Hierarchically Self-Supervised Transformer for Human Skeleton Representation Learning

Yuxiao Chen[1] *, Long Zhao[2], Jianbo Yuan[3], Yu Tian[3], Zhaoyang Xia[1], Shijie Geng[1], Ligong Han[1], and Dimitris N. Metaxas[1]

[1] Rutgers University, [2] Google Research, [3] ByteDance Inc.

Abstract. Despite the success of fully-supervised human skeleton sequence modeling, utilizing self-supervised pre-training for skeleton sequence representation learning has been an active field because acquiring task-specific skeleton annotations at large scales is difficult. Recent studies focus on learning video-level temporal and discriminative information using contrastive learning, but overlook the hierarchical spatial-temporal nature of human skeletons. Different from such superficial supervision at the video level, we propose a self-supervised hierarchical pre-training scheme incorporated into a hierarchical Transformer-based skeleton sequence encoder (Hi-TRS), to explicitly capture spatial, short-term, and long-term temporal dependencies at frame, clip, and video levels, respectively. To evaluate the proposed self-supervised pre-training scheme with Hi-TRS, we conduct extensive experiments covering three skeleton-based downstream tasks including action recognition, action detection, and motion prediction. Under both supervised and semi-supervised evaluation protocols, our method achieves the state-of-the-art performance. Additionally, we demonstrate that the prior knowledge learned by our model in the pre-training stage has strong transfer capability for different downstream tasks. The source code can be found at https://github.com/yuxiaochen1103/Hi-TRS.

Keywords: Skeleton Representation Learning, Self-supervised Learning, Action Recognition, Action Detection, Motion Prediction

1 Introduction

Human skeleton data [37,26,25] are sequences of human body joints with 2D or 3D coordinates that are extracted from human activity videos. Compared with data from other modalities such as RGB frames [44,9] and depth images [47,48], human skeletons are light-weight and more robust against variations in illumination, texture, and background [40,15]. Therefore, leveraging skeletons as the input in deep neural networks to understand human activities has become prevalent recently [46,15,52,40,56,20].

Different from other modalities, skeletons have naturally inherent spatial-temporal hierarchies. The main challenge of skeleton-based methods is how to

* Correspondence to: Yuxiao Chen (yc984@cs.rutgers.edu).

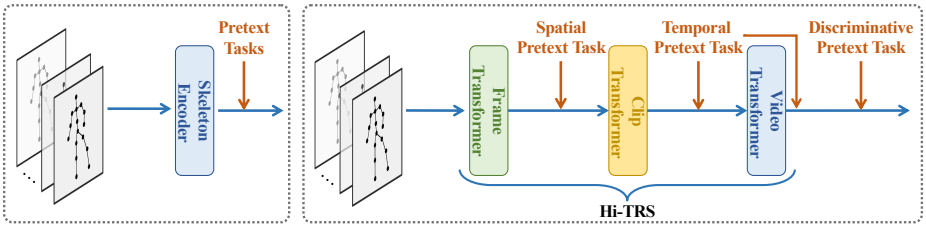

Fig. 1. Comparison of pre-training strategies. **Left:** Previous methods apply pretext tasks to supervise the final output of a skeleton encoder. **Right**: We propose to hierarchically supervise outputs of the encoder at different levels during pre-training.

properly capture the domain knowledge (*i.e.*, the correlations among the joints in the spatial and temporal domains) while extract effective feature representations from skeletons. Recent studies [50,38,40] have achieved remarkable performance improvement by learning skeleton encoders in a fully-supervised manner. These methods require massive skeleton training data with task-specific annotations which are expensive and labor-intensive to be collected. Some studies [22,43,24] tackle the problem by directly applying the self-supervised learning scheme designed for videos or images to skeleton data. Their pretext tasks extract video-level temporal and discriminative information but are only employed to supervise the final encoder outputs, as shown in Figure 1 (Left). However, these approaches do not consider the hierarchical nature of human skeletons and thus ignore the structural domain knowledge carried by them.

To address the above limitations, we propose a novel skeleton representation learning framework to capture the hierarchical spatial-temporal domain knowledge of human skeletons. As shown in Figure 1 (Right), it consists of (1) a hierarchical Transformer-based skeleton sequence encoder, namely *Hi-TRS*, incorporating with (2) a hierarchical self-supervised pre-training scheme.

Specifically, the proposed Hi-TRS models skeleton sequence in three levels. Given a skeleton sequence, the Frame Transformer *(F-TRS)* and the Clip Transformer *(C-TRS)* learn the spatial structures (**frame level**) and short-term fine-grained temporal dynamic dependencies (**clip level**) among the skeleton joints by applying self-attentions [45] on the spatial and temporal domains, respectively. Then, the clip-level embeddings are fed to the Video Transformer *(V-TRS)* to summarize long-term abstract information from clips (**video level**) and produce the feature representation of the skeleton sequence. The clip-level embeddings can be applied to short-term skeleton-based tasks, such as action detection [25,23], while embeddings from V-TRS can be used in long-term skeleton-based tasks, such as action recognition [50] and motion prediction [28].

Instead of only supervising the final output of the encoder as in previous work [24,22,43], our framework leverages different pretext tasks to supervise the encoder at different levels. As a result, the encoder acquires different types and levels of prior knowledge on human skeletons. To be specific, the *spatial pretext task* infers the information of one joint conditioned on the other joints

from the same time step. It is applied to the output of the F-TRS for learning the spatial dependencies among joints. The *temporal pretext task* assists our model to capture the temporal dynamic prior by distinguishing between valid and invalid motion patterns. It supervises the outputs of C-TRS and V-TRS. The *discriminative pretext task* captures discriminative information for supervising the output of V-TRS, which enforces the model to predict future information in a contrastive manner.

We conduct extensive experiments covering a wide range of tasks and problem settings to evaluate the proposed method. Our approach outperforms the state-of-the-art skeleton representation learning methods on three downstream tasks, including *action recognition*, *action detection*, and *motion prediction*, under both *semi-supervised* and *supervised learning* evaluation protocols. Most noticeably, Hi-TRS improves previous state-of-the-art methods on action recognition by 5.8% (semi-supervised), by 8.1% (supervised) on action detection, and by 4.2% (4.6mm) (semi-supervised) on motion prediction. Additionally, we conclude the following key observations: (1) With the help of our hierarchical supervision, the prior knowledge learned during pre-training is more versatile to support downstream tasks at different levels than prior work using contrastive learning only on the video level (see Sections 4.5 and 4.6); (2) Our approach demonstrates strong transfer capability under the transfer learning setting, where we achieve significant improvement on action recognition, action detection, and motion prediction tasks by 5%, 4.5%, and 11.7% (12.3mm), respectively; (3) Our ablation study shows that pre-training at lower levels is beneficial to higher level downstream tasks. Interestingly, we observe similar improvement obtained on lower level downstream tasks when leveraging higher level pre-training.

2 Related Work

Self-supervised Learning. Self-supervised learning targets learning effective feature representations from unlabeled data. It trains the model to solve predesigned pretext tasks, where labels are automatically generated from data without human efforts. Great efforts have been made in previous work to design pretext tasks [53,33,7]. In computer vision, colorizing grayscale images [53], image inpainting [33], and image jigsaw puzzles [31] are proposed to learn image feature representations. Motion prediction [12], temporal jigsaw puzzle recognition [31], clip orders prediction [49], and sequential verification [29] tasks are employed to learn temporal dynamic information in videos. Recently, contrastive-based pretext tasks [4,13] are introduced to learn instance discriminative information. On the other hand, language-based pre-training objectives are widely used in language domains [7,2,34]. Motivated by the success of these methods, our work leverages in-domain pretext tasks to supervise the encoder at different levels.

Skeleton Representation Learning. Early skeleton representation learning methods [11,19,42,55,5] are mainly based on the encoder-decoder architecture. Zheng *et al.* [55] trained a GAN-based model to reconstruct the original skeleton information from the corrupted input. Su *et al.* [42] trained the model to decode

the future motion of the input skeleton sequences. Recent studies adopt the self-supervised learning schemes designed for videos or images to skeleton data. Lin *et al.* [24] trained the model to jointly solve motion prediction, temporal jigsaw puzzle, and contrastive learning discriminative tasks. Li *et al.* [22] presented a memory augmented contrastive learning framework and further improved its performance by pursuing cross-view consistency constraints. Su *et al.* [43] guided the model to learn motion consistency and continuity from videos. A shortcoming of these methods is that they do not explicitly encourage the model to learn the spatial structure of skeletons. Yang *et al.* [51] proposed to represent skeleton sequences as skeleton clouds and learn the spatial and temporal information of skeletons by solving the skeleton cloud colorization problem. However, it required training two different models to learn the spatial and temporal information, respectively. Different from these methods, we use multiple pretext tasks hierarchically to train our model so that the spatial structure, temporal dynamics, and discriminative information can be learned simultaneously.

Downstream Tasks. *Action recognition* aims to predict the action category of a skeleton sequence. Studies in this area mainly focus on designing skeleton-specific architectures for feature encoding. Early methods [52,40,56,20,16,17] applied CNNs or RNNs to extract the representation of skeleton data. Recent methods [50,38] modeled the skeleton data as spatial-temporal graphs and extracted skeleton embeddings from graphs by Graph Convolutional Networks [18]. More recent studies [36,6] leveraged the self-attention mechanism to extract global dependencies among joints. In this work, we use this task to evaluate the effectiveness of skeleton representation learning methods for long-term discriminative tasks. *Action detection* temporally localizes and recognizes the presence of the action in untrimmed videos [25,41,23]. Studies in this area can be categorized into two streams. The first stream [25,23] formulates the task as a frame prediction problem, and generates detection results directly from the predicted categories of each frame in a skeleton sequence. The second stream [41,21] first generates action proposals, and then recognizes action categories from them. This paper follows the first stream to evaluate skeleton representation learning methods for short-term discriminative tasks. *Motion prediction* targets predicting future human poses based on a short observation of human motion [54,3,1,27]. Previous methods employed RNNs to encode observed information and predict future motions [10,28]. These models are trained to generate deterministic results. Recent work incorporated VAEs or GANs to decode multiple possible motions [1,27,3,35]. To evaluate the effectiveness of learned prior knowledge, we fine-tune models to predict deterministic motion in generation tasks.

3 Our Method

3.1 Hierarchical Transformer-based Encoder

The Hi-TRS model consists of three components: F-TRS, C-TRS, and V-TRS. Given a skeleton sequence, the F-TRS first learns the spatial dependencies among

the joints by applying the self-attention operation on the spatial domain. Then, the obtained results are fed to the C-TRS model to further encode the temporal fine-grained dynamics dependencies among joints and extract a feature representation at the clip level. Finally, the V-TRS infers the temporal relations among the clips and extracts the embedding of the input skeleton sequence. In the following sections, we provide details on each component.

Frame Transformer (F-TRS). Given a skeleton sequence, the positional feature of each joint is first extracted from its coordinates by a fully-connected layer with the GELU activation [14]. F-TRS utilizes the positional features from all the joints within a frame of the skeleton sequence as input. It is composed of a stack of F-TRS layers, each of which encodes the spatial dependencies among the joints based on the self-attention mechanism.

To be specific, in the l-th F-TRS layer, the model starts by projecting the input feature of each joint to query, key, and value vectors [45] by three learnable project matrices $\mathbf{W}_Q^l$, $\mathbf{W}_K^l$, and $\mathbf{W}_V^l$, respectively, as described by the following equation:

$$\mathbf{Q}_t^l = \mathbf{W}_Q^l \mathbf{X}_t^{l-1},\ \mathbf{K}_t^l = \mathbf{W}_K^l \mathbf{X}_t^{l-1},\ \mathbf{V}_t^l = \mathbf{W}_V \mathbf{X}_t^{l-1}, \tag{1}$$

where $\mathbf{X}_t^{l-1}$ is the matrix of the input features. When $l = 1$, it consists of the positional features of the joints at the t-th frame; otherwise, it is the output of the previous F-TRS layer. $\mathbf{Q}_t^l$, $\mathbf{K}_t^l$ and $\mathbf{V}_t^l$ are the transformed outputs of query, key, and value vectors, respectively. We note that the i-th rows of these four matrices are correspondent to the i-th joint in the skeleton.

The dot-product between each pair of query and key vectors is then calculated and scaled by the dimension number of the key or value vectors. Finally, the attention weights are obtained by normalizing the scaled dot-product with a Softmax function. This process is defined in the following equation:

$$\mathbf{A}_t^l = \text{Softmax}(\frac{\mathbf{Q}_t^l (\mathbf{K}_t^l)^T}{\sqrt{d_k}}), \tag{2}$$

where $(\mathbf{K}_t^l)^T$ is the transpose of $\mathbf{K}_t^l$; d_k is the dimension number of key or value vectors; $\mathbf{A}_t^l$ is the matrix of spatial attention weights among the joints at the t-th frame, and its element at the i-th row and j-th column is the attention weight of the i-th joint with respect to the j-th joint. These attention weights can be regarded as the measure of the spatial dependencies among the joints. The output feature of each joint is updated as the weighted sum of the value vectors, as shown in the following equation:

$$\mathbf{X}_t^l = \mathbf{A}_t^l \mathbf{V}_t^l. \tag{3}$$

As a result, the spatial dependence among joints is encoded into their features.

Following the multi-head attention mechanism [45], the above self-attention operation is performed h times with h different learnable projection matrices $\mathbf{W}_Q^l$, $\mathbf{W}_K^l$, $\mathbf{W}_V^l$, and the obtained h outputs for each joint are concatenated. The results are then fed to the Feedforward Network (FFN) [45], generating the final output of the l-th F-TRS laye.

Clip Transformer (C-TRS). Since a skeleton sequence typically contains a large number of joints and the self-attention operation scales quadratically with respect to the number of joints, learning the fine-grained temporal dependencies over the entire skeleton sequence using self-attention is computationally expensive. To alleviate this problem, we temporally split a skeleton sequence into a sequence of clips C with a sliding window. Then, the temporal dependencies among the joints within each clip of C are learned using the C-TRS model.

Specifically, the input of C-TRS contains the spatial features of the joints within a clip and a [CLS] token [7]. The [CLS] token summarizes useful information from all the joints of the clip, because its output embedding is the weighted sum of all joints' features, where the weights are calculated using self-attention [7,8]. The output of the [CLS] token from the C-TRS model is used as the feature representation of the entire clip. The C-TRS model is composed of a stack of C-TRS layers, each of which learns the temporal dependencies among joints by applying the multi-head self-attention mechanism on the temporal domain. We leverage the following equation to compute the attention weights $\mathbf{A}_c$:

$$\mathbf{A}_c = \text{Softmax}(\text{Mask}(\frac{\mathbf{Q}_c(\mathbf{K}_c)^T}{\sqrt{d_k}})), \tag{4}$$

where $\mathbf{Q}_c$ and $\mathbf{K}_c$ are the matrices of the query and key vectors for all the joints within the c-th skeleton clip of C, which are generated following the same way as Equation 1. More importantly, the Mask function is used for discarding spatial attention weights. It achieves this by setting the scaled dot-product among the joints from the same frame as the negative infinity, and keeps the other joints unchanged. After Softmax, all spatial attention weights in $\mathbf{A}_c$ are equal to 0.

The joint features are updated following the same method as in Equation 3 to further encode the temporal dependencies information. The output clip-level embeddings of all the clips in C are fed to the V-TRS model to extract the feature representation of the skeleton sequence.

Video Transformer (V-TRS). The V-TRS model summarizes the long-term abstracted video level information. It consists of stacked standard transformer encoder layers [45] and takes clip-level embeddings of all the clips in C together with a [CLS] token as inputs. Each of the V-TRS layers learns the temporal dependencies among the clips. The output embedding of the [CLS] token is used as the feature representation of the skeleton sequence.

3.2 Hierarchical Self-supervised Pre-training

In this section, we introduce the proposed pre-training tasks and describe how they can be applied to supervise the training of the proposed model.

Spatial Pretext Task. The spatial task is to predict the coordinates of a joint based on other joints from the same time step, as shown in Figure 2(a). Given a skeleton sequence, we first randomly sample 15% of the joints and replace the coordinate of the i-th sampled joints by: (1) the randomly generated coordinate 80% of the time, (2) the coordinate randomly sampled from other

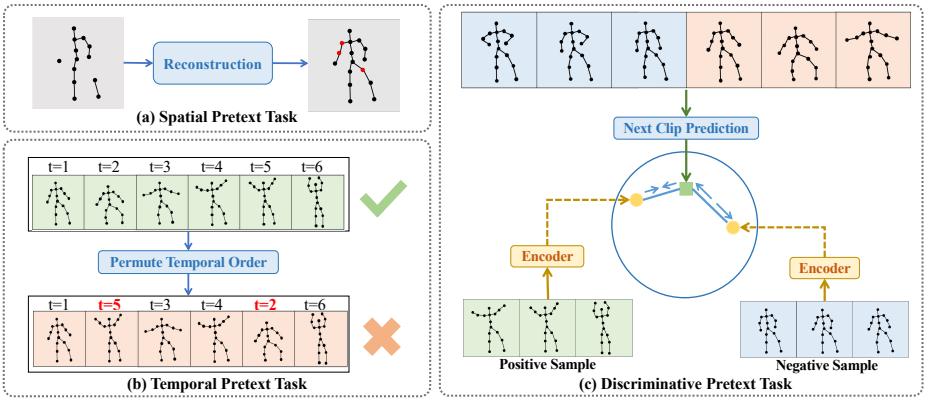

Fig. 2. Overview of our pre-training tasks which include: (**a**) Spatial pretext task: predicting the 3D coordinates of joints based on those of other joints from the same time step; (**b**) Temporal pretext task: predicting whether the temporal dynamic pattern of a skeleton clip or sequence is valid; (**c**) Discriminative pretext task: forecasting the embedding of the next clip of a skeleton sequence.

joints 10% of the time, and (3) the unchanged coordinate 10% of the time. These strategies are inspired by the masking strategies in BERT [7]. The modified skeleton sequence is fed to the F-TRS model, and then the extracted spatial embeddings of the modified joints are fed to a fully-connected layer to regress their original coordinates. The model is trained to minimize the absolute error between the predicted and ground truth coordinates by the following L1 loss $\mathcal{L}_S$:

$$\mathcal{L}_S = \frac{1}{|M|} \sum_{i \in M} ||\bar{y}_i - y_i||_1, \tag{5}$$

where M is the set of modified joints; $|M|$ is the size of M; $\bar{y}_i$ and y_i are the predicted and ground truth coordinates of the i-th modified joint, respectively.

Temporal Pretext Task. The temporal task requires the model to determine whether the temporal dynamic pattern of a skeleton clip or sequence is valid, as shown in Figure 2(b). This is a binary classification problem, where the positive samples are the original skeleton sequences or the skeleton clips cropped from the original skeleton sequences, while negative samples are generated by permuting the temporal order of the positive samples. It guides the model to learn the prior knowledge of temporal dynamics. When this task is applied to the output of C-TRS, a positive skeleton clip is generated by temporally cropping a few frames from the skeleton sequence, while the negative sample is created by swapping two randomly sampled frames of the positive clips. The output embeddings from the C-TRS model of the two clips are fed to a fully connected layer for prediction. We train the model by using the cross-entropy loss $\mathcal{L}_T^C$:

$$\mathcal{L}_T^C = -(log(p^+) + log(1 - p^-)), \tag{6}$$

where p^+ and p^- are the predicted positive possibilities for the positive and negative samples, respectively.

When this task is applied to the output of V-TRS, a negative sample is generated by temporally swapping two randomly sampled clips in the skeleton clip sequence C. We then use a linear layer to classify whether a sample is negative or positive. The model is trained by the loss function $\mathcal{L}_T^V$, which follows the definition in Equation 6.

Discriminative Pretext Task. This task predicts the embedding of the future clip of a skeleton sequence, as shown in Figure 2(c). It encourages the model to learn discriminative information by supervising the task in a contrastive way. Specifically, the model is trained to predict the embedding of the last clip in C based on the information from all other clips in C. The output from the C-TRS model of the last clip is used as the ground truth, and all other clips are fed to the V-TRS to extract a video-level embedding, which encodes the past information of the last clip. The obtained video-level embedding is fed to a fully-connected layer to regress the feature of the last clip. The model is trained by using the InfoNCE loss [32] $\mathcal{L}_D$:

$$\mathcal{L}_D = \frac{exp(\bar{e_i} \cdot e_i/\tau)}{\sum_{j=1}^{B} exp(\bar{e_i} \cdot e_j/\tau)}, \tag{7}$$

where $\bar{e_i}$ and e_i are the predicted and ground truth embedding of the last clip of the i-th video, respectively; τ is a temperature hyper-parameter [33]; B is batch size. $\mathcal{L}_D$ enforces the predicted embedding of a sample to be more similar to its ground truth than to those of other negative samples. Compared with previous studies where the contrastive learning methods are based on data augmentation [24,22], our method potentially requires lower computation as it does not require encoding augmented views of input data.

Full Pre-training Objective. The full objective of the proposed hierarchical self-supervised pre-training framework $\mathcal{L}_H$ is: $\mathcal{L}_H = \mathcal{L}_S + \mathcal{L}_T^C + \mathcal{L}_T^V + \mathcal{L}_D$.

4 Experiments

To evaluate the proposed method, we begin by introducing the datasets, evaluation protocols, and implementation details in Sections 4.1, 4.2, and 4.3, respectively. We then compare our method with the state-of-the-art skeleton representation learning approaches for the action recognition, action detection, and motion prediction tasks in Sections 4.4, 4.5, and 4.6, respectively. We further evaluate the transfer capability of the learned prior knowledge on human skeletons through pre-training in Section 4.7. Finally, we conduct an ablation study to evaluate the proposed pre-training strategy in Section 4.8.

4.1 Datasets

NTU RGB+D 60 Dataset (NTU-60). The NTU-60 dataset [37] contains 56,880 videos of 60 action categories. These videos are performed by 40 actors

and captured by three Microsoft Kinect v2 cameras from different views. Each video contains at most two subjects. A subject has 25 joints per frame. The 3D joint locations of these joints are extracted by the Microsoft Kinect cameras. Two common evaluation benchmarks [37] are recommended on this dataset. In Cross-Subject (xsub) benchmark, the training videos are from 20 selected subjects, and the testing videos are from the other 20 subjects. In Cross-View (xview) benchmark, the videos from the second and third cameras are used for training, while the videos from the first camera are used for evaluation purpose.

NTU RGB+D 120 Dataset (NTU-120). The NTU-120 dataset [26] is an extended version of the NTU-60 dataset. It contains 113,945 skeleton sequences from 120 action categories. There are two common protocols [26] for this dataset. In Cross-Subject (xsub) benchmark, the samples of the selected 53 subjects are used for training, and the samples of the remaining subjects are used for testing. In Cross-Setup (xset) benchmark, the samples with even setup IDs are used for training, and those with odd setup IDs are used for testing.

PKU Multi-Modality Dataset (PKUMMD). PKUMMD [25] is a new large-scale benchmark for continuous multi-modality 3D human action understanding. It contains almost 20,000 action instances and 5.4 million frames from 52 action categories. Actions are labeled at frame level [25]. The 3D joints are also extracted via the Microsoft Kinect v2 cameras. PKUMMD consists of two subsets: Part I and Part II. Following the common settings [25,24], the training and testing data are split under the Cross-Subject [25] protocol for each subset.

4.2 Evaluation Protocol

Following previous work [22,51], our model is evaluated under two settings: (1) the supervised setting and (2) the semi-supervised setting. Under the supervised setting, the pre-trained encoder is jointly fine-tuned with a linear classifier or a LSTM-based motion decoder [28] for downstream tasks using all the labeled pre-training data. Under the semi-supervised setting, we use the same setup as the supervised setting described above except that the amount of annotated training samples used for fine-tuning is limited.

4.3 Implementation Details

In the F-TRS, C-TRS, and V-TRS models, the number of their layers, attention heads, and dimensions of query vectors are all set as 2, 8, and 64, respectively. The input and output dimensions of the F-TRS, C-TRS, and V-TRS model are 128, 256, and 512, respectively. Before being fed to F-TRS, the input 3D coordinates of each joint are projected to 128 dimensions by a fully connected layer with the GELU activation [14]. The output of F-TRS and C-TRS are fed into a fully-connected layer to increase feature dimension to 256 and 512, respectively, before being fed into C-TRS and V-TRS. Positional encodings [45] are added to the joint features or clip features to retain their spatial identity and temporal information. Specifically, standard learnable 1D positional embeddings

Table 1. Top-1 classification accuracy (%) for action recognition on the NTU-60 and NTU-120 datasets under the supervised setting. "-2S" and "-3S" mean two-stream and three-stream based models, respectively. The best results are highlighted in bold.

Method	NTU-60		NTU-120	
	xsub	xview	xsub	xview
MS^{2}L [24] (ACMMM'20)	78.8	81.8	-	-
VPD [30] (ECCV'20)	-	81.4	-	-
MCC [43] (ICCV'21)	83.0	89.7	77.0	77.8
MCC-2S [43] (ICCV'21)	89.7	**96.3**	81.3	83.3
CrosSCLR-3S [22] (CVPR'21)	86.2	92.5	80.5	80.4
SCC-3S [51] (ICCV'21)	88.0	94.9	-	-
Hi-TRS (Ours)	86.0	93.0	80.6	81.6
Hi-TRS-2S (Ours)	89.2	95.1	84.7	86.6
Hi-TRS-3S (Ours)	**90.0**	95.7	**85.3**	**87.4**

[8] are added to the input of F-TRS and V-TRS, while learnable 2D positional embeddings [8] are used for the input of C-TRS. More details can be found in the supplementary materials.

4.4 Results on Action Recognition

In this section, we evaluate our method on the action recognition task. Given a skeleton sequence, the entire Hi-TRS model is used as the encoder, and the outputs from the V-TRS model are fed into a linear classifier (*i.e.*, a fully-connected layer) to predict action categories. For a fair comparison with the two-streams (2S) and three-streams (3S) based methods [51,22,43], we implement a 2S and a 3S version of our method. Specifically, we train three individual models from three different views of skeleton sequences, including joints, motions, and bones following [22]. During the evaluation, the 3S prediction results are obtained by fusing the prediction scores of the three models [22], while the 2S prediction results are obtained by fusing the results of the joint and bone models [22,43].

Supervised Setting. We compare the proposed Hi-TRS with other approaches on NTU-60 and NTU-120 under the supervised setting. The top-1 classification accuracy is reported on each benchmark. The obtained results are shown in Table 1. We can see that our 3S method achieves the state-of-art performance on NTU-60 and NTU-120. Note that the encoders used by several previous methods achieve better performance than our model when the parameters are randomly initialized. For example, when trained from scratch , MCC outperforms the proposed Hi-TRS by 1.9% under the cross-subject setting on the NTU-60 dataset, and the 3S-encoders used by CrosSCLR outperforms Hi-TRS by around 3% on the NTU-120 dataset. However, our method is able to outperform them when the models are pre-trained. These results demonstrate that the proposed hierar-

Table 2. Top-1 classification accuracy (%) for action recognition on the NTU-60 dataset under the semi-supervised setting. "-2S" and "-3S" mean two-stream and three-stream based models. The best results are highlighted in bold.

Method	1% data		5% data		10% data	
	xsub	xview	xsub	xview	xsub	xview
ASSL [39] (ECCV'20)	-	-	57.3	63.6	64.3	69.8
MS^{2}L [24] (ACMMM'20)	33.1	-	-	-	65.2	-
MCC-2S [43] (ICCV'21)	-	-	47.4	53.3	60.8	65.8
CrosSCLR-3S [22] (CVPR'21)	**51.1**	50.0	-	-	74.4	77.8
SCC-3S [51] (ICCV'21)	48.3	**52.5**	65.7	70.3	71.7	78.9
Hi-TRS (Ours)	39.1	42.9	63.3	68.3	70.7	74.8
Hi-TRS-3S (Ours)	49.3	51.5	**71.5**	**74.8**	**77.7**	**81.1**

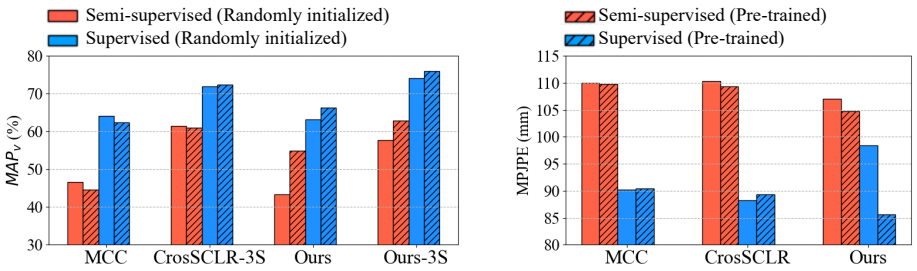

Fig. 3. Left: mAP_v (%) results on the action detection task (the higher the better). **Right**: MPJPE (mm) results on the motion prediction task (the lower the better). We note that the reported results of both MCC [43] and CrosSCLR [22] are based on our implementation. Please refer to the supplementary material for the implementation details and exact numbers of each model.

chical pre-training scheme enables Hi-TRS to learn powerful prior knowledge on human skeletons which can be successfully leveraged in the downstream task.

Semi-supervised Setting. Following the standard setup in [24,22,51], we fine-tune our pre-trained encoder and the randomly initialized linear classifier with randomly sampled 1%, 5%, and 10% of the training data on the NTU-60 dataset, respectively. From the results reported in Table 2, we observe that the proposed Hi-TRS outperforms the state of the art by a large margin under the 5% and 10% settings. On the other hand, we note that our model performs slightly worse under 1% setting. We hypothesize this is due to the fact that 1% of the training data is insufficient to train Transformer-based encoders with a large number of parameters as explained in [45].

4.5 Results on Action Detection

In this section, we compare our method with previous approaches on the action detection task. This experiment aims to evaluate the effectiveness of the learned skeleton representations for short-term discriminative tasks.

We formulate the action detection task as a per-frame classification problem following the setting in [25,23]. Given one certain frame, we extract a short clip that contains its surrounding information from the entire skeleton sequence. (Due to space limitation, please refer to the supplementary material for more details on how video clips are extracted). The obtained video clip is then fed into F-TRS and C-TRS to extract its feature representation. Finally, a linear classifier is applied to predict the action category of the input frame based on the obtained feature representation.

Following the evaluation setting of [22,23,38], the experiments are conducted on PKUMMD Part I subset. According to [25], we adopt mAP_v (mean average precision of different videos) and mAP_a (mean average precision of different actions) with the overlapping ratio of 0.5 as the evaluation metrics. The experimental results of the mAP_v metric are presented in Figure 3 (Left). From this figure, we can find that our method outperforms previous approaches under both supervised and semi-supervised settings. More importantly, we find that MCC underperforms its randomly initialized encoder by 2.1% and 1.8% in the supervised and semi-supervised settings, respectively. Meanwhile, CrosSCLR-3S also underperforms its randomly initialized encoder by 0.5% in the semi-supervised setting. One possible reason is that these two methods focus on learning long-term temporal representations [43,22]. As a result, their learned prior knowledge is not effective for short-term downstream tasks. In contrast, the proposed Hi-TRS surpasses its randomly initialized counterpart by a large margin. This demonstrates that our method can capture powerful prior knowledge for short-term downstream tasks, thanks to the proposed hierarchical pre-training strategy. We also have the same observations when the mAP_a metric is utilized, and please refer to the supplementary material for the corresponding results and qualitative analysis.

4.6 Results on Motion Prediction

In this task, the model is trained to predict the motions in the future 400 milliseconds based on an observation of two seconds, following the short-term motion prediction protocol defined in [28]. We adopt this task to evaluate the effectiveness of learned prior knowledge for generation tasks.

Specifically, the observed skeletons are fed into the proposed Hi-TRS to extract feature representations. The outputs of the V-TRS model are fed into a GRU-based decoder [28] to predict the joint coordinates of skeletons for the future 400 milliseconds. The model is then trained to minimize the Euclidean distance between the predicted poses and ground truths.

Following previous work [3,54], we employ MPJPE (mm) as the evaluation metric, which measures the distance between the ground truths and the generated results. The experiments are conducted on the NTU-60 cross-subject

Table 3. Results of motion prediction, action recognition, and action detection under the transfer learning setting. The best results are highlighted in bold.

Pre-training Dataset	PKU Part II		PKU Part I	
	MPJPE ↓	Accuracy ↑	mAP_a ↑	mAP_v ↑
Randomly Initialized Encoder	105.4	50.9	53.4	63.2
NTU-60-xsub	94.2	55.0	55.2	66.6
NTU-120-xsub	**93.1**	**55.9**	**57.9**	**67.3**

benchmark as shown in Figure 3 (Right). From this figure, we can find that our method outperforms previous methods by a large margin under both supervised and semi-supervised settings. Additionally, the learned prior knowledge of the previous methods is not useful under the semi-supervised setting. On the other hand, our method significantly outperforms the randomly initialized counterpart under different settings. It is consistent with the observations on the action detection task, demonstrating that our learned prior knowledge is more versatile to support different downstream tasks than the previous approaches. We also provide qualitative results in the supplementary material.

4.7 Evaluation of Transfer Learning

In this section, we evaluate whether the learned knowledge of Hi-TRS through the pre-training process is transferable across datasets. To this end, we first pre-train two encoders under the cross-subject protocol on NTU-60 and NTU-120, respectively. The pre-trained encoders are then fine-tuned on PKUMMD Part I and PKUMMD Part II for action detection, action recognition, and motion prediction. The obtained results are then compared with the ones of a randomly initialized encoder. These results are reported in Table 3. We can observe that pre-training can improve performance for different-level downstream tasks by a large margin, because the learned prior knowledge is transferable and versatile. Additionally, from the results of "NTU-120-xsub" and "NTU-60-xsub", we find that pre-training on larger datasets can further improve transfer capability.

4.8 Ablation Study

In this section, we evaluate the effectiveness of the proposed hierarchical pre-training strategy. This is achieved by comparing the performance of the encoders that are pre-trained on different levels.

We first show how pre-training on low levels affects the performance of the high-level downstream tasks. The experiments are conducted on the NTU-60 cross-subject benchmark for action recognition and motion prediction under the supervised protocol. The obtained results are reported in Table 4. We find that pre-training on each level (frame level, clip level, and video level) can achieve performance improvement over the randomly initialized encoder, thanks to the

Table 4. Results of the ablation study under the supervised setting on the NTU-60 cross-subject benchmark for action recognition and motion prediction. "-" means the encoder's parameters are randomly initialized. F, C, and V mean that the pre-training tasks are applied on the output of F-TRS, C-TRS, and V-TRS, respectively. The best results are highlighted in bold.

Pre-trained Level	-	F	C	V	F+C	F+V	C+V	F+C+V
Accuracy(%) ↑	79.6	80.8	81.1	82.0	83.9	84.1	84.0	**86.0**
MPJPE(mm) ↓	98.4	97.3	96.7	88.1	95.4	87.4	90.2	**85.6**

Table 5. Results of the ablation study under the supervised setting on the PKUMMD Part I subset for action detection. "-" means that the encoder's parameters are randomly initialized. F, C, and V mean that the pre-training tasks are applied on the output of F-TRS, C-TRS, and V-TRS, respectively. The best results are in boldface.

Pre-trained Level	-	F+C	F+C+V
mAP_a	53.4	55.6	**58.4**
mAP_v	63.2	65.1	**66.3**

powerful prior knowledge learned from the pre-training tasks of each level. Additionally, pre-training on any combination of two levels achieves higher performance improvement than pre-training on only one level. More importantly, the best improvement is achieved when the encoder is pre-trained on all levels. This confirms the fact that our full model manages to combine prior knowledge containing spatial structure, temporal dynamics, and discriminative information for human skeletons during the pre-training stage.

To further explore how the high-level pre-training tasks affect the low-level downstream tasks, we conduct experiments on the PKUMMD Part I subset for action detection. The results are shown in Table 5. Please refer to the supplementary material for the results of more model variants. We can see that pre-training on high level (video level) leads to performance improvement on the low level downstream task as well, since it can introduce temporal dynamic information and complementary discriminative information.

5 Conclusion

In this work, we proposed a novel method that encodes skeleton sequences using a hierarchical Transformer-based encoder and designed a pre-training scheme consisting of three pretext tasks at three different levels. We conducted extensive experiments under different learning settings. For the supervised and semi-supervised settings, our method achieves the state-of-the-art performance against competitive baselines. Moreover, the learned prior knowledge through hierarchical pre-training shows strong transfer learning capability for downstream tasks at different levels. The experimental results demonstrate that our method is an effective way for learning feature representations of skeleton data.

References

1. Barsoum, E., Kender, J., Liu, Z.: Hp-gan: Probabilistic 3d human motion prediction via gan. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 1418–1427 (2018)
2. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
3. Cai, Y., Wang, Y., Zhu, Y., Cham, T.J., Cai, J., Yuan, J., Liu, J., Zheng, C., Yan, S., Ding, H., et al.: A unified 3d human motion synthesis model via conditional variational auto-encoder. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11645–11655 (2021)
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
5. Chen, Y., Zhao, L., Peng, X., Yuan, J., Metaxas, D.N.: Construct dynamic graphs for hand gesture recognition via spatial-temporal attention. In: BMVC (2019)
6. Cheng, Y.B., Chen, X., Chen, J., Wei, P., Zhang, D., Lin, L.: Hierarchical transformer: Unsupervised representation learning for skeleton-based human action recognition. In: 2021 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6. IEEE (2021)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
8. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
9. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6202–6211 (2019)
10. Fragkiadaki, K., Levine, S., Felsen, P., Malik, J.: Recurrent network models for human dynamics. In: Proceedings of the IEEE international conference on computer vision. pp. 4346–4354 (2015)
11. Gui, L.Y., Wang, Y.X., Liang, X., Moura, J.M.: Adversarial geometry-aware human motion prediction. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 786–803 (2018)
12. Han, T., Xie, W., Zisserman, A.: Video representation learning by dense predictive coding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. pp. 0–0 (2019)
13. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2020)
14. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
15. Hussein, M.E., Torki, M., Gowayyed, M.A., El-Saban, M.: Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations. In: Twenty-third international joint conference on artificial intelligence (2013)
16. Ke, Q., Bennamoun, M., An, S., Sohel, F., Boussaid, F.: A new representation of skeleton sequences for 3d action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3288–3297 (2017)

17. Kim, T.S., Reiter, A.: Interpretable 3d human action analysis with temporal convolutional networks. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW). pp. 1623–1631. IEEE (2017)
18. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
19. Kundu, J.N., Gor, M., Uppala, P.K., Radhakrishnan, V.B.: Unsupervised feature learning of human actions as trajectories in pose embedding manifold. In: 2019 IEEE winter conference on applications of computer vision (WACV). pp. 1459–1467. IEEE (2019)
20. Li, C., Zhong, Q., Xie, D., Pu, S.: Skeleton-based action recognition with convolutional neural networks. In: 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW). pp. 597–600. IEEE (2017)
21. Li, C., Zhong, Q., Xie, D., Pu, S.: Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. arXiv preprint arXiv:1804.06055 (2018)
22. Li, L., Wang, M., Ni, B., Wang, H., Yang, J., Zhang, W.: 3d human action representation learning via cross-view consistency pursuit. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4741–4750 (2021)
23. Li, Y., Lan, C., Xing, J., Zeng, W., Yuan, C., Liu, J.: Online human action detection using joint classification-regression recurrent neural networks. In: European conference on computer vision. pp. 203–220. Springer (2016)
24. Lin, L., Song, S., Yang, W., Liu, J.: Ms2l: Multi-task self-supervised learning for skeleton based action recognition. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 2490–2498 (2020)
25. Liu, C., Hu, Y., Li, Y., Song, S., Liu, J.: Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. arXiv preprint arXiv:1703.07475 (2017)
26. Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C.: Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence **42**(10), 2684–2701 (2019)
27. Mao, W., Liu, M., Salzmann, M.: Generating smooth pose sequences for diverse human motion prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13309–13318 (2021)
28. Martinez, J., Black, M.J., Romero, J.: On human motion prediction using recurrent neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2891–2900 (2017)
29. Misra, I., Zitnick, C.L., Hebert, M.: Shuffle and learn: unsupervised learning using temporal order verification. In: European Conference on Computer Vision. pp. 527–544. Springer (2016)
30. Nie, Q., Liu, Z., Liu, Y.: Unsupervised 3d human pose representation with viewpoint and pose disentanglement. In: European Conference on Computer Vision. pp. 102–118. Springer (2020)
31. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: European conference on computer vision. pp. 69–84. Springer (2016)
32. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
33. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2536–2544 (2016)

34. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
35. Petrovich, M., Black, M.J., Varol, G.: Action-conditioned 3d human motion synthesis with transformer vae. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10985–10995 (2021)
36. Plizzari, C., Cannici, M., Matteucci, M.: Spatial temporal transformer network for skeleton-based action recognition. In: International Conference on Pattern Recognition. pp. 694–701. Springer (2021)
37. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1010–1019 (2016)
38. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12026–12035 (2019)
39. Si, C., Nie, X., Wang, W., Wang, L., Tan, T., Feng, J.: Adversarial self-supervised learning for semi-supervised 3d action recognition. In: European Conference on Computer Vision. pp. 35–51. Springer (2020)
40. Song, S., Lan, C., Xing, J., Zeng, W., Liu, J.: An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In: Proceedings of the AAAI conference on artificial intelligence (2017)
41. Song, S., Lan, C., Xing, J., Zeng, W., Liu, J.: Spatio-temporal attention-based lstm networks for 3d action recognition and detection. IEEE Transactions on image processing **27**(7), 3459–3471 (2018)
42. Su, K., Liu, X., Shlizerman, E.: Predict & cluster: Unsupervised skeleton based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9631–9640 (2020)
43. Su, Y., Lin, G., Wu, Q.: Self-supervised 3d skeleton action representation learning with motion consistency and continuity. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13328–13338 (2021)
44. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 4489–4497 (2015)
45. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
46. Vemulapalli, R., Arrate, F., Chellappa, R.: Human action recognition by representing 3d skeletons as points in a lie group. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 588–595 (2014)
47. Wang, P., Li, W., Gao, Z., Zhang, J., Tang, C., Ogunbona, P.O.: Action recognition from depth maps using deep convolutional neural networks. IEEE Transactions on Human-Machine Systems **46**(4), 498–509 (2015)
48. Xiao, Y., Chen, J., Wang, Y., Cao, Z., Zhou, J.T., Bai, X.: Action recognition for depth video using multi-view dynamic images. Information Sciences **480**, 287–304 (2019)
49. Xu, D., Xiao, J., Zhao, Z., Shao, J., Xie, D., Zhuang, Y.: Self-supervised spatiotemporal learning via video clip order prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10334–10343 (2019)
50. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Thirty-second AAAI conference on artificial intelligence (2018)

51. Yang, S., Liu, J., Lu, S., Er, M.H., Kot, A.C.: Skeleton cloud colorization for unsupervised 3d action representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13423–13433 (2021)
52. Zhang, P., Lan, C., Xing, J., Zeng, W., Xue, J., Zheng, N.: View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2117–2126 (2017)
53. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: European conference on computer vision. pp. 649–666. Springer (2016)
54. Zhao, L., Peng, X., Tian, Y., Kapadia, M., Metaxas, D.N.: Semantic graph convolutional networks for 3d human pose regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3425–3435 (2019)
55. Zheng, N., Wen, J., Liu, R., Long, L., Dai, J., Gong, Z.: Unsupervised representation learning with long-term dynamics for skeleton based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence (2018)
56. Zhu, W., Lan, C., Xing, J., Zeng, W., Li, Y., Shen, L., Xie, X.: Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In: Proceedings of the AAAI conference on artificial intelligence (2016)