CHAMP: A Large-Scale Dataset for Skeleton-Based Composite HumAn Motion Prediction

Wanying Zhang, Mengyuan Liu, Xinshun Wang, Shen Zhao, and Can Wang

***Abstract*— Skeleton-based human motion prediction task aims to forecast future skeleton frames conditioned by observed skeleton sequence. Different from previous methods that focus on human motion prediction for atomic actions, we observe that people are witnessed to perform composite actions which consist of atomic actions that simultaneously happen. Considering the large number of action types, it is more laborious to collect composite actions than atomic actions. This paper presents a practical composite human motion prediction task, whose training data just contains atomic actions meanwhile the test data contains both atomic actions and composite actions. To evaluate this task, we collect a large-scale Composite HumAn Motion Prediction (CHAMP) dataset, whose training data has 16 types of atomic actions and test data has 50 types of composite actions. Despite the success of previous human motion prediction methods using Graph Convolutional Networks (GCN), these methods achieve inferior performances on our CHAMP dataset due to the huge domain gap between the training and test data. To solve this problem, we present a composite human motion prediction framework containing three modules. First, a Composite Motion Synthesis (CMS) module is designed to generate synthesized composite human actions from atomic actions. Second, a Composite GCN module is presented to predict human motion by modeling different human body parts. Third, a human body partition policy network is used to choose the best partition strategy for both the CMS and Composite GCN modules. Extensive experiments on the CHAMP dataset verify the effectiveness of our framework which obviously outperforms GCN-based methods.**

***Index Terms*— Human motion prediction, graph convolution, skeleton data.**

I. Introduction

HUMAN perception tasks including action recognition and motion prediction have always been active research fields for their wide potential applications in intelligent surveillance and human-robot interaction [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. Recent researchers take advantage of depth sensors to capture accurate 3D human skeletons in real-time and pay special attention to skeleton-based human motion prediction task [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21]. Taking an atomic action as an example, this task aims to learn a model to complement an unfinished atomic action by predicting future poses from observed poses. To ensure the generalization ability of the model, existing datasets commonly follow an underlying assumption that both training and test data contain the same types of atomic actions.

However, human actions in daily scenarios are more complex than samples in existing datasets. Instead of atomic actions, people are usually witnessed to perform composite human actions which are composed of atomic actions that simultaneously happen. For example, people may perform atomic actions like "walking", and "waving", and may also perform composite actions like "walking meanwhile waving", which is a combination of those two atomic actions. To predict composite actions, one direct solution is to collect more training data to cover corresponding composite action types. **Consider that it is laborious to collect samples for composite actions since the possible composite action types can be extremely large.** Therefore, we present a practical setting for human composite motion prediction by using atomic actions for training and using both atomic actions and composite actions for testing. Figure 1 shows the comparison between the human motion prediction task and our proposed composite human motion prediction task.

Our proposed task aims to address the lack of action categories due to the difficulties in collecting all classes of human motions. **To handle our setting, a natural problem arises: "can existing human motion prediction methods also work on our proposed human composite motion task?"** Due to the absence of proper datasets for the human composite motion task, we answer this question by first collecting a new Composite HumAn Motion Prediction (CHAMP) dataset, and then evaluating the most recent GCN-based human motion prediction methods on the CHAMP dataset. Despite the success of GCN-based methods on human motion prediction tasks, these methods achieve only inferior results on the CHAMP dataset, mainly due to the huge differences between the training data and test data.

In order to address the task of composite human motion prediction, we present our approach consisting of two main modules: the Composite Motion Synthesis (CMS) module and

Manuscript received 16 September 2023; revised 10 April 2024; accepted 27 May 2024. Date of publication 3 June 2024; date of current version 30 October 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62203476, in part by the Natural Science Foundation of Guangdong Province under Grant 2024A1515012089, and in part by the Natural Science Foundation of Shenzhen under Grant JCYJ20230807120801002. This article was recommended by Associate Editor M. H. Yousaf. *(Corresponding author: Mengyuan Liu.)*

Wanying Zhang, Mengyuan Liu, and Xinshun Wang are with the National Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School, Shenzhen 518055, China (e-mail: zhangwanying0201@163.com; nkliuyifang@gmail.com; wangxsh36@mail2.sysu.edu.cn).

Shen Zhao is with the School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 518107, China (e-mail: z-s-06@163.com).

Can Wang is with the Laboratory on Multimedia Information Processing, Department of Computer Science, Kiel University, 24098 Kiel, Germany (e-mail: canwang@pku.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2024.3408332.

Digital Object Identifier 10.1109/TCSVT.2024.3408332

1051-8215 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

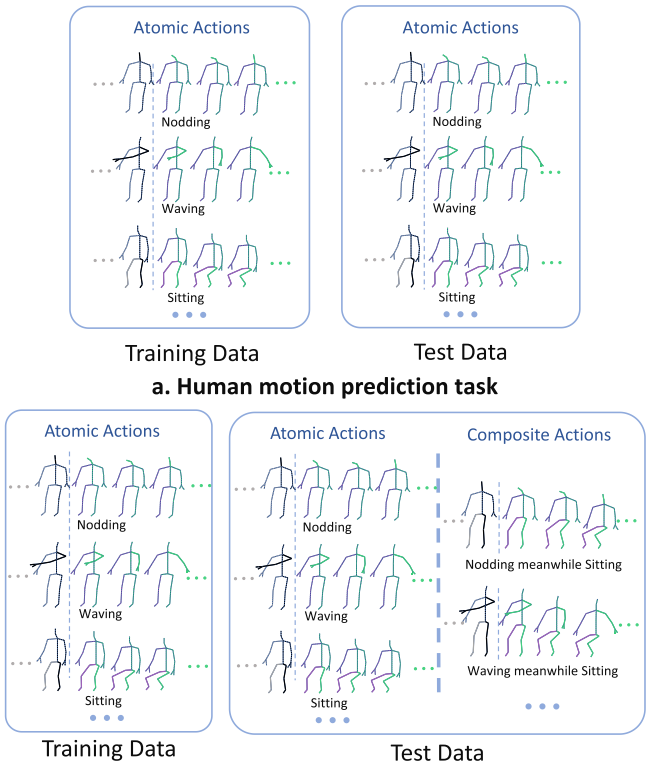


Fig. 1. Comparison between human motion prediction task and our proposed composite human motion prediction task. (a) Human motion prediction uses all classes of actions (e.g., "sitting", "waving" and "nodding") for both training and testing. (b) Our proposed composite motion prediction uses atomic actions for training and uses both atomic actions and composite actions (e.g., "waving meanwhile sitting", and "nodding meanwhile sitting") for testing. Each action sample is denoted by a skeleton for clarity.

the Composite Graph Convolutional Network (GCN) module. In the subsequent paragraphs, we will provide a detailed introduction to these modules.

Intuitively, what makes our task challenging is the lack of training motion data and it can be alleviated by Data Augmentation to synthesize the missing data. In this paper, we propose a Composite Motion Synthesis (CMS) module in terms of the data insufficiency issue, which is based upon a Bi-LSTM AutoEncoder for motion synthesis. Using only atomic actions as input, this module can generate actions analogous to the composite actions, thus filling the vacancy of composite actions in the training data. We follow the encode-fuse-decode scheme, encoding different atomic actions with different encoders, fusing their latent attributes, and using the same decoder for composite action synthesis. More precisely, human skeletons are split into parts before encoding atomic actions, in which way we can combine multiple body parts into a complete human body to obtain composite poses after decoding. Since there are varied ways to partition, we design a policy module using Straight-Through (ST) Gumbel Estimator [22] to select the most appropriate partition automatically.

Another solution for this task is to predict the movements of parts of the body separately using our Composite GCN module. Based on our observation, a specific atomic action always happens at the level of the body part and there is no necessary correlation between the two. In terms of our CHAMP dataset, atomic actions can be classified as upper-body actions or lower-body actions, and a person can do any upper-body actions when doing a lower-body action and the other way around. For example, a person can either wave hands or shake their head when walking, and such combinations of upper- and lower-body movements are arbitrary. We use this intuition when modeling the predictor based on Graph Convolutional Network [23], [24]. Since only a few parts of the human body move in atomic actions, modeling the entire structure of the human skeleton creates a lot of redundancy and reduces the effect of prediction. We model the human body in parts and forecast their future poses separately according to the partition given by the Composite Motion Synthesis module. However, this segmentation-based prediction strategy does not work for the generated composite actions because it makes it pointless to synthesize the whole-body actions. So we build another branch in the predictor modeling the whole body skeleton and train it with the generated composite actions.

In summary, our contributions are as follows:

- We set up a new composite human motion prediction task, using atomic actions as training data and composite actions as testing data, which is more practical than the common human motion prediction task.
- We present a composite human motion prediction framework, which contains three modules. The Composite Motion Synthesis (CMS) module is proposed to solve the data insufficiency issue, while the Composite GCN module efficiently predicts future poses by segmenting the human skeleton. The third is a human body partition policy network which chooses the best partition strategy for both the CMS and Composite GCN modules. Our approach achieves the best performance compared with the state-of-the-art approaches and can be used as a baseline method for composite motion prediction.
- We collect the CHAMP dataset, a novel large-scale dataset that is composed of 50 composite actions and 16 corresponding atomic actions, annotated with 3D joint coordinates. We also define the benchmark for the composite motion prediction task with suitable train/test splits and evaluation protocols. Our code and dataset are publicly available at https://github.com/WanyingZhang/CHAMP

II. Related Work

A. Human Motion Prediction Methods

Nonlinear Markov models [25] and Gaussian Process dynamical models [26] are used in traditional models of human motion prediction. However, these approaches have difficulties in solving long-term and complex motion predictions. As human motion can be viewed as a time series, many works in human motion prediction have been addressed with recurrent models and most of them have an encoder-decoder structure, in which the encoder generates the hidden attributes for given motion sequences or frames and propagates to the decoder, where generates the future motion sequences or frames. For instance, [27] proposed an Encoder-Recurrent-Decoder (ERD) model that incorporates a nonlinear network

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

before and after recurrent layers to encoder and decoder human motions. Reference [28] encoder human motion by a structural RNN model with spatio-temporal graphs, which is manually designed. Reference [29] choose to predict velocities instead of poses using a residual-based model and uses a standard sequence-to-sequence learning model with GRU [30] cells as the encoder and decoder. Although RNNs have a better performance in human motion prediction than traditional methods, they suffer from the same error accumulation problem, and discontinuities between the last observed frame and the first predicted frame often show up in some earlier RNN-based models.

To avoid the drawbacks of RNN-based models, feed-forward networks are considered as a solution recently. Reference [31] forward the recent pose frame via a fully-connected network, and encode temporal dependencies via convolutions and spatial information by using the kinematic tree. Reference [32] use a convolutional sequence to sequence model to encode and decode the motion sequence. The model contains a two-dimensional matrix with columns representing the temporal dimension and rows representing the spatial dimension, which models the temporal and spatial dependencies effectively but is limited by the size of the convolution filter. Recently, GCN-based methods have had a great performance in human perception [11], [16], [18], [33], [34], [35], [36], [37], regarding the structure of the human skeleton as a graph. Reference [11] use a feed-forward graph convolutional network with learnable adjacent matrices in human motion prediction. The attitude is regarded as a fully connected graph and GCN is used to model the relationship between any pair of joints. The joint trajectories are represented by Discrete Cosine Transform (DCT) coefficients in the temporal domain. Based on [11], self-attention [35] was added to [16] later, making use of historical information by comparing previous sub-sequences and the current sub-sequence, which makes a great improvement in the prediction effect. Reference [36] further study the use of different types of attention calculated at the level of joints, body parts, and full postures. References [18] and [34] extend the graph of the human pose to the abstraction levels of the human skeleton on a multi-scale.

B. Motion Synthesis

One of the main challenges in our composite human motion prediction task is the insufficiency of training data and data augmentation is one approach to address this challenge. The concept of "data augmentation" is introduced by [38], associating the augmented data with the observed data by a many-to-one mapping. Classical data augmentation methods typically involve transformations such as cropping, flipping, rotating, adding random noise, scaling, and random warping. Gaussian noise is a simple but effective approach for data augmentation, where [27] increase the variety of input data by destroying the input motion using zero-mean Gaussian noise and [39] add Gaussian noise to prevent the model from overfitting. Reference [40]. Color adjustment, blurring, sharpening, white balance, and other distortions have also been used in Image data augmentation [41], [42].

Some classical data augmentation methods can be used for skeleton-based motion synthesis, such as cutting different parts of the skeleton from different movements and piecing them together. However, the cut-and-piece method is too crude and might synthesize physically-implausible motions.

C. Datasets for Human Motion Prediction

Actions in commonly used datasets for 3D human perception tasks are usually weak correlated or pseudo. **Human3.6M** [43] is one of the most commonly used datasets for human motion prediction tasks, containing 3.6 million 3D human poses and the corresponding images. Pose data is given concerning a skeleton of 32 joints. The dataset consists of 15 different actions (e.g. Phoning, Walking, Eating) performed by 7 subjects. Most of the actions are independent of each other, except for "Walking", "Walking dog" and "Walking together". There are also some motions with many types of asymmetries (e.g. walking with a hand in a pocket, walking with a bag on the shoulder). But in general, the Human3.6M dataset does not meet the requirements of multiple classes of composite actions and corresponding atomic actions. **AMASS** [44] is a large database of human motion. The dataset unifies 15 different optical mocap datasets (e.g. CMU, KIT, and BMLrub) using SMPL [45], [46] which provides a standard skeletal representation as well as a fully rigged surface mesh. It has more than 11000 actions performed by over 300 subjects, and a human pose is represented by 52 joints, including 22 body joints and 30 hand joints. The official testing split of AMASS contains sequences that consist of transitions between unrelated actions, which is not suitable for our composite motion prediction task. **3DPW** [47]dataset consists of 60 video sequences taken from a moving phone camera. Each pose is represented by 26 joints. The dataset encompasses indoor and outdoor actions containing 51,000 frames captured at 30Hz. The actions include everyday activities such as walking around the city, walking upstairs, drinking coffee, or taking the bus. **CMU-Mocap** dataset contains samples that are mainly generated by the different actions of people attached to sensors in each joint in a certain period. There are five main categories in the dataset which include "human interaction", "interaction with environment", "locomotion", "physical activities & sports" and "situations & scenarios".

Some researchers start to pay attention to the interaction between multiple persons [48] or human-scene interaction [49]. However, there is a lack of research dedicated to the composite motion perception task, resulting in a shortage of datasets that include sufficient data on both atomic actions and their composite actions. Consequently, there is a need to gather a more task-specific dataset for the Skeleton-based Composite Human Motion Prediction task.

III. Our Proposed Framework

A. The CHAMP Dataset

CHAMP Dataset is a large-scale dataset for composite human motion prediction performed by 22 subjects. It is recorded in a multimedia studio equipped with Microsoft Kinect V2 cameras [50], which is one of the most common

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.



Fig. 2. Left: Different modalities in the CHAMP dataset, from left to right are RGB, Depth, and joints. Right: Examples of unrecorded body parts.

RGB-D devices. Overall, our dataset contains 229134 visual frames for each viewpoint and each image type.

Using Kinect V2 cameras, we collected three major data modalities provided by this sensor: depth images, RGB frames, and 3D joint information. Depth images maintain the two-dimensional depth values in millimeters, with a resolution of 702 × 538 pixels at 50FPS, while RGB frames have the same resolution and frame rate as depth images. Joint information provides 3D locations of 25 major joints of a person, which is automatically generated according to the algorithm [51] provided by Microsoft. The skeleton generation is based on pixel location classification and pixel depth clustering, using the mean-shift algorithm with weighted Gaussian kernels. In this way, we can obtain the skeleton nodes lying on the surface of the human body. There needs to be a backward Z-value offset to get the correct 3D skeleton node while this offset value can be learned automatically. Figure 2 illustrates different modalities present in the CHAMP dataset. Additionally, we include an example showcasing unrecorded body parts, attributable to either body parts extending beyond the frame or occlusions between limbs. These instances may result in inaccuracies in joint coordinate estimation. However, such scenarios reflect real-world conditions encountered during prediction tasks, thereby adding an extra layer of complexity to the prediction task.

We have 66 pose classes in total, which are divided into two groups(atomic actions and composite actions). In detail, there are 16 atomic actions, including 10 upper body actions (raise up, nod, wave, etc.), 5 lower body actions (sit down, squat, walking, etc.), and a still state action. The 50 composite actions are the pairwise combination of atomic actions. the various methods for constructing composite actions in real-world scenarios, those composed of atomic actions from both upper and lower body parts are the most common, thus serving as a typical example for evaluating this new task.

To comprehensively analyze the characteristics of CHAMP, we evaluate the variability of actions by computing the variance of joint coordinate points within action sequences. Specifically, for each joint coordinate point, we compute its variance within a sequence, and then we aggregate these values across all joint coordinate points to calculate the average variance for each action sequence. Finally, we compute the average variance for all sequences within each action category to obtain a measure of variability for the entire action category. Higher variance indicates greater variability within action sequences, potentially corresponding to more complex action types. Table I presents the action classification (Omit the still-state action) of the CHAMP dataset, including the average number of frames per action sequence and the variability of each action (the average variance of each action). We can see that the average number of frames of actions in our dataset is around 30, indicating that our actions are completed and have a large amplitude in a short period. Because of this characteristic of the actions in our dataset, we focus on short-term prediction in the Composite human motion prediction task. As observed, the average variance of key points for composite actions is generally higher than that for atomic actions, suggesting that composite actions exhibit greater variability in key point movements and are more complex.

Figure 3 shows 2 groups of examples of the CHAMP dataset. Each group of the examples visualizes an upper-body atomic action (Clockwise, RaiseUp, KeepClose, and Left), a lower-body atomic action (StandUp, SquatUp, SitDown, and Squat), and their corresponding composite action (StandUp + Clockwise, SquatUp + RaiseUp, SitDown + KeepClose, and Squat + Left). Because the data associated with "walking" need further processing, we do not use them in our experiments while the rest of the datasets are still sufficient to evaluate our model in this task. Our training dataset includes 15 atomic actions done by all of the 22 subjects, while the validation dataset and test dataset contain 40 classes of composite actions done by 11 subjects respectively.

We employ three cameras located at the same height to capture the poses from three different perspectives: front, front-left, and front-right. For each pose, three front views are recorded at the same time. Each pose is repeated 5 times by each subject. For each sequence we manually labeled poses. The format of the pose labels consists of the initial frame number and terminal frame number of the poses, both followed by the pose label. Table II presents the comparison between commonly utilized datasets for human perception and our CHAMP dataset. Our CHAMP dataset stands out in terms of its scale, diversity, and comprehensiveness of the data. It demonstrates competitiveness in terms of the number of sequences and it features multiple modalities and supports multiple tasks.

B. Problem Definition

The task of composite motion prediction is to predict the future motion sequences of a single person. We use $\mathbf{X}_{1:N} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$ to represent the N frames of history sequences, and $\mathbf{X}_{N+1:N+T}$ as the future T frames we aim to forecast, where $\mathbf{x}_i \in \mathbf{R}^L$ is the 3D locations of L human major joints. The pipeline of our method is shown in Figure 4. Our goal is to utilize only original atomic actions $\mathbf{A}$ as training

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

TABLE I

ACTION CLASSIFICATION OF THE CHAMP DATASET, WITH THE AVERAGE NUMBER OF FRAMES PER ACTION IN A SEQUENCE AND THE VARIABILITY OF EACH ACTION (THE AVERAGE VARIANCE OF EACH ACTION). THE 50 CLASSES OF COMPOSITE ACTIONS ARE PAIRS OF 5 LOWER-BODY ATOMIC ACTIONS AND 10 UPPER-BODY ATOMIC ACTIONS

Atomic actions					
act (lower-body)	walking	sitDown	standUp	squat	squatUp
Avg. frames	45.1	29.8	29.3	27.6	23.3
Avg.Var.(mm^2)	52.2	52.5	52.9	54.3	55.9
act (upper-body)	wave	raiseUp	clockwise	counterclockwise	keepClose
Avg. frames	33.4	39.8	34.6	34.4	32.6
Avg.Var.(mm^2)	62.5	65.1	57.6	58.2	57.8
act (upper-body)	keepFar	left	right	nod	shake
Avg. frames	33.5	32.1	31.7	27.1	27.7
Avg.Var.(mm^2)	58.5	57.3	60.3	47.9	47.8
Composite actions					
act	sitDown+clockwise	sitDown+counterclockwise	sitDown+keepClose	sitDown+keepFar	sitDown+left
Avg. frames	29.1	29.5	27.6	28.4	29.5
Avg.Var.(mm^2)	63.0	63.8	65.8	67.0	62.0
act	sitDown+nod	sitDown+right	sitDown+shake	sitDown+wave	squatUp+clockwise
Avg. frames	26.3	28.9	26.4	29.8	27.7
Avg.Var.(mm^2)	50.3	63.4	50.4	66.5	66.0
act	squatUp+counterclockwise	squatUp+keepClose	squatUp+keepFar	squatUp+left	squatUp+nod
Avg. frames	28.4	26.3	27	27.7	24.9
Avg.Var.(mm^2)	65.7	68.0	68.7	65.9	56.5
act	squatUp+raiseUp	squatUp+right	squatUp+shake	squatUp+wave	squat+clockwise
Avg. frames	27.6	26.7	23.8	27.6	29.4
Avg.Var.(mm^2)	70.0	67.8	55.1	69.2	63.0
act	squat+counterclockwise	squat+keepClose	squat+keepFar	squat+left	squat+nod
Avg. frames	28.7	28.1	28.9	28.9	27.1
Avg.Var.(mm^2)	63.1	64.8	65.8	62.2	52.0
act	squat+raiseUp	squat+right	squat+shake	squat+wave	standUp+clockwise
average frames	31.8	28.8	27.1	28.3	31.2
Avg.Var.(mm^2)	69.0	64.0	51.5	66.3	64.9
act	standUp+counterclockwise	standUp+keepClose	standUp+keepFar	standUp+left	standUp+nod
Avg. frames	31.0	29.5	30.2	32.1	27.2
Avg.Var.(mm^2)	65.6	68.5	67.0	63.5	52.9
acts	standUp+raiseUp	standUp+right	standUp+shake	standUp+wave	sitDown+raiseUp
Avg. frames	32.8	30.8	28.0	31.5	31.4
Avg.Var.(mm^2)	68.3	65.1	53.2	68.9	67.1
act	walking+clockwise	walking+counterclockwise	walking+keepClose	walking+keepFar	walking+left
Avg. frames	34.0	35.2	32.8	32.9	31.2
Avg.Var.(mm^2)	59.1	61.8	60.5	61.6	58.1
act	walking+raiseUp	walking+right	walking+shake	walking+wave	walking+nod
Avg. frames	36.8	21.7	29.9	34.4	28.8
Avg.Var.(mm^2)	66.2	59.9	52.8	67.7	51.4

data and composite actions $\mathbf{C}$ as testing data. We use two strategies, one of which is motion synthesis and the other is to model the human body parts separately. Therefore, we design a Composite Motion Synthesis (CMS) module to increase the number of training samples:

$$\hat{\mathbf{C}} = \mathcal{C}(\mathbf{A}) \tag{1}$$

where $\mathcal{C}$ denotes the CMS module. $\hat{\mathbf{C}}$ denotes the synthesized composite actions. Using only atomic actions $\mathbf{A}$ as inputs, we generate composite actions $\hat{\mathbf{C}}$ as a supplement to the training data in the prediction progress. We then feed $\hat{\mathbf{C}}$ into the Composite GCN module $\mathcal{G}$ as well as the original atomic actions $\mathbf{A}$ to train the predictor:

$$\mathbf{X}_{N+1:N+T} = \mathcal{G}(\mathbf{X}_{1:N}), \quad \mathbf{X} \in \{\mathbf{A}, \hat{\mathbf{C}}\} \tag{2}$$

where we use N frames of history sequences as input and forecast the future T frames in both the synthesized composite actions $\hat{\mathbf{C}}$ and the atomic actions $\mathbf{A}$. Last we use the trained predictor CGCN to forecast all original actions including the composite actions as well as the atomic actions.

There are three branches in the predictor CGCN two of which divide a human body into two different parts, and the third branch models the entire human body. The three branches forecast independently and then fuse their prediction results as the final result. Both processes in the CMS module and the predictor CGCN involve the partitioning of the human skeleton, and there are several ways of partitioning. To deal with it, we design a policy module to automatically choose the most appropriate partition. We cover these modules in detail in the following subsections.

C. Composite Motion Synthesis

This module serves the purpose of expanding the action classes within the training set and addressing the deficiency of composite motions. Consequently, it can be considered as a data augmentation technique. Our hypothesis is that a composite action can be deconstructed into multiple atomic actions executed by non-overlapping body parts, which occur simultaneously. In the context of our dataset, atomic actions combine in pairs to characterize composite actions. As discussed in Section II, a basic method for synthesizing new actions involves directly manipulating the human skeleton by cutting and combining atomic actions. Specifically, we extract two non-overlapping human skeletons, denoted as $\mathbf{x}^u$ and $\mathbf{x}^l$, from a frame corresponding to two different atomic actions, u and l, respectively. These skeletons are then spliced together to

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

TABLE II
COMPARISON OF THE CHAMP DATASET WITH SOME PUBLISHED DATASETS FOR HUMAN PERCEPTION

Datasets	Seq.	Test Seq.	C.A.	Muti-view	RGB	Depth	Joint	Action Label	Apps.	Sensors	Year
CAD-60 [52]	60	30	-	-	✓	✓	✓	✓	HAR	Kinect v1	2011
MSRDailyActivity3D [53]	320	160	-	-	✓	✓	✓	✓	HAR	Kinect v1	2012
G3D [54]	210	70	-	-	✓	✓	✓	✓	HMP	Microsoft Kinect device	2012
CAD-120 [55]	120	60	-	-	✓	✓	✓	✓	HAR	Kinect v1	2013
Multiview 3D Event [56]	3,815	-	-	✓	✓	✓	✓	✓	HAR	Kinect v1	2013
Human3.6M [43]	839	30	-	✓	✓	✓	✓	✓	HMP, HPE	10 Vicon T40	2014
Office Activity [57]	1,180	1,180	-	✓	✓	✓	-	✓	HAR	Kinect v1	2014
CMU Mocap	2,605	26	-	-	-	-	✓	✓	HAR, HMP	38 infrared cameras	2015
WBHM [58]	3,704	-	-	-	✓	-	✓	✓	HMP, HMG	56 Vicon MX	2015
SYSUDHOI [59]	480	240	-	✓	✓	✓	✓	✓	HAR	Kinect V1	2017
3DPW [47]	60	24	-	-	✓	-	✓	-	HMP, HPE	A hand-held smartphone camera	2018
FNTU [60]	18,000	6,101	-	✓	-	-	✓	✓	HMP	Kinect V2	2019
Amass [44]	>11,000	3,062	-	✓	-	-	✓	-	HMP	OptiTrack mocap system	2019
COUCH [61]	>500	120	-	-	✓	✓	✓	✓	HMG	IMUs and Kinect devices	2022
CIRCLE [62]	>7,000	453	-	✓	✓	✓	-	✓	HMG	a VR headset	2023
CHAMP	**>20,000**	**9,240**	✓	✓	✓	✓	✓	✓	HMP	Kinect V2	2024

"Seq." denotes the number of sequences, "C.A." denotes composite actions, "Apps."denotes applications.

generate a new frame, denoted as $\hat{\mathbf{x}}$. By performing this operation every frame, we can get a new sequence of composite actions $\hat{\mathbf{X}}_{1:N} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \cdots, \hat{\mathbf{x}}_N]$. However, this fundamental approach is undeniably simplistic. Despite the removal of global translation for each human pose, the resulting posture $\hat{\mathbf{x}}$ is prone to misplacement and inconsistency with the typical structure of the human body. Due to this, we utilize an autoencoder as an alternative to this basic method for composite motion synthesis. Based on a Bi-LSTM AutoEncoder, our motion synthesis consists of two steps: model training and motion synthesis.

1) Model Training: In the model training process, we expect that the Bi-LSTM AutoEncoder can reconstruct the sequences of atomic actions as much as possible. We first divide the human body skeleton into two parts, for each atomic action sequence $\mathbf{A}_{1:N+T}$, we use $\mathbf{A}^u_{1:N+T} = \mathbf{a}^u_1, \mathbf{a}^u_2, \cdots, \mathbf{a}^u_{n+t}$ and $\mathbf{A}^l_{1:N+T} = \mathbf{a}^l_1, \mathbf{a}^l_2, \cdots, \mathbf{a}^l_{n+t}$ to represent sequences of the two different body parts, then feed them into two different encoders and obtain their latent attributes:

$$\begin{aligned} \mathbf{h}^u &= \rho_{\phi_1}(\mathbf{a}^u_1, \mathbf{a}^u_2, \cdots, \mathbf{a}^u_{n+t}) \\ \mathbf{h}^l &= \rho_{\phi_2}(\mathbf{a}^l_1, \mathbf{a}^l_2, \cdots, \mathbf{a}^l_{n+t}) \end{aligned} \tag{3}$$

where ρ_{ϕ_1} and ρ_{ϕ_2} denote two different encoders, each of which is defined to be a Bi-LSTM [63]. The two encoders represent a sequence of an entire human body into two latent attributes $\mathbf{h}^u$ and $\mathbf{h}^l$, each of which contains the information of a sequence of a body part. The number of features in the hidden state of a Bi-LSTM is set to 100.

The decoder aims to reconstruct atomic actions $\hat{\mathbf{A}} = \hat{\mathbf{a}}_1, \cdots, \hat{\mathbf{a}}_{n+t}$ of an entire human body from atomic actions of two body parts using $\mathbf{h}^u$ and $\mathbf{h}^l$:

$$(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2, \cdots, \hat{\mathbf{a}}_{n+t}) = \rho_{\phi_3}([\mathbf{h}^u \parallel \mathbf{h}^l]) \tag{4}$$

where ρ_{ϕ_3} denotes the decoder, which is also a Bi-LSTM model. We concatenate the latent attributes output from the encoders and feed it into the decoder. To measure the reconstruction effect of the model, we calculate the reconstruction loss by making use of the Mean Per Joint Position Error (MPJPE) proposed in [43]:

$$l_r = \frac{1}{J(N+T)} \sum_{t=1}^{N+T} \sum_{j=1}^{J} \|\hat{\mathbf{a}}_{t,j} - \mathbf{a}_{t,j}\|^2 \tag{5}$$

where $\hat{\mathbf{a}}_{t,j} \in \mathbf{R}^3$ represent the 3D coordinates of the j^{th} joint of the t^{th} motion pose in $\hat{\mathbf{A}}_{1:N+T}$ and $\mathbf{a}_{t,j} \in \mathbf{R}^3$ is the corresponding ground truth.

2) Motion Synthesis: The subsequent step involves motion synthesis, with the objective of obtaining novel categories of composite actions. As discussed, a composite action can

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

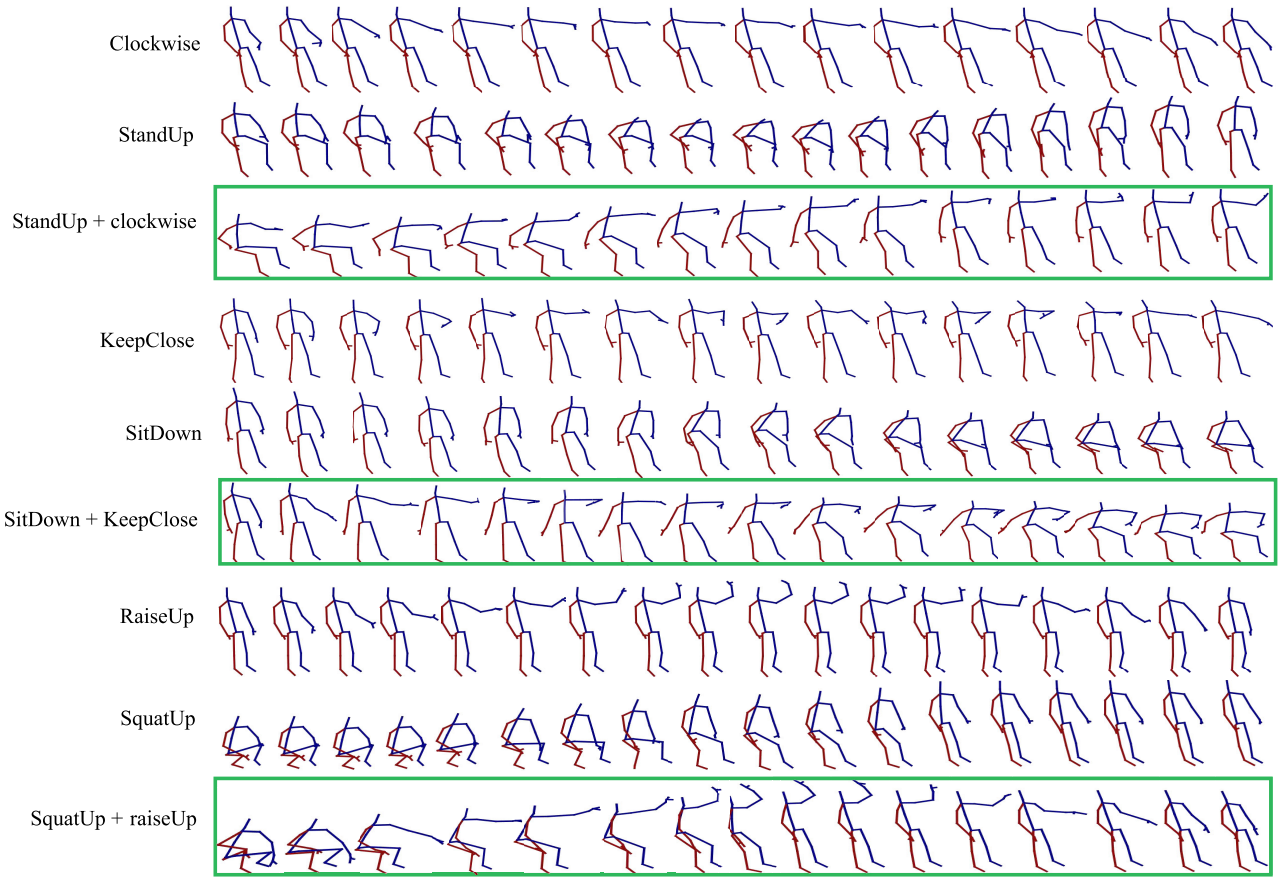


Fig. 3. Some examples of the CHAMP dataset. The green boxes show the composite actions while the rest show the atomic actions.

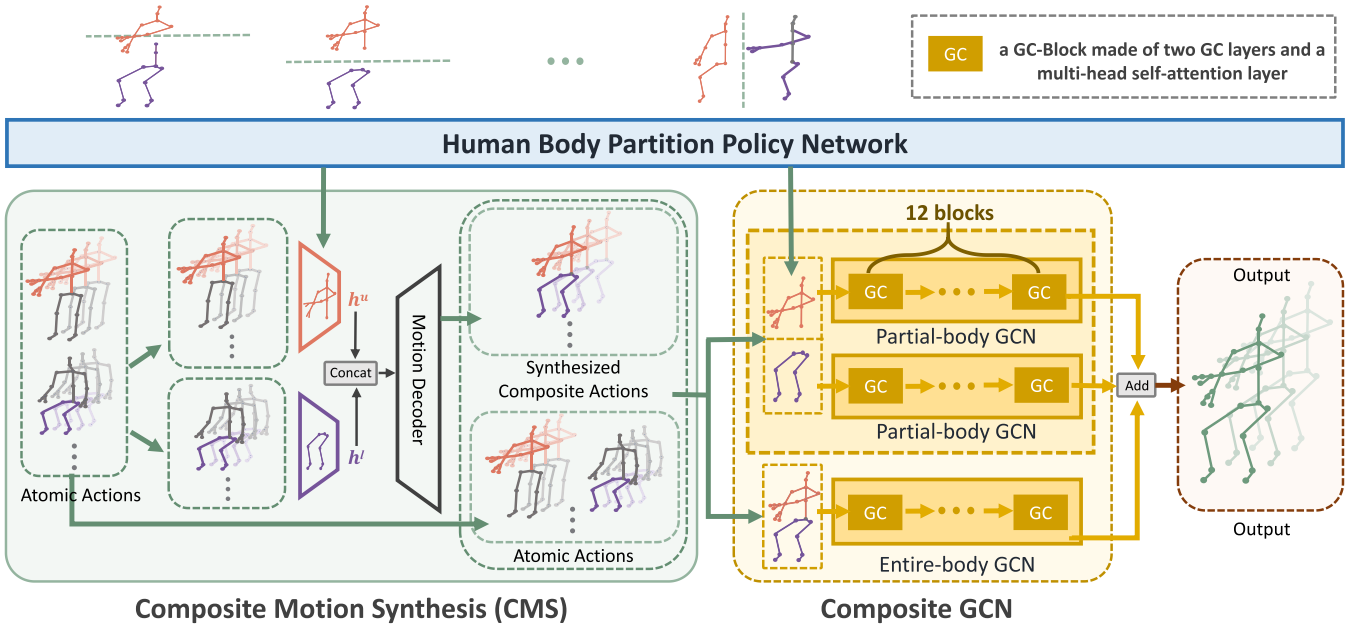


Fig. 4. Overview of our composite motion prediction framework. **Composite Motion Synthesis Module:** based on a Bi-LSTM AutoEncoder, we generate characterized composite actions utilizing only atomic actions. **Composite GCN Network:** two of the three branches of the predictor model the partial human skeleton and the third branch models the entire human skeleton.

be combined into two atomic actions that are done by non-overlapping body parts at one time in our CHAMP dataset. Utilizing the Bi-LSTM AutoEncoder, we fuse two latent attributes from different atomic actions to reconstruct their corresponding composite actions.

In detail, for each atomic action, we divide the human body skeleton into two parts the same as the model training process, and represent their sequences as $\mathbf{h}^u$ and $\mathbf{h}^l$ employing the two encoders. Next, given any two atomic actions, we take two latent attributes from different body parts and

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

TABLE III
PREDICTION RESULT OF 40 CLASSES OF COMPOSITE ACTIONS IN 3D JOINT COORDINATES

	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms
scenarios	sitDown+clockwise				sitDown+counterclockwise				sitDown+keepClose				sitDown+keepFar			
Hisrep	56.43	92.12	114.15	123.70	59.82	101.23	126.92	136.82	57.16	90.52	114.84	128.69	53.79	89.70	120.01	132.84
PGBIG	61.48	107.71	141.56	157.22	57.05	106.49	142.81	160.08	58.19	108.32	146.48	165.15	56.31	101.20	136.81	154.49
Ours	**52.69**	**86.66**	**106.60**	**113.92**	**55.66**	**91.98**	**113.87**	**122.41**	**53.21**	**81.25**	**98.33**	**107.22**	**51.50**	**84.49**	**107.48**	**114.07**
scenarios	sitDown+left				sitDown+nod				sitDown+right				sitDown+shake			
Hisrep	60.34	103.14	130.21	139.92	44.78	74.93	88.36	95.37	53.38	92.49	119.31	129.70	43.31	76.12	94.08	102.48
PGBIG	56.18	105.38	140.86	158.40	36.49	66.90	81.30	88.08	54.36	100.54	132.20	147.07	29.99	61.88	82.07	90.56
Ours	**55.42**	**91.68**	**116.04**	**122.69**	43.37	72.98	**79.98**	**80.98**	**50.37**	**86.84**	**112.88**	**121.37**	39.67	68.78	83.33	**87.96**
scenarios	sitDown+wave				squatUp+clockwise				squatUp+counterclockwise				squatUp+keepClose			
Hisrep	54.89	101.74	133.17	145.52	60.58	99.71	120.09	125.53	70.64	107.60	127.72	133.45	60.22	95.06	113.76	117.38
PGBIG	52.91	104.14	143.69	164.18	66.25	107.29	129.27	135.43	69.08	109.62	132.01	136.75	63.56	99.93	123.59	126.23
Ours	**50.70**	**90.64**	**115.82**	**126.64**	**55.23**	**83.38**	**102.32**	**106.71**	**64.76**	**93.35**	**109.53**	**116.61**	**57.31**	**87.49**	**107.74**	**112.05**
scenarios	squatUp+keepFar				squatUp+left				squatUp+nod				squatUp+raiseUp			
Hisrep	53.08	88.86	106.25	110.29	72.24	111.98	133.29	136.96	52.49	84.52	99.19	102.73	46.91	85.14	113.05	120.03
PGBIG	61.96	102.23	126.81	132.63	67.89	114.67	139.44	143.97	45.47	74.29	94.27	101.52	66.01	116.48	132.17	130.65
Ours	**47.92**	**72.25**	**89.77**	**96.80**	**65.19**	**95.34**	**110.62**	**113.93**	48.81	**71.29**	**81.97**	**85.66**	**43.96**	**78.02**	**105.03**	**109.79**
scenarios	squatUp+right				squatUp+shake				squatUp+wave				squat+clockwise			
Hisrep	62.89	104.28	129.98	140.08	57.75	89.53	108.67	109.06	53.45	91.88	117.81	125.46	75.81	126.57	163.15	174.88
PGBIG	65.92	114.46	142.31	148.93	48.83	74.18	93.78	99.14	63.29	102.56	126.75	133.97	65.97	118.61	160.60	181.65
Ours	**58.57**	**94.43**	**117.79**	**125.13**	52.98	75.15	**90.44**	**90.65**	**49.59**	**77.79**	**95.23**	**103.66**	70.16	**113.15**	**143.47**	**155.39**
scenarios	squat+counterclockwise				squat+keepClose				squat+keepFar				squat+left			
Hisrep	76.66	122.47	151.41	161.38	70.16	113.85	149.45	160.92	66.20	116.73	150.02	160.83	69.72	115.81	145.67	157.07
PGBIG	64.15	116.71	152.47	170.66	63.41	119.06	165.11	186.92	59.43	113.93	159.62	182.70	61.87	115.60	156.51	177.69
Ours	71.52	**107.41**	**129.86**	**140.59**	66.01	**102.59**	**131.37**	**142.77**	62.35	**107.95**	**136.75**	**146.88**	63.93	**101.91**	**126.71**	**138.00**
scenarios	squat+nod				squat+raiseUp				squat+right				squat+shake			
Hisrep	53.65	90.60	105.56	110.51	47.92	89.95	121.95	132.86	71.76	121.05	157.94	171.00	46.03	95.19	129.95	148.09
PGBIG	40.68	79.33	101.05	109.36	56.94	123.22	171.98	192.76	62.44	118.04	157.97	177.56	36.69	77.57	105.56	118.94
Ours	49.36	81.10	**97.41**	**102.72**	**46.84**	**89.66**	**120.02**	**131.96**	65.81	**107.04**	**136.31**	**150.43**	41.57	81.97	110.87	124.89
scenarios	squat+wave				standUp+clockwise				standUp+counterclockwise				standUp+keepClose			
Hisrep	68.38	121.95	158.88	168.25	55.58	81.79	96.89	102.31	57.65	85.30	100.59	104.68	52.45	85.19	106.24	114.60
PGBIG	63.97	127.71	175.47	199.19	59.16	97.86	115.61	123.10	60.19	98.13	116.70	124.72	55.98	96.96	122.19	130.94
Ours	**62.38**	**103.89**	**133.36**	**146.83**	**52.59**	**73.26**	**88.95**	**94.86**	**54.30**	**77.42**	**92.10**	**96.54**	**51.52**	**80.38**	**99.11**	**103.24**
scenarios	standUp+keepFar				standUp+left				standUp+nod				standUp+raiseUp			
Hisrep	56.06	89.40	107.81	114.76	56.30	88.86	103.20	105.77	43.53	76.19	89.77	93.74	39.74	83.53	111.80	115.16
PGBIG	56.34	97.59	121.08	130.00	56.19	98.62	118.54	124.72	35.56	61.25	74.76	78.04	53.52	106.84	130.86	135.46
Ours	**53.10**	**79.48**	**96.17**	**102.53**	**52.25**	**81.20**	**94.13**	**94.54**	42.40	70.87	76.35	**73.12**	**32.71**	**61.44**	**84.80**	**90.44**

scenarios	standUp+right				standUp+shake				standUp+wave				sitDown+raiseUp				Average			
	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms	40ms	100ms	160ms	200ms
Hisrep	53.24	88.39	107.52	111.77	42.82	70.20	79.06	81.31	57.30	98.22	122.16	128.24	38.79	76.62	105.59	112.97	56.85	95.46	119.39	127.18
PGBIG	56.18	98.53	119.45	124.46	35.97	60.07	73.83	79.50	56.97	105.10	135.18	146.95	49.46	101.74	138.40	151.83	55.81	100.27	129.03	140.54
Ours	**50.11**	**81.69**	**98.25**	**97.79**	40.98	64.26	**68.11**	**65.09**	**51.87**	**82.95**	**102.52**	**105.79**	**35.82**	**68.86**	**93.20**	**100.43**	**52.86**	**85.06**	**105.11**	**111.58**

concatenate them as the input to the decoder. For example, we concatenate $\mathbf{h}^u$ from action m and $\mathbf{h}^l$ from action n, and acquire the corresponding sequence of composite action $\mathbf{C}^{mn}_{1:N+T} = (\mathbf{c}^{mn}_1, \mathbf{c}^{mn}_2, \cdots, \mathbf{c}^{mn}_{n+t})$:

$$(\mathbf{c}^{mn}_1, \mathbf{c}^{mn}_2, \cdots, \mathbf{c}^{mn}_{n+t}) = \rho_{\phi_3}([\mathbf{h}^{m,u} \parallel \mathbf{h}^{n,l}]) \tag{6}$$

This way, we synthesize a sequence of composite action " m meanwhile n". This operation is performed in pairs of different atomic actions to synthesize corresponding composite actions and fill in the gap in the training set. We synthesize 10 upper-body atomic actions and 4 lower-body atomic actions in pairs and acquire 40 classes of composite actions.

3) Human Body Partition Policy Network: Since a specific composite action is combined with two atomic actions in the CHAMP dataset, we divide the human skeleton into two parts before feeding motion sequences into the encoders, and there are varied partitions. We design a policy module to select the most appropriate partition automatically. We construct several representative body partitions and define a trainable probability parameter $\mathbf{p}_{bp} \in \mathbf{R}^L$, representing the probabilities of different body partitions.

Given the probabilities, we obtain the body partitions strategy b through $b = \arg\max(p_\mathrm{b})$. However, since arg max operation is not differentiable, we utilize the Straight-Through (ST) Gumbel Estimator [22] as a substitute for arg max during the training of the probabilities:

$$b = \text{Gumbel-SoftMax}(\mathbf{p}_\mathrm{b}) \tag{7}$$

In detail, Gumbel-Max selects a body partitions strategy b uisng strategy probability $\mathbf{p}_{\mathrm{b},i}$ in the forward process:

$$b = \arg\max_i(\log \mathbf{p}_{\mathrm{b},i} + G_i) \tag{8}$$

where $G_i = -\log(-\log U_i)$ is a standard Gumbel distribution, where $U_i \sim Uniform(0, 1)$. In order to back-propagate the gradient to the Human body Partition Policy Network, Gumbel-softmax is used to calculate the approximation of b:

$$\tilde{b} = \frac{\exp(\log \mathbf{p}_{\mathrm{b},i} + G_i)/\tau}{\sum_{j=0}^{L} \exp(\log \mathbf{p}_{\mathrm{b},j} + G_j)/\tau} \tag{9}$$

where we denote $\hat{b} = one_hot(b)$ as a one-hot vector and treat $\tilde{b}$ as the continuous approximation of $\hat{b}$. The temperature parameter τ is initialized as a constant 0.1, making it similar to Gumbel-Max, and the samples from Gumbel-SoftMax are close to one-hot. The chosen body partition strategy bp is used in both CMS and the Composite GCN network, guiding the division of the human skeleton.

D. Composite GCN

Modeling the complete human skeleton for training atomic actions results in redundancy, as only a limited number of

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

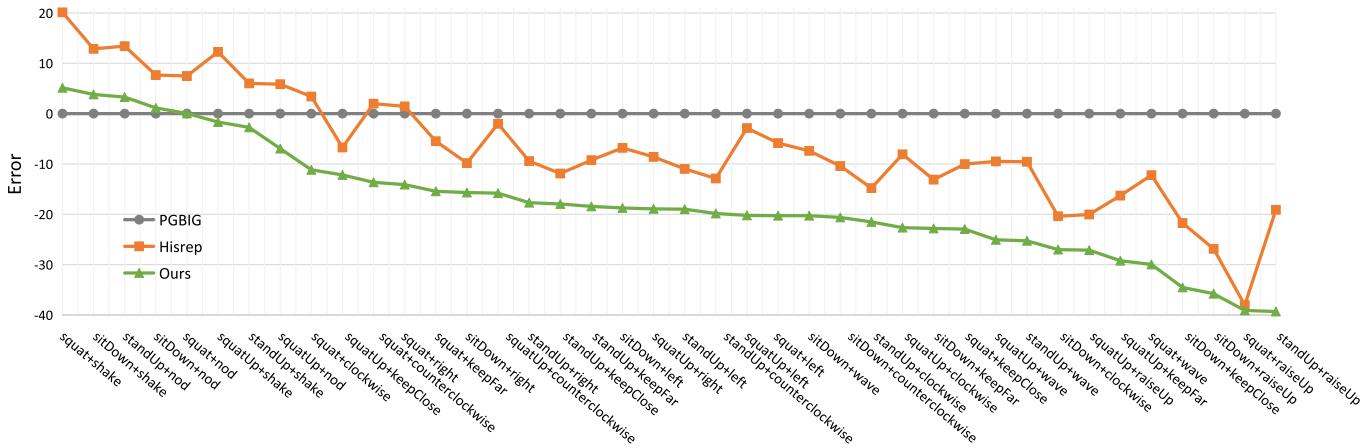


Fig. 5. Advantage analysis of composite actions on the CHAMP dataset. Using PGBIG as a baseline, we show the mean prediction error per action of PGBIG [20] subtracted from Hisrep and our method. The advantage of our approach is most evident in the action of "standUp+raiseUp".

skeletal components are essential for these actions. For example, when doing the atomic action "sit down", the movement of the legs is more important to the formation of this movement than the upper body. But it is unsuitable to model the human skeleton in parts for synthesized composite actions as it compromises its integrity. Therefore, we need two kinds of strategies to model these two types of actions.

We are inspired by Hisrep [16] and design our predictor as a GCN-based network, capturing spatial dependencies among human skeleton joints. Following the motion attention model in Hisrep, we first find the most relevant past sub-sequence by measuring the similarity between the last observed sub-sequence and a set of previous sub-sequences. We then represent the sub-sequences as Discrete Cosine Transform (DCT) coefficients and input them into the predictor.

To model the spatial dependencies between different joint coordinates in different parts of the human skeleton, we design a three-branch predictor using trainable adjacency matrices of different sizes and take the human body as three kinds of fully-connected graphs. The graph convolutional layers of each branch's output matrices are in the form:

$$\mathbf{H}^{(p+1)} = \sigma(\mathbf{A}^{(p)}\mathbf{H}^{(p)}\mathbf{W}^{(p)}) \tag{10}$$

where $\mathbf{H}^{(p+1)}$ can be represented as the convolved output matrix, and $\mathbf{H}^{(p)}$ can be represented as the input matrices $\mathbf{H}_u^{(p)} \in \mathbf{R}^{U\times F}$, $\mathbf{H}_l^{(p)} \in \mathbf{R}^{L\times F}$ or $\mathbf{H}_e^{(p)} \in \mathbf{R}^{E\times F}$ of three branches, which denote U, L or E trajectories with F features. The size U and L depend on the body partition strategy given by the partition policy module, while the third branch models the entire human skeleton with E joint coordinates. Each $\mathbf{A}^{(p)}$ is the trainable adjacency matrices of layer p, representing the strength of the connectivity between nodes. And their size also depends on the branches' partition strategy. $\mathbf{W}^{(p)} \in \mathbf{R}^{F\times\hat{F}}$ encodes trainable weights with the size of 256 × 256, and $\sigma(\cdot)$ is the activation function $tanh(\cdot)$.

For each branch in the predictor, we use the same GCN with residual structure as in Hisrep [16]. There are 12 residual blocks in each branch, each of which contains two graph convolutional layers, and two additional layers for encoding and decoding DCT coefficients. In addition, we add a multi-head self-attention layer and place it at the end of every residual block in the predictor. This attention layer aims to highlight the important features of each joint coordinate. The feature $\mathbf{f} \in \mathbf{R}^{d_{model}}$ of the output of the graph convolutional layer is passed through the attention layer fused to obtain new feature $\dot{\mathbf{f}}$. We use 4 parallel self-attention modules ($h = 4$) to construct an attention layer.

Specifically, the feature $\mathbf{f}$ is multiplied by three matrices $\mathbf{W}_i{}^Q \in \mathbf{R}^{d_{model}\times d_q}$, $\mathbf{W}_i{}^K \in \mathbf{R}^{d_{model}\times d_k}$, $\mathbf{W}_i{}^V \in \mathbf{R}^{d_{model}\times d_v}$ to get the matrices $\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i$:

$$\mathbf{Q}_i = \mathbf{f}\times\mathbf{W}_i^Q, \mathbf{K}_i = \mathbf{f}\times\mathbf{W}_i^K, \mathbf{V}_i = \mathbf{f}\times\mathbf{W}_i^V \tag{11}$$

where the feature dimensions d_q, d_k, d_v of each head are set to 32 and the $\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i$ matrices belong to the i^{th} self-attention module. The output feature $\dot{\mathbf{f}}_i$ of the i^{th} self-attention module can be obtained according to the formula:

$$\dot{\mathbf{f}}_i = \text{softmax}(\frac{\mathbf{Q}_i\mathbf{K}_i{}^T}{\sqrt{d_k}})\mathbf{V}_i \tag{12}$$

where the attention matrix $\mathbf{A} = \mathbf{Q}_i\mathbf{K}_i{}^T$ and is divided by the square root of d_k to prevent the attention fraction from increasing as the dimension increases. Finally, the outputs $\dot{\mathbf{f}}_i$ are concatenated and dotted with the weight $\mathbf{W}_O \in \mathbf{R}^{d_v\times d_{model}}$ to get the new feature $\dot{\mathbf{f}}$:

$$\dot{\mathbf{f}} = \text{concat}(\dot{\mathbf{f}}_1, \ldots, \dot{\mathbf{f}}_4)\mathbf{W}_O \tag{13}$$

The resulting feature $\dot{\mathbf{f}}$ then serves as input for the subsequent residual GC block.

IV. Experiments

The experiment is divided into two parts: (a) composite human motion prediction task and (b) common human motion prediction task. The main purpose of our paper is to solve the composite human motion prediction task. In this task, we evaluate the performance of our method on our CHAMP dataset. Since it is the first time to present this task in the literature, there is no available approach to compare with. We assess the state-of-the-art methods in common human motion prediction tasks and consider Hisrep [16] and PGBIG [20] as comparisons, as they have become baseline methods used in newly published approaches.

Furthermore, to demonstrate the generalizability of our model, we also present the results of the common human

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

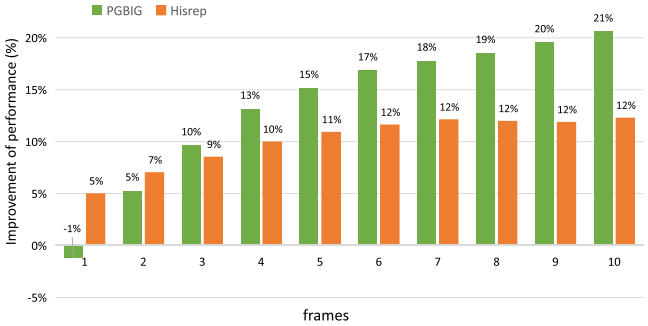


Fig. 6. Percentages of improvement of composite actions on the CHAMP dataset, comparing our method with PGBIG [20] and Hisrep [16] at different forecast frames. Our method is 5% to 21% better than the other two except for the first frame compared to PGBIG.

motion prediction tasks in comparison to Hisrep and PGBIG, utilizing the Human3.6M dataset.

A. Experiment Settings

1) Composite Human Motion Prediction Task: We use the CHAMP dataset in the evaluation of the composite human motion prediction task. As introduced in Section III, for our CHAMP dataset, we use 15 classes of atomic actions and 40 classes of composite actions in our evaluation. 15 classes of atomic actions performed by 21 subjects are used as training data, where the labels are specified as: "still", "sitDown", "standUp", "squat", "squatUp", "clockwise", "counterclockwise", "keepClose", "keepFar", "left", "right", "nod", "shake", "wave", and "raiseUp".

Validation data contains 40 classes of composite actions performed by 11 subjects. Specifically, we use composite actions performed by subjects 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, and 22 as validation data. As for test data, the rest of the composite actions performed by subjects 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, and 21 are included in the test data, as well as 15 classes of atomic actions performed by subject 2. We follow the data processing of other commonly used datasets and remove the global translation of each human pose.

2) Common Human Motion Prediction Task: The Human3.6M dataset [43] is widely used for the common human motion prediction task, with consistent action categories in both training and testing. Consistent with previous work [20], we use S1, S6, S7, S8, and S9 for training, S5 for testing,and S11 for validation. To ensure fair evaluation, global translations of each human pose are eliminated, and we report results on all sequences of S5 without any sampling.

3) Metrics: We report the results of the 3D joint coordinates on both the CHAMP dataset and the Human3.6M dataset and show the Mean Per Joint Position Error (MPJPE) in millimeters introduced in Section III.

4) Implementation Details: On the CHAMP dataset, We train our Composite Motion Synthesis module for 100 epochs and the Composite Graph Convolutional Network (GCN) module for 50 epochs. For the Composite GCN module, our method is trained with 20 frames of input on the CHAMP dataset and 50 frames on the Human3.6M dataset and forecasts 10 frames of future poses. The learning rate for the CMS module is set to 0.00001, while the predictor uses a learning rate of 0.00025 with a 0.96 decay every epoch. Both the CMS module and the predictor have a batch size of 32. Our code is in Pytorch and uses ADAM [64] as an optimizer. All the models are trained and tested on an NVIDIA RTX 3080 Ti GPU.

TABLE IV
PREDICTION RESULTS OF REPRESENTATIVE ATOMIC ACTIONS AND THE AVERAGE ERRORS OF ALL ATOMIC ACTIONS

	100ms	200ms	100ms	200ms	100ms	200ms	100ms	200ms	100ms	200ms
scenarios	keepFar		clockwise		standUp		wave		Average	
Hisrep	65.74	126.05	60.90	96.78	71.25	77.89	29.30	56.35	45.11	70.40
PGBIG	52.16	107.67	54.47	77.50	66.08	65.70	29.74	69.80	48.69	68.66
Ours	**46.91**	**87.51**	**48.43**	**67.45**	**62.62**	**61.28**	**25.31**	**55.92**	**39.95**	**65.82**

B. Comparison With State-of-the-Arts

We compared our method with Hisrep [16] and PGBIG [20] on both the Composite Human Motion Prediction Task and Common Human Motion Prediction Task.

1) Composite Human Motion Prediction Task: Table III and Table IV report the results of all classes of composite actions and some representative atomic actions in the CHAMP dataset. Because of the narrow time span of actions in the CHAMP dataset, we predict and show results for the next 200 milliseconds. The symbol "+" means a person is doing one action while doing another action. In Figure 5 We use a line plot to show the comparison of predictive errors for composite actions. We treat PGBIG as the baseline and subtract the average prediction errors per action of PGBIG from those of Hisrep and our method. It can be seen that Hisrep performs better than PGBIG in the majority of composite actions, and our proposed method outperforms other methods for most composite actions, except for "squat + shake", "sitDowm + shake", "standUp + nod", "sitDown + nod", which involve small amplitude of movement in the upper body. Our method works best on actions with gross amplitude of movement in the upper body such as "standUp + raiseUp". It shows that our model has advantages in actions with large movements compared to those with relatively stationary movements. We also calculate the percentages of improvement of the composite actions and compare our method with PGBIG and Hisrep. As shown in Figure 6, except for the first frame compared with PGBIG, our methods outperform the other two methods by 5% to 21%, and this advantage becomes more pronounced as the number of forecast frames increases.

Figure 7 shows visualization results of prediction samples "sitDown + KeepClose", and "squatUp + Clockwise". In the visualization of "sitDown + KeepClose", we notice that both arms and legs' movements get better results on our method, especially in the last few frames. The legs in our results are already completely seated the same as the ground truth, while the legs in the results of PGBIG and Hisrep are still not fully bent. The same is true in the action "squatUp + Clockwise", where the legs in our results are closer to the ground truth. We infer that there are two reasons for the better results of our method. The first is that we solve the data insufficiency issue. New classes of actions were synthesized from atomic actions to increase data size and action categories of the training set in our methods, while Hisrep and PGBIG have no such motion synthesis module. Second, there are multiple

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

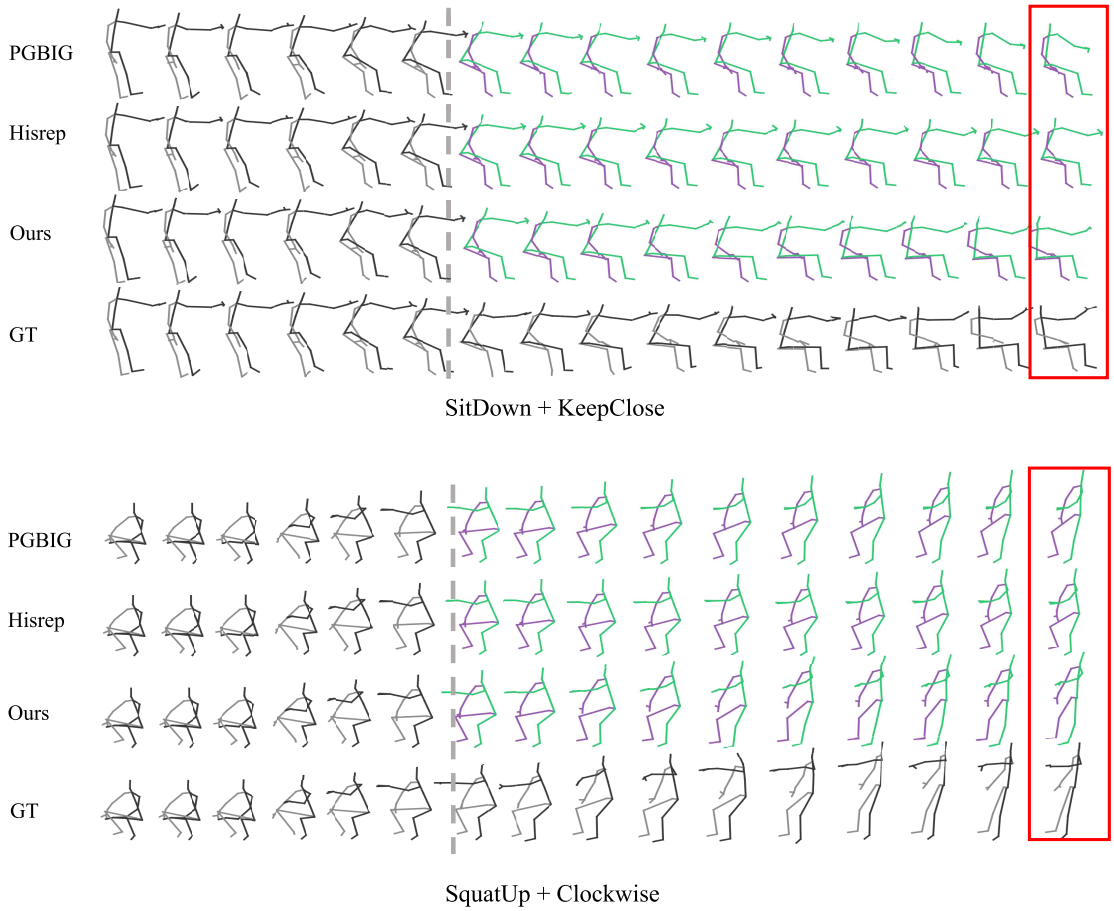


Fig. 7. Comparison of our visualization results with Hisrep and PGBIG [20]. Here we use the action "SitDown + KeepClose" and "squatUp + Clockwise" as examples. The ground truth is shown as black-grey skeletons and the predictions as green-purple. The red boxes mark the frame where our method is closest to ground truth.

TABLE V
PREDICTION RESULTS OF REPRESENTATIVE ACTIONS AND THE AVERAGE ERRORS OF ALL 15 ACTIONS ON HUMAN3.6M

	320ms	400ms	320ms	400ms	320ms	400ms	320ms	400ms	320ms	400ms
scenarios	smoking		purchases		walkingdog		walkingtogether		Average	
Hisrep	27.98	34.39	60.57	73.75	73.72	86.22	33.73	40.01	47.72	58.78
PGBIG	27.85	34.65	59.78	73.61	74.26	88.25	33.96	41.21	46.93	58.43
Ours	**27.29**	**33.74**	**58.19**	**71.06**	**70.94**	**84.16**	**32.49**	**38.37**	**46.26**	**57.30**

branches modeling different body parts, which better extract the key features of the atomic action with less interference from unimportant body parts.

2) Common Human Motion Prediction Task: Table V reports the results of four representative actions "smoking", "purchases", "walkingdog", and "walkingtogether" as well as the average MPJPE of all 15 actions in the short-term forecast of the Human3.6M dataset. Our method is superior to other methods in average action error on the scales of 320ms and 400ms, which shows that our method achieves good performance on common human motion prediction tasks using the Human3.6M dataset.

As shown in Table VII, it provides an overview of the average prediction results on both the Composite Human Motion Prediction Task using the CHAMP dataset and the Common Human Motion Prediction Task using the Human3.6M dataset. Each row represents a different method, and the columns represent different prediction time intervals for composite and atomic actions. From the table, it is evident that our proposed method consistently outperforms existing approaches across all tasks and datasets.

C. Robustness Across Varied Data Scarcity

We conduct additional experiments usingthe CHAMP dataset to further explore the robustness of the models under different data scarcity scenarios. We introduce two new experimental settings, denoted as setting 2 and setting 3, in addition to the original setting 1, where the training set does not contain any composite actions. In setting 2, a small number of composite actions from subjects 2 and 4 in the validation set are added to the training set, while the test set remains unchanged. In setting 3, a larger number of composite action samples from ten subjects in the validation set are included in the training set, maintaining the same test set configuration.

The results presented in Table VI illustrate that our model consistently outperforms existing methods across all settings, indicating its robust predictive performance in diverse conditions. This additional analysis enhances the comprehensiveness of our evaluation and further reinforces the effectiveness of our proposed framework.

D. Ablation Studies

1) Body Partition Strategy: As discussed in Section III, we divide the human skeleton into 2 parts since the CHAMP

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

TABLE VI

AVERAGE PREDICTION RESULTS OF THREE SETTINGS IN 3D JOINT COORDINATES. RESULTS ARE SHOWN AT 100MS, AND 200MS IN THE FUTURE

	setting 1			setting 2			setting 3		
	100ms	200ms	average	100ms	200ms	average	100ms	200ms	average
Hisrep [16]	81.73	111.69	86.49	81.15	110.21	85.61	81.19	110.97	85.89
PGBIG [20]	86.20	120.94	91.54	82.13	128.89	91.91	80.87	127.99	91.01
Ours	**72.75**	**99.10**	**77.27**	**76.26**	**98.91**	**79.58**	**74.27**	**101.29**	**78.74**

TABLE VII

AVERAGE PREDICTION RESULTS ON COMPOSITE HUMAN MOTION PREDICTION TASK (USING CHAMP DATASET), AND COMMON HUMAN MOTION PREDICTION TASK (USING HUMAN3.6M DATASET)

	CHAMP				Human3.6M	
	composite actions		atomic actions			
	100ms	200ms	100ms	200ms	320ms	400ms
Hisrep [16]	95.46	127.18	45.11	70.40	47.72	58.78
PGBIG [20]	100.27	140.54	48.69	68.66	46.93	58.43
Ours	**85.06**	**111.58**	**39.95**	**65.82**	**46.26**	**57.30**

TABLE VIII

ABLATIONS ON THE CHAMP DATASET. AVERAGE LOSS OF 3D JOINT COORDINATES. THE ERROR IS MEASURED IN MILLIMETERS

	40ms	100ms	160ms	200ms	average
3-parts	45.73	75.11	94.74	101.65	79.31
5-parts	46.25	77.24	97.06	108.00	82.14
replace C-GCN by hisrep	45.66	76.52	99.48	114.92	84.15
w/o CMS	50.35	80.72	102.01	107.83	85.23
simple CMS	46.60	76.21	95.92	103.21	80.49
full model	**44.73**	**72.75**	**92.51**	**99.10**	**77.27**

TABLE IX

ABLATIONS ON HUMAN3.6M. AVERAGE LOSS OF 3D JOINT COORDINATES. THE ERROR IS MEASURED IN MILLIMETERS

	80ms	160ms	320ms	400ms	average
replace C-GCN by hisrep	10.53	22.65	47.23	58.16	34.64
w/o CMS	10.07	22.11	46.87	57.86	34.23
simple CMS	9.91	21.99	46.90	57.99	34.20
full model	**9.98**	**21.85**	**46.26**	**57.30**	**33.85**

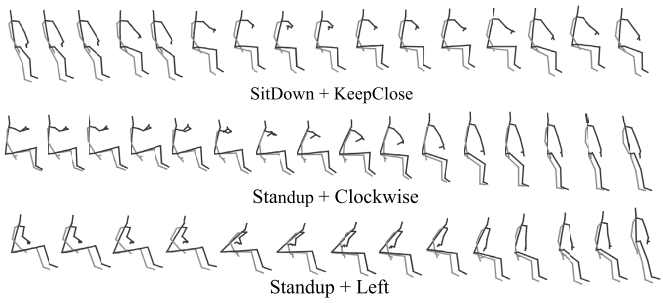


Fig. 8. Visualization of composite actions generated by the CMS module.

dataset combines two atomic actions into corresponding composite actions. However, this 2-part body partition strategy is based on our intuition and the human skeleton can be divided into more parts in motion synthesis and prediction processes. Thus, we try and separate the human skeleton into 3 parts (trunk, upper arms, and lower limbs) and 5 parts (trunk, left arm, right arm, left leg, and right leg), and conduct the experiment on the CHAMP dataset. In more detail, the 3-part strategy uses 3 atomic actions to reconstruct a corresponding composite action in the CMS module, and a 4-branch predictor is set to model and forecast the three skeleton parts and the whole skeleton separately. Our method has similar variations in the 5-part strategy. As shown in Table VIII, the average prediction results increase from 77.27 to 79.7 and 82.14 when the body partition strategy changes into 3-part and 5-part. By inference separating the human skeleton into more parts does not lead to better performance and the 2-part body partition strategy still gains the best result in the CHAMP dataset.

2) Architecture: To better evaluate the influence of our proposed components, we do the experiments which are (a) replace the Composite GCN (C-GCN) Network by hisrep. (b) remove the Composite Motion Synthesis (CMS) module. (c) as discussed in Section III, there is a simple way to generate composite motions, which is to cut and connect the human skeleton of atomic actions directly. In order to verify the influence of the autoencoder in the CMS module, we replace the CMS module with the simpler one.

The results on the CHAMP dataset and Human3.6M dataset are shown in Table VIII and Table IX. As we can see, using both two components turns out for the best on both datasets. (a) Replacing the C-GCN module by hisrep yields a sharp increase in the average prediction error from 77.27 to 84.15 on the CHAMP dataset and 33.85 to 34.64 on the Human3.6M dataset. (b) The errors increase to 85.23 on the CHAMP dataset and to 34.23 on the Human3.6M dataset after removing the CMS module. (c) Compared with no motion synthesis, using the simplified CMS module has better performance with the error reducing from 85.23 to 80.49 on the CHAMP dataset, while the CMS module with the autoencoder has the best prediction results. The experiments verify that both the Composite Motion Synthesis module and Composite GCN Network are effective in both the composite human motion prediction task and the common human motion prediction task.

Here we provide visual examples of the effectiveness of the CMS module in generating composite actions. Figure 8 shows the visualization of some composite actions generated by the CMS module. We take "StandUp + Left", "SitDown + KeepClose", "StandUp + Clockwise"as examples.

V. CONCLUSION

In this paper, we introduce a new composite human motion prediction task, which takes atomic actions as the training set and composite actions as the testing set. We collect a new dataset called Composite HumAn Motion Prediction dataset that is suitable for our task. Our proposed method consists of two parts. The first part synthesizes composite actions using only atomic actions, while the second part models both the partial and entire human skeleton and forecasts future poses using the GCN-based network. Our method outperforms baseline approaches (PGBIG, Hisrep), and experiments confirm the effectiveness of our proposed modules (Composite

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

Motion Synthesis module and Composite GCN Network) in the composite human motion prediction task. We also verify that our method has good performance on common human motion prediction tasks using the Human3.6M dataset.

REFERENCES

[1] M. Liu and J. Yuan, "Recognizing human actions as the evolution of pose estimation maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1159–1168.

[2] G. Zhu, L. Zhang, H. Li, and P. Shen, "Topology-learnable graph convolution for skeleton-based action recognition," *Pattern Recognit. Lett.*, vol. 135, pp. 286–292, Jul. 2020.

[3] Q. Men, E. S. L. Ho, H. P. H. Shum, and H. Leung, "A quadruple diffusion convolutional recurrent network for human motion prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 9, pp. 3417–3432, Sep. 2021.

[4] Y. Ji, Y. Yang, F. Shen, H. T. Shen, and W.-S. Zheng, "Arbitrary-view human action recognition: A varying-view RGB-D action dataset," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 289–300, Jan. 2021.

[5] H. Wang, B. Yu, J. Li, L. Zhang, and D. Chen, "Multi-stream interaction networks for human action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 5, pp. 3050–3060, May 2022.

[6] Q. Wang, G. Sun, J. Dong, Q. Wang, and Z. Ding, "Continuous multi-view human action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 6, pp. 3603–3614, Jun. 2022.

[7] G. Zhu, L. Yang, L. Zhang, P. Shen, and J. Song, "Recurrent graph convolutional networks for skeleton-based action recognition," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 1352–1359.

[8] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognit.*, vol. 68, pp. 346–362, Aug. 2017.

[9] N. Wang, G. Zhu, L. Zhang, P. Shen, H. Li, and C. Hua, "Spatio-temporal interaction graph parsing networks for human-object interaction recognition," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 4985–4993.

[10] Y. Yang, G. Liu, and X. Gao, "Motion guided attention learning for self-supervised 3D human action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8623–8634, Dec. 2022.

[11] W. Mao, M. Liu, M. Salzmann, and H. Li, "Learning trajectory dependencies for human motion prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9489–9497.

[12] Z. Liu et al., "Towards natural and accurate future motion prediction of humans and animals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10004–10012.

[13] X. Liu, J. Yin, J. Liu, P. Ding, J. Liu, and H. Liu, "TrajectoryCNN: A new spatio-temporal feature learning network for human motion prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 6, pp. 2133–2146, Jun. 2021.

[14] A. Li, S. Wang, W. Li, S. Liu, and S. Zhang, "Predicting human mobility with federated learning," in *Proc. 28th Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2020, pp. 441–444.

[15] E. Corona, A. Pumarola, G. Alenya, and F. Moreno-Noguer, "Context-aware human motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 6992–7001.

[16] W. Mao, M. Liu, and M. Salzmann, "History repeats itself: Human motion prediction via motion attention," in *Proc. Eur. Conf. Comput. Vision (ECCV)*, Aug. 2020, pp. 474–489.

[17] E. Aksan, M. Kaufmann, P. Cao, and O. Hilliges, "A spatio-temporal transformer for 3D human motion prediction," in *Proc. Int. Conf. 3D Vis. (3DV)*, Dec. 2021, pp. 565–574.

[18] L. Dang, Y. Nie, C. Long, Q. Zhang, and G. Li, "MSR-GCN: Multi-scale residual graph convolution networks for human motion prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11467–11476.

[19] P. Ding and J. Yin, "Towards more realistic human motion prediction with attention to motion coordination," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5846–5858, Sep. 2022.

[20] T. Ma, Y. Nie, C. Long, Q. Zhang, and G. Li, "Progressively generating better initial guesses towards next stages for high-quality human motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 6437–6446.

[21] J. Tang, J. Zhang, R. Ding, B. Gu, and J. Yin, "Collaborative multi-dynamic pattern modeling for human motion prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, pp. 3689–3700, Aug. 2023.

[22] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with Gumbel–Softmax," 2016, *arXiv:1611.01144*.

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

[24] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[25] M. Brand and A. Hertzmann, "Style machines," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, 2000, pp. 183–192.

[26] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–298, Feb. 2008.

[27] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4346–4354.

[28] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5308–5317.

[29] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2891–2900.

[30] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," 2014, *arXiv:1409.1259*.

[31] J. Butepage, M. J. Black, D. Kragic, and H. Kjellstrom, "Deep representation learning for human motion prediction and classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6158–6166.

[32] C. Li, Z. Zhang, W. S. Lee, and G. H. Lee, "Convolutional sequence to sequence model for human dynamics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5226–5234.

[33] Q. Cui and H. Sun, "Towards accurate 3D human motion prediction from incomplete observations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4801–4810.

[34] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian, "Dynamic multiscale graph neural networks for 3D skeleton based human motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 214–223.

[35] A. Vaswan et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2017, pp. 5998–6008.

[36] W. Mao, M. Liu, M. Salzmann, and H. Li, "Multi-level motion attention for human motion prediction," *Int. J. Comput. Vis.*, vol. 129, no. 9, pp. 2513–2535, Sep. 2021.

[37] H. Chen, J. Hu, W. Zhang, and P. Su, "Spatiotemporal consistency learning from momentum cues for human motion prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, pp. 4577–4587, Sep. 2023.

[38] M. A. Tanner and W. H. Wong, "The calculation of posterior distributions by data augmentation," *J. Amer. Statist. Assoc.*, vol. 82, no. 398, pp. 528–540, 1987.

[39] R. G. Lopes, D. Yin, B. Poole, J. Gilmer, and E. D. Cubuk, "Improving robustness without sacrificing accuracy with patch Gaussian augmentation," 2019, *arXiv:1906.02611*.

[40] B. K. Iwana and S. Uchida, "An empirical survey of data augmentation for time series classification with neural networks," *PLoS ONE*, vol. 16, no. 7, Jul. 2021, Art. no. e0254841.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[42] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8697–8710.

[43] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3D human sensing in natural environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1325–1339, Dec. 2013.

[44] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "AMASS: Archive of motion capture as surface shapes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Oct. 2019, pp. 5442–5451.

[45] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–16, Oct. 2015.

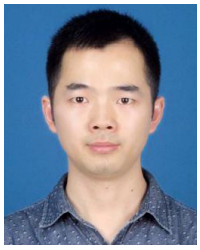
[46] G. Varol et al., "Learning from synthetic humans," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 109–117.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.

[47] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3D human pose in the wild using IMUs and a moving camera," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 601–617.
[48] W. Guo, X. Bie, X. Alameda-Pineda, and F. Moreno-Noguer, "Multi-person extreme motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 13053–13064.
[49] Z. Cao, H. Gao, K. Mangalam, Q.-Z. Cai, M. Vo, and J. Malik, "Long-term human motion prediction with scene context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 387–404.
[50] Z. Zhang, "Microsoft Kinect sensor and its effect," *IEEE Multimedia Mag.*, vol. 19, no. 2, pp. 4–10, Feb. 2012.
[51] J. Shotton et al., "Real-time human pose recognition in parts from single depth images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1297–1304.
[52] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from RGBD images," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 842–849.
[53] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1290–1297.
[54] V. Bloom, D. Makris, and V. Argyriou, "G3D: A gaming action dataset and real time action recognition evaluation framework," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 7–12.
[55] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from RGB-D videos," *Int. J. Robot. Res.*, vol. 32, no. 8, pp. 951–970, Jul. 2013.
[56] P. Wei, Y. Zhao, N. Zheng, and S.-C. Zhu, "Modeling 4D human-object interactions for event and object recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3272–3279.
[57] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo, "3D human activity recognition with reconfigurable convolutional neural networks," in *Proc. 22nd ACM Int. Conf. Multimedia*, Nov. 2014, pp. 97–106.
[58] C. Mandery, Ö. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "The kit whole-body human motion database," in *Proc. Int. Conf. Adv. Robot. (ICAR)*, 2015, pp. 329–336.
[59] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for RGB-D activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5344–5352.
[60] X. Liu, J. Yin, H. Liu, and Y. Yin, "PISEP^2: Pseudo-image sequence evolution-based 3D pose prediction," *Vis. Comput.*, vol. 38, no. 7, pp. 2603–2616, Jul. 2022.
[61] X. Zhang, B. L. Bhatnagar, S. Starke, V. Guzov, and G. Pons-Moll, "Couch: Towards controllable human-chair interactions," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 518–535.
[62] J. P. Araújo et al., "CIRCLE: Capture in rich contextual environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21211–21221.
[63] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[64] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.



Wanying Zhang received the B.Eng. degree in automation and the master's degree in mechanical engineering from the School of Intelligent Systems Engineering, Sun Yat-sen University, China, in 2021 and 2024, respectively. She is currently an Intern with Peking University Shenzhen Graduate School. Her research interests include skeleton-based human motion prediction.



Mengyuan Liu received the Ph.D. degree from the School of Electronics Engineering and Computer Science, Peking University, China, in 2017, under the supervision of Prof. Hong Liu. He was a Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, supervised by Prof. Junsong Yuan and Prof. Kai-Kuang Ma. He is currently an Assistant Professor with Peking University Shenzhen Graduate School. Related methods have been published in IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, *Pattern Recognition*, CVPR, ECCV, ACM MM, AAAI, and IJCAI. His research interests include human action recognition, human motion prediction, and human motion generation using RGB, depth, and skeleton data. He has been invited to be a Technical Program Committee (TPC) Member for ACPR, ACM MM, and AAAI. He also serves as a reviewer for many international journals and conferences, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON ROBOTICS, IEEE TRANSACTIONS ON IMAGE PROCESSING, ICCV, CVPR, and ICLR.



Xinshun Wang received the master's degree from Sun Yat-sen University in 2024. He is currently an Intern with Peking University Shenzhen Graduate School, advised by Asst. Prof. Mengyuan Liu. His works have been published in IEEE TRANSACTIONS ON IMAGE PROCESSING, CVPR, and AAAI. His research interests include human-centric in-context learning and human motion prediction.



Shen Zhao received the Ph.D. degree from the Department of Electrical Engineering, Tsinghua University, China, in 2015, under the supervision of Prof. Xinxin Wang. He was a Research Fellow with Shenzhen Institute of Advanced Technology, China, supervised by Prof. Yaoqin Xie and Heye Zhang. He is currently an Associate Professor with Sun Yat-sen University. He has published articles in *Medical Image Analysis*, IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, *Medical Image Computing*, and *Computer-Assisted Intervention*. His research interests include medical image analysis and artificial intelligence. He serves as a reviewer for these journals and conferences.



Can Wang received the Ph.D. degree from the School of Electronics Engineering and Computer Science, Peking University (PKU), China, in 2016, under the supervision of Prof. Hong Liu. He studied under Prof. Reinhard Koch, the Founder of the SFM method and the Winner of the Marr Prize in 1998, the highest award in machine vision. He was a former Associate Researcher with the Institute of Automation, Chinese Academy of Sciences, and a Visiting Scholar with California State University. He is currently a Post-Doctoral Fellow with the Multimedia Information Processing Laboratory, Kiel University, Germany. He is also the Vice President and a Technical Partner of LINX ROBOT. His research interests include 3D vision and computational imaging.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:00:04 UTC from IEEE Xplore. Restrictions apply.