]. In recent years, instance-level 6D pose estimation by Wang et al. [32] focuses on estimating the pose of known objects using color and depth information. In SGPCR [33], Salihu et al. introduced a rotation-equivariant representation, enabling efficient object registration of noisy instances. However, existing learning-based registration methods rely on extensive registration label data, rendering them impractical due to the time-consuming nature of acquiring 3D registration labels. As the objects required for object registration are typically absent in existing datasets, we introduce a generic method based on 2D object detection to enhance generalization.

III. Fusing 3D Geometric Object Features into Skeleton-based Graph Neural Networks

We present our effective framework for integrating geometric object features with skeleton-based graph convolutional networks (GCNs). This challenge entails determining the optimal approach to combine these diverse data types in a manner that enhances GCN performance and enables the network to effectively capture complex spatial and geometric relationships. Figure 2 illustrates our feature fusion module, where we explore two strategies for integrating the various data: the concatenation method and the object nodes method. The feature fusion process incorporates the 3D skeleton feature, object position, category, and quaternion into a compact format conducive to the model training process. In addition to this, we have constructed various node connections for the graphs composed of humans and objects. This facilitates the construction of the adjacency matrix in GCN and analyzes the most efficient feature update mechanism.

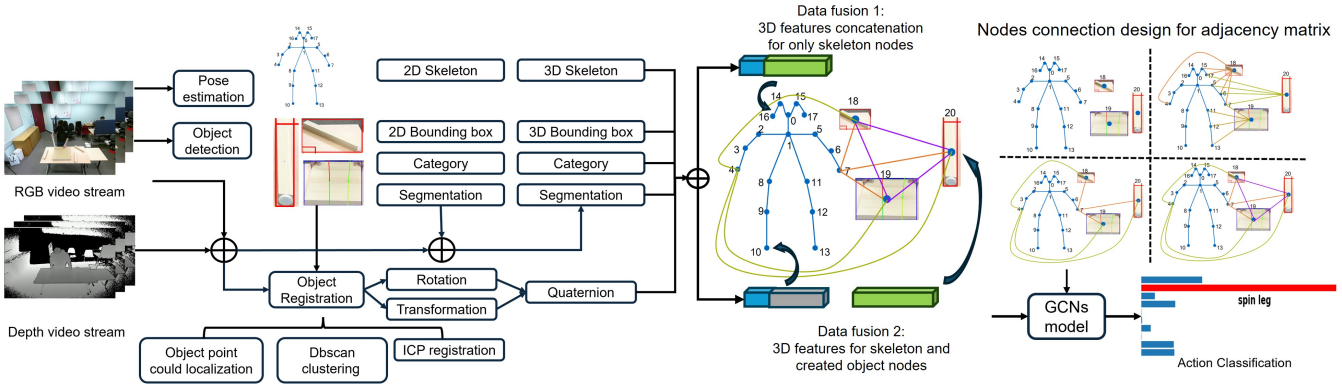


Fig. 2: Overview of our proposed approach. The skeleton and object information is obtained by using a pose estimator and object detector. After that, it is combined with the depth stream to get the 3D skeleton information and the object point cloud. The object point cloud is then registered to finally get the low dimensionality quaternion information. The graph convolutional networks is updated with different data fusion strategies designed by us to make the final action prediction.

A. *Feature extraction and generic object registration*

Since the working environment of the assistive robots perform is a three-dimensional space, our proposed method is better adapted to build a more accurate human activity perception system for the assistive robots. The extraction and processing of 3D geometric features form the core of our proposed method. We will introduce the details below.

1) ***2D geometric object features extraction****:* Our process commences with the input of an RGB video stream into object detection utilizing YOLOv5 [34], and human pose estimation employing Openpose [35]. This stage is tasked with extracting object features, encompassing bounding boxes (which determine the object location in the image), category information (which identifies the type of object), and segmentation (which isolates the object from its surrounding context using k-means [36], k = 2). Within the process, after having detected the object by object detector, we use the k-means method to distinguish the main object from the background. Human poses are denoted as $\mathbf{S}^j$, where j indicates the body joints. We utilize the location and category to formulate a feature vector for object obj:

$$\mathbf{f}_{obj} = (x_{obj}, y_{obj}, Category_{obj}), \tag{1}$$

where x_{obj}, y_{obj}, and $Category_{obj}$ denote object location and category. The segmentation information aids in object registration to segregate object and context details. The extracted information allows for a more intuitive representation of the semantic features of objects and human beings in the video, comparing to the direct use of each frame information in the video. And it reduces a lot of computational complexity for the model inference.

2) ***Converting 2D object geometric features to 3D object geometric features****:* After extracting the 2D features as described above, we obtain the 3D human pose and 3D object positions by combining the detected human pose information with the depth stream. The segmented object features and the depth stream are then merged to create a point cloud of the objects. Utilizing DBSCAN clustering [37] on the reconstructed object point cloud helps eliminate outliers and identify the largest cluster as the object body. Subsequently, we employ Point-to-Point Iterative Closest Point (ICP) [38] to determine the transformation matrix of the objects. In this case, we can obtain more information about the movement of the object in three-dimensional space, providing more features for the human action recognition model. For the correspondence set $\mathcal{K} = \mathbf{p}, \mathbf{q}$ from the target point cloud $\mathbf{P}$, the mathematical formula for the Point-to-Point ICP algorithm is expressed as follows:

$$E(\mathbf{T}) = \sum_{(\mathbf{p},\,\mathbf{q})\in\mathcal{K}} ||\mathbf{p} - \mathbf{Tq}||^2, \tag{2}$$

where $\mathbf{T}$ represents the 4×4 transformation matrix.

Due to the high dimensionality of the transformation matrix, it may be redundant to effectively represent object features. To address this issue, our aim is to simplify the matrix and make it more compact. This motivates us to convert the transformation matrix into a quaternion. Essentially, a quaternion can describe the same information as the rotation matrix. Quaternions and rotation matrices are interchangeable, providing flexibility to choose the format that best suits our data fusion requirements. The quaternion $\mathbf{q}$ can be expressed as:

$$\mathbf{q} = (q_w, q_x, q_y, q_z), where \tag{3}$$

$$\begin{aligned} q_w &= 0.5 \times \sqrt{1 + n_{11} + n_{22} + n_{33}} \\ q_x &= (n_{32} - n_{23})/(4 \times q_w) \\ q_y &= (n_{13} - n_{31})/(4 \times q_w) \\ q_z &= (n_{21} - n_{12})/(4 \times q_w) \end{aligned} \tag{4}$$

where n_{ij} corresponds to the elements in matrix $\mathbf{T}$ from Equation 2. At this stage, we have acquired the 3D skeleton features denoted as $\mathbf{S}^J$, 3D object position and category features $\mathbf{f}_{obj}$, and the quaternion $\mathbf{q}_{obj}$. Our next step involves fusing these features to obtain a comprehensive feature representation.

3) Data fusion strategies: 1) 3D features concatenation for only skeleton nodes. In this strategy, we concate the extracted object features directly behind the features of the skeleton (features represented by the same vector). Given the skeleton data $\mathbf{S} \in \mathbb{R}^{J\times 2}$ or $\mathbf{S} \in \mathbb{R}^{J\times 3}$ (3D), object position and category features $\mathbf{F} \in \mathbb{R}^{K\times 3}$ or $\mathbf{F} \in \mathbb{R}^{K\times 4}$ (3D), quaternions $\mathbf{Q} \in \mathbb{R}^{K\times 4}$, we can concatenate them to obtain new geometric input features $\mathbf{G} \in \mathbb{R}^{J\times 9}$ or $\mathbf{G} \in \mathbb{R}^{J\times 11}$ (3D). K denotes the number of objects detected in the current frame, and J is the number of joints of the skeleton data. 2) 3D features for skeleton and created object nodes. Under this approach, we design new object nodes and connections between skeleton and object nodes (features represented by the different vectors). The geometric features $\mathbf{G}$ is settled as a fix dimension $\mathbf{G} \in \mathbb{R}^{(J+K)\times 7}$ or $\mathbf{G} \in \mathbb{R}^{(J+K)\times 8}$ (3D).

B. Graph construction and adjacency matrix design

By incorporating fused data, we can construct a skeleton graph denoted as $\mathcal{G}t = (\mathcal{V}_{jt}, \mathcal{E}_{jt})$ at a given time step t. Here, $\mathcal{V}_{jt}$ represents the set of joints and $\mathcal{E}_{jt}$ represents the set of edges connecting these joints. In our approach, $\mathcal{V}_{jt}$ = $\mathbf{G}$, which we extracted before. To integrate object data into our skeleton graph, we introduce additional object nodes. These nodes represent the objects in the video frame at a given time step t and are denoted by the set $\mathcal{V}_{ot}$. By incorporating these additional nodes and edges into our existing graph, we obtain a joint-object graph. Furthermore, we define the graph structure using an adjacency matrix A. With the introduction of extra nodes, the corresponding adjacency matrix also requires updating. Thus, the matrix now represents both the original skeleton graph and the newly added object nodes. This process can be summarized as follows:

$$\mathcal{G}t = (\mathcal{V}_t, \mathcal{E}_t), \tag{5}$$

$$\mathcal{V}_t = \mathcal{V}_{jt} \cup \mathcal{V}_{ot}, \quad \mathcal{E}_t = \mathcal{E}_{jt} \cup \mathcal{E}_{ot}, \tag{6}$$

$$A \in \mathbb{R}^{N\times N} \rightarrow A \in \mathbb{R}^{(N+O)\times(N+O)}, \tag{7}$$

where O is the number of objects added to the graph, and N corresponds to the total number of joints within each frame.

Additionally, we introduce a trainable mask to the adjacency matrix to help the model learn hidden relations that the original matrix did not define. Consequently, the feature inference is reformulated as:

$$\mathbf{f}_{out} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I} + \mathbf{M}^{l^{th}})\mathbf{D}^{\frac{1}{2}}\mathbf{G}^{l^{th}}\mathbf{W}^{l^{th}}, \tag{8}$$

where I is the identity matrix, $D_{ii} = \sum_j A_{ij}$, $M^{i^{th}}$ represents the trainable mask at the l^{th}layer, and $W^{l^{th}}$ is the trainable weight matrix.

Compared to the adjacency matrix without the trainable mask, the masked adjacency matrix learns hidden connections between joints. For instance, while two hands may not be directly connected, in tasks like assembling a table, they may work jointly and interact with each other (e.g., picking up a leg from another hand). By incorporating this trainable mask, the model can capture these hidden connections.

Finally, at the end of the model, the feature matrix passes through an adaptive pooling layer to ensure uniform output size. The final output after a fully connected layer is $X^{fc} \in \mathbb{R}^{C_{class}\times 1}$, where C_{class} represents the number of actions. The label for this video clip is then determined as:

$$Label = Map(argmax(\mathbf{X}^{fc})), \tag{9}$$

IV. Experimental results

We evaluate the proposed method on two challenging datasets: IKEA ASM Dataset [24] and Bimanual Action Dataset [25]. Additionally, we conduct an ablation study to evaluate and analyze the impact of individual object features and various node connections (adjacency matrix).

A. Dataset

The IKEA Assembly (IKEA ASM) dataset stands out as a comprehensive resource tailored for furniture assembly tasks, including multi-modal attributes such as depth stream and human pose data. To ensure broad applicability and resilience, recordings are conducted across five distinct environments. Giving 33 diverse object-related atomic actions, this dataset offers us opportunities to enhance skeleton-based approaches with object information.

The Bimanual Action Dataset serves as a valuable repository for bimanual actions, comprising 540 RGB-D videos, including specific object-related tasks like cooking. With a repertoire of 14 distinct actions and 12 interactive objects, this dataset facilitates the association of objects with actions. Furthermore, the inclusion of RGB and depth streams aligns seamlessly with the requirements of our approach.

B. Experimental setup

Our model training and evaluation are conducted on a single GeForce GTX 1080TI. We retrain and evaluate our proposed method against three prominent skeleton-based state-of-the-art models: ST-GCN [12], 2s-AGCN [13], and EGCN[14]. We employed the SGD optimizer for optimization. For STGCN, training comprised 80 epochs, initializing with a learning rate of 0.01. Meanwhile, 2s-AGCN underwent training for 100 epochs, with an initial learning rate of 0.1. For EGCN, training spanned 150 epochs, commencing with a learning rate of 0.1. To prevent resource waste and mitigate overfitting, we implemented an early-stopping strategy across all model training sessions, utilizing a tolerance score of 8. Additionally, a warm-up strategy employing the cosine function was applied to regulate the learning rate.

C. Ablation studies

1) Effectiveness of adding geometric object features: We conducted experiments to assess the impact of integrating geometric object features with skeleton data. Three models are evaluated with feature concatenation and without feature concatenation (only skeleton) as inputs. The results shown in Table I demonstrate that upon integrating object features (center location, category, and quaternion), the Top-1 accuracy of the SOTA models increased by a maximum of

TABLE I: Comparison of model performance between only skeleton data (w/o Concat) and after concatenating object data (w Concat) on IKEA ASM.

Model	Inputs	Top-1	F1@10%	F1@25%	F1@50%
STGCN	w/o Concat	52.06	0.41	0.35	0.23
	w Concat	**59.89**	0.46	0.40	0.27
AGCN	w/o Concat	55.54	0.42	0.36	0.24
	w Concat	**63.20**	0.51	0.45	0.24
EGCN	w/o Concat	56.83	0.45	0.39	0.25
	w Concat	**65.14**	0.54	0.49	0.33

TABLE II: Comparison of model complexity only skeleton data (w/o Concat) and after concatenating object data (w Concat) on IKEA ASM.

Model	Inputs	Params	FLOPS
STGCN	w/o Concat	3.08 M	1.8733 G
	w Concat	3.08 M	1.8809 G
AGCN	w/o Concat	3.45 M	2.1082 G
	w Concat	3.45 M	2.1103 G
EGCN	w/o Concat	3.38 M	2.0039 G
	w Concat	3.38 M	2.0113 G

8.32%. Moreover, there was a notable improvement in the F1@k scores, indicating enhanced precision and recall of the models. Importantly, as shown in Table II, integrating object features did not increase the number of parameters, as there were no changes made to the architecture or the introduction of additional layers in these three SOTA models. Furthermore, the increase in Floating Point Operations Per Second (FLOPS) is only 0.4% compared to models without concatenated object features.

2) Data fusion analysis: The efficacy of data fusion strategies significantly influences model performance, with an effective fusion approach capable of reducing computations and data redundancy. We introduce two data fusion approaches. One of the methods is to concatenate the geometric object features directly onto the end of the skeleton features. Another method is to first create new object nodes and then connect them to all skeletal nodes. As shown in Table III, the object nodes approach outperforms the concatenation method across all metrics. This superiority is particularly obvious in the F1@k scores.

3) Object-joints connection strategy annalysis: We compare different connection strategies between objects and joints, as shown in Figure 2. As results are shown in Table IV, strategy **OH** demonstrates superior performance compared to the other strategies, where objects are connected to both hand joints. This suggests that incorporating interaction information between objects and both hand joints can enhance overall model performance. Strategy **OOH**, which involves connecting objects to two hand joints and interconnecting objects, performs slightly less effectively. This could be due to introducing too many node connections that may disrupt the efficient exchange of information between nodes.

TABLE III: Comparison between concatenating object directly to skeleton node (concatenation) and embedding object node (object nodes) on IKEA-ASM dataset by Top-1 and F1@k score.

Model	Mtehods	Top-1	F1@10%	F1@25%	F1@50%
STGCN	concatenation	59.89	0.46	0.40	0.27
	object nodes	**61.86**	0.49	0.43	0.29
AGCN	concatenation	63.20	0.51	0.45	0.24
	object nodes	**65.21**	0.55	0.58	0.33
EGCN	concatenation	65.14	0.54	0.49	0.33
	object nodes	**66.31**	0.58	0.52	0.33

TABLE IV: Comparison of model performance between different connection strategies on IKEA ASM dataset by Top-1 and F1 score. **OJ**: Objects are connected to all joints, with no connections between objects. **OH**: Objects are connected to two hand joints. **OOH**: Objects are connected to two hand joints, and objects are connected to each other.

Model	Strategy	Top-1	F1@10%	F1@25%	F1@50%
EGCN	OJ	64.82	0.55	0.49	0.33
	OH	**66.31**	0.58	0.52	0.36
	OOH	66.09	0.57	0.52	0.35

TABLE V: Contribution of different geometric object features using object nodes by Top1 and F1@k score on IKEA ASM. **J**: Skeleton joint. **C**: Object category. **L**: Object center location. **Q**: Quaternion.

Model	Object features	Top-1	F1@10%	F1@25%	F1@50%	#FLOPs
EGCN	J + C	57.11	0.46	0.39	0.25	2.0080 G
	J + L	59.86	0.49	0.43	0.27	2.0091 G
	J + Q	57.41	0.47	0.40	0.25	2.0112 G
	J + Q + C	58.32	0.48	0.42	0.26	2.0120 G
	J + C + L	60.18	0.51	0.44	0.28	2.0100 G
	J + C + L + Q	**66.31**	0.58	0.52	0.36	2.0137 G

4) Contribution of different geometric object features: As shown in Table V, we assess the contribution of various geometric object features to the model performance. Specifically, we examine the impact of object category, object center location, and object quaternion representation on the overall effectiveness and complexity of the model. Our results indicate that object center location (L) significantly influences model performance. Conversely, the object category (C) has a lesser impact on model performance, it may be because of lacking spatial information exchange with other skeleton joints. Therefore, the GCN model is more sensitive to spatial location information. Additionally, while the object quaternion (Q) provides supplementary benefits, its contribution may not be as substantial as object location. However, the orientation information offered by quaternions could aid the model in understanding actions of the temporal domain, particularly for interactions involving objects. Regarding computational complexity, introducing quaternion features leads to a mere increase in computational processing due to their thin dimension.

TABLE VI: Comparison of model performance between different registration frame steps by Top-1 and F1 score on IKEA ASM dataset.

Model	Step	Top-1	F1@10%	F1@25%	F1@50%
EGCN	5	64.39	0.55	0.49	0.32
	10	**66.31**	0.58	0.52	0.36
	50	65.31	0.56	0.50	0.33

TABLE VII: Comparison of model performance with 2D and 3D skeleton and object location features by Top-1 and F1@k on IKEA ASM dataset.

Model	Location features	Top-1	F1@10%	F1@25%	F1@50%
STGCN	2D	61.36	0.49	0.43	0.29
	3D	**61.86**	0.49	0.42	0.29
AGCN	2D	65.21	0.55	0.49	0.34
	3D	**66.92**	0.56	0.49	0.38
EGCN	2D	66.31	0.58	0.52	0.36
	3D	**67.69**	0.59	0.53	0.37

5) Object registration frame step analysis: In the object registration phase, determining the appropriate frame step is essential to enhance efficiency and ensure informative transformation features. To investigate this, we conducted an ablation study on different frame step intervals. A frame step that is too short may result in minimal translation along the x, y, and z axes, with negligible rotation information. Consequently, we initially set the time step interval to 5 frames. However, as demonstrated in Table VI, experiments on registration time steps of 5, 10, and 50 frames are conducted. Based on the results, a time step of 10 frames is selected for our proposed method. Therefore, selecting the optimal time step for object registration is pivotal for effective feature extraction and the overall performance of our approach.

6) 2D and 3D geometric inputs analysis: The addition of 3D position information offers a notable improvement over 2D position data, primarily because of the enhanced accuracy provided by the additional axis z in the 3D coordinate system. This extra axis allows for a more precise representation of both object and joint locations. By leveraging the depth stream, we can accurately capture the 3D positions of objects and joints. As demonstrated in Table VII, the incorporation of this 3D location feature positively impacts our mode performance. The enhanced accuracy provided by 3D information enables the model to better understand and learn object features in various scenarios, particularly in situations involving overlapping positions or complex spatial interactions with humans.

D. Comparison with SOTA methods

To achieve the best performance, we use EGCN as the basic structure, combining 3D features and "**OH**" nodes connection strategy to compare with the state-of-the-art models. We proposed a comprehensive evaluation of our proposed method both on the IKEA ASM and Bimanual Actions dataset. As Table VIII shows that our approach achieves a frame-wise Top-1 score improvement of 10.8% and an average F1@k improvement of 13.3%, compared to EGCN. While on the Bimanual Actions dataset on Table IX, it achieves a frame-wise Top-1 score improvement of 11.4% and an average F1@k improvement of 5.3%, with negligible increases in model complexity. The results confirm that our method effectively enhances action recognition performance with a very small impact on computational demands.

TABLE VIII: Comparison of model performance of final results of our proposed method with other SOTA models by Top-1 and F1@k score on IKEA ASM dataset.

Model	Top-1	F1@10%	F1@25%	F1@50%	#Params	FLOPS
STGCN	52.06	0.41	0.35	0.23	3.08 M	1.8733 G
AGCN	55.54	0.42	0.36	0.24	3.45 M	2.1082 G
EGCN	56.83	0.45	0.39	0.25	3.38 M	2.0039 G
Ours	**67.69**	0.59	0.53	0.37	3.38 M	2.0146 G

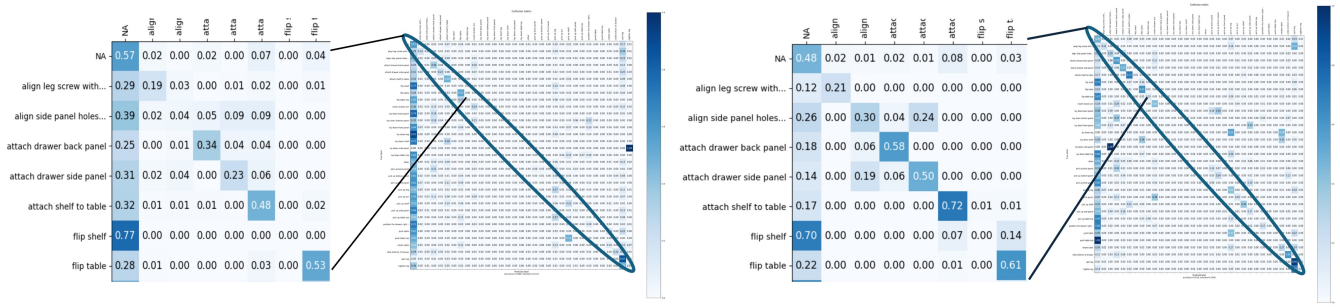
TABLE IX: Comparison of model performance of final results of our proposed method with other SOTA models by Top-1 and F1@k score on Bimanual Actions dataset.

Model	Top-1	F1@10%	F1@25%	F1@50%	#Params	FLOPS
STGCN	52.06	0.45	0.41	0.33	3.08 M	1.8733 G
AGCN	55.54	0.49	0.43	0.38	3.45 M	2.1082 G
EGCN	59.11	0.51	0.45	0.41	3.38 M	2.0039 G
Ours	**70.56**	0.58	0.51	0.44	3.38 M	2.0146 G

E. Qualitative Analysis

We present qualitative results comparing our model and related methods on the IKEA Assembly dataset and Bimanual Actions dataset. Figure 3 illustrates the diagonal part of the confusion matrix, indicating the accuracy of each class on the IKEA Assembly dataset. Our results indicate that incorporating object features alongside skeleton data enhances the model ability to identify actions. Without geometric object features, the model struggled to differentiate between similar actions involving different objects. For instance, actions like $pick\ up\ a\ shelf$ and $pick\ up\ leg$ could be easily confused. However, with the addition of object features, this confusion notably decreases.

We display the ground truth and model inference for various models on the IKEA ASM and Bimanual Actions datasets, respectively. We observe that existing models such as STGCN, AGCN, and EGCN encounter challenges in classifying short-term fine segments. As shown in Figure 4 for instance, actions like $pick\ up\ panel$, which lasts approximately 30 frames, are often misclassified by these models. Furthermore, long-term actions like $putting\ a\ drawer\ right\ side\ up$ or $aligning\ legs\ and\ panels$ exhibit instability in classification across all models. However, our proposed model, incorporating object rotation into the analysis, demonstrates improved stability and accuracy, especially in long-term action classification.



(a) Confusion matrix of EGCN without 3D object geometric features

(b) Confusion matrix of our proposed method with 3D object geometric features

Fig. 3: Confusion matrix for the top-1 prediction of accumulative framewise classification correctness on IKEA ASM dataset. Compared to the EGCN results that do not utilize 3D geometric object features ((a) only 7 actions can be highly recognized), our proposed method of fusing 3D geometric objects and skeleton can classify actions more accurately ((b)18 actions can be highly recognized).

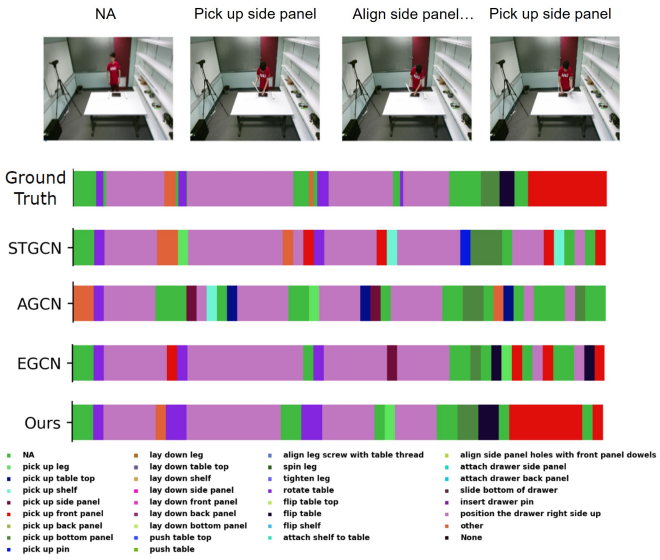


Fig. 4: Comparison of the qualitative results on IKEA Assembly dataset for $Kallax\ shelf\ drawer$ task.

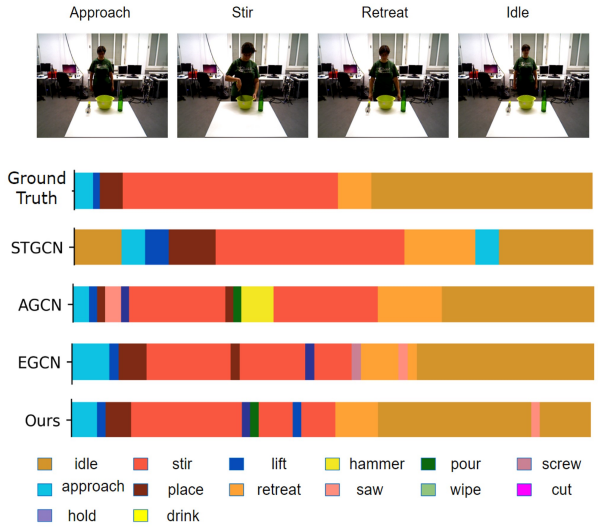


Fig. 5: Comparison of the qualitative results on Bimanual Actions dataset for $Cooking\ with\ bowl$ task.

Additionally, as shown in Figure 5, our model shows increased stability in classifying actions on the Bimanual Action dataset after incorporating object features. This enhancement is particularly evident when compared to existing models such as STGCN, which often encounter delays in predicting actions due to a lack of information regarding hand-object interactions.

In summary, the incorporation of 3D geometric object features into action recognition models provides valuable context and stability, leading to the enhancement of the general performance of state-of-the-art skeleton-based action recognition models.

V. Conclusion

In this work, we introduced a comprehensive method to incorporate low-dimensional 3D geometric object features into skeleton-based action recognition. Our approach efficiently integrates 3D geometric object features into skeleton data through two distinct fusion strategies, enhancing the overall performance. We also examined various node-linking configurations for the adjacency matrix, identifying an optimal graph representation structure for improved model performance. Moreover, we conducted evaluations on two demanding datasets, demonstrating that the integration of 3D geometric object features significantly enhances action recognition performance for the state-of-the-art skeleton-based models with negligible growth of model complexity.

Acknowledgment

We gratefully acknowledge the funding of the Lighthouse Initiative Geriatronics by StMWi Bayern (Project X, grant no. 5140951) and LongLeif GaPa GmbH (Project Y, grant no. 5140953).

REFERENCES

[1] F. Krebs, A. Meixner, I. Patzer, and T. Asfour, "The kit bimanual manipulation dataset," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, 2021, pp. 499–506.

[2] F. Krebs, L. Leven, and T. Asfour, "Recognition of bimanual manipulation categories in rgb-d human demonstration," in *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, 2023, pp. 1–8.

[3] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6479–6488.

[4] M. Javeed and A. Jalal, "Deep activity recognition based on patterns discovery for healthcare monitoring," in *2023 4th International Conference on Advancements in Computational Sciences (ICACS)*, 2023, pp. 1–6.

[5] I. Rodomagoulakis, N. Kardaris, V. Pitsikalis, E. Mavroudi, A. Katsamanis, A. Tsiami, and P. Maragos, "Multimodal human action recognition in assistive human-robot interaction," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2702–2706.

[6] R. Dai, S. Das, K. Kahatapitiya, M. S. Ryoo, and F. Brémond, "Ms-tct: Multi-scale temporal convtransformer for action detection," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 20 009–20 019.

[7] C. Patsch, J. Zhang, Y. Wu, M. Zakour, D. Salihu, and E. Steinbach, "Long-term action anticipation based on contextual alignment," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 5920–5924.

[8] Z. Zhuang, Y. Ben-Shabat, J. Zhang, S. Gould, and R. Mahony, "Goferbot: A visual guided human-robot collaborative assembly system," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 8910–8917.

[9] Y. Wu, X. Su, D. Salihu, H. Xing, M. Zakour, and C. Patsch, "Modeling action spatiotemporal relationships using graph-based class-level attention network for long-term action detection," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 6719–6726.

[10] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4724–4733.

[11] T. d. Blegiers, I. R. Dave, A. Yousaf, and M. Shah, "Eventtransact: A video transformer-based framework for event-camera based action recognition," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 1–7.

[12] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[13] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 026–12 035.

[14] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, "Constructing stronger and faster baselines for skeleton-based action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 2, pp. 1474–1488, 2022.

[15] R. Dai, S. Das, L. Minciullo, L. Garattoni, G. Francesca, and F. Bremond, "Pdan: Pyramid dilated attention network for action detection," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 2969–2978.

[16] R. Dai, S. Das, and F. Bremond, "Ctrn: Class-temporal relational network for action detection," *arXiv preprint arXiv:2110.13473*, 2021.

[17] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in neural information processing systems*, vol. 27, 2014.

[18] Z. Yang, A. Zeng, C. Yuan, and Y. Li, "Effective whole-body pose estimation with two-stages distillation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4210–4220.

[19] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 2969–2978.

[20] J. Lee, M. Lee, D. Lee, and S. Lee, "Hierarchically decomposed graph convolutional networks for skeleton-based action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 10 444–10 453.

[21] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional lstm network for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1227–1236.

[22] T. Chen, D. Zhou, J. Wang, S. Wang, Y. Guan, X. He, and E. Ding, "Learning multi-granular spatio-temporal graph network for skeleton-based action recognition," in *Proceedings of the 29th ACM international conference on multimedia*, 2021, pp. 4334–4342.

[23] H. Duan, J. Wang, K. Chen, and D. Lin, "Dg-stgcn: Dynamic spatial-temporal modeling for skeleton-based action recognition," *arXiv preprint arXiv:2210.05895*, 2022.

[24] Y. Ben-Shabat, X. Yu, F. Saleh, D. Campbell, C. Rodriguez-Opazo, H. Li, and S. Gould, "The ikea asm dataset: Understanding people assembling furniture through actions, objects and pose," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 847–859.

[25] C. R. G. Dreher, M. Wächter, and T. Asfour, "Learning object-action relations from bimanual human demonstration using graph networks," *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 187–194, 2020.

[26] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] V.-M. Khong and T.-H. Tran, "Improving human action recognition with two-stream 3d convolutional neural network," in *2018 1st international conference on multimedia analysis and pattern recognition (MAPR)*. IEEE, 2018, pp. 1–6.

[29] L. Zhang, G. Zhu, P. Shen, J. Song, S. Afaq Shah, and M. Bennamoun, "Learning spatiotemporal features using 3dcnn and convolutional lstm for gesture recognition," in *Proceedings of the IEEE international conference on computer vision workshops*, 2017, pp. 3120–3128.

[30] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[31] Y. Guo, F. Sohel, M. Bennamoun, J. Wan, and M. Lu, "An accurate and robust range image registration algorithm for 3d object modeling," *IEEE Transactions on Multimedia*, vol. 16, no. 5, pp. 1377–1390, 2014.

[32] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3343–3352.

[33] D. Salihu and E. Steinbach, "Sgpcr: Spherical gaussian point cloud representation and its application to object registration and retrieval," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 572–581.

[34] G. Jocher, A. Stoken, J. Borovec, NanoCode012, ChristopherSTAN, L. Changyu, Laughing, tkianai, A. Hogan, lorenzomammana, yxNONG, AlexWang1900, L. Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, F. Ingham, Frederik, Guilhen, Hatovix, J. Poznanski, J. Fang, L. Yu, changyu98, M. Wang, N. Gupta, O. Akhtar, PetrDvoracek, and P. Rai, "ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements," Oct. 2020.

[35] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[36] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 881–892, 2002.

[37] M. Hahsler, M. Piekenbrock, and D. Doran, "dbscan: Fast density-based clustering with r," *Journal of Statistical Software*, vol. 91, pp. 1–30, 2019.

[38] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and vision computing*, vol. 10, no. 3, pp. 145–155, 1992.