Unified Pose Sequence Modeling

Lin Geng Foo[1*] Tianjiao Li[1*] Hossein Rahmani[2] Qiuhong Ke[3] Jun Liu[1†]
[1]Singapore University of Technology & Design [2]Lancaster University [3]Monash University
{lingeng_foo,tianjiao_li}@mymail.sutd.edu.sg, h.rahmani@lancaster.ac.uk,
qiuhong.ke@monash.edu, jun_liu@sutd.edu.sg

Abstract

We propose a Unified Pose Sequence Modeling approach to unify heterogeneous human behavior understanding tasks based on pose data, e.g., action recognition, 3D pose estimation and 3D early action prediction. A major obstacle is that different pose-based tasks require different output data formats. Specifically, the action recognition and prediction tasks require class predictions as outputs, while 3D pose estimation requires a human pose output, which limits existing methods to leverage task-specific network architectures for each task. Hence, in this paper, we propose a novel Unified Pose Sequence (UPS) model to unify heterogeneous output formats for the aforementioned tasks by considering text-based action labels and coordinate-based human poses as language sequences. Then, by optimizing a single auto-regressive transformer, we can obtain a unified output sequence that can handle all the aforementioned tasks. Moreover, to avoid the interference brought by the heterogeneity between different tasks, a dynamic routing mechanism is also proposed to empower our UPS with the ability to learn which subsets of parameters should be shared among different tasks. To evaluate the efficacy of the proposed UPS, extensive experiments are conducted on four different tasks with four popular behavior understanding benchmarks.

1. Introduction

Pose sequences, which capture the movements of the human body via human joint coordinates, are well-known to be an efficient and effective representation of human motion and behaviour [58,80]. This is mainly because pose sequences often provide enough information to characterize complex motion patterns [31], while being robust against superficial visual variations such as the background, clothing texture and illumination conditions [43,44]. At the same time, by using depth sensors such as the Kinect, pose data can also be conveniently obtained in real-time to facilitate downstream applications. Therefore, the potential of pose sequences to tackle behaviour understanding has attracted a lot of attention in recent years.

Notably, the usage of pose sequences has been widely explored across many practical applications, including human-robot interaction [1, 59], augmented reality [4, 51] and security surveillance [18, 71]. Specifically, pose sequences, as informative inputs, can facilitate certain aspects in these applications, such as action recognition [12, 44, 48, 66, 67, 86–88], 3D pose estimation [41, 45, 84, 92, 95, 96] and early action prediction [23, 35, 39, 77, 78], making these tasks popular and important areas of research.

However, existing methods for each task still often require *task-specific* architectures, e.g., hourglass networks for pose estimation [84] and specialized GCN architectures for action recognition [11,69], while the performing of multiple pose-based tasks with a single model is not well explored. Therefore, in order to perform multiple tasks, users will often need to design and train multiple separate models, which can be inconvenient and inefficient.

Hence, in this work, we seek to simplify and unify the modeling for several popular and important pose-based tasks: 3D action recognition, 2D action recognition, 3D pose estimation and 3D early action prediction. This is a challenging goal that has not been achieved before, requiring a single model to cover a large scope involving 2D tasks, 3D tasks, as well as 2D to 3D lifting. By unifying these pose-based tasks and removing the need to design and train separate task-specific models to tackle different pose-based tasks, we can greatly reduce the difficulty and complexity involved in tackling these tasks. Moreover, a unified model is also an elegant way of handling multiple tasks that brings us one step closer in our pursuit of general purpose vision systems [25], i.e., an efficient multi-purpose AI model akin to the human brain.

To this end, we propose a Unified Pose Sequence (UPS) model to unify the architecture and output format for multiple popular pose-based tasks. Our UPS is a single uni-

*equal contribution
†corresponding author

fied model that simultaneously tackles multiple tasks without task-specific designs or branches, i.e., with a unified decoder. In order to unify the output formats of different tasks (which can be very different) to be produced by a single decoder, our UPS *predicts a sequence of output tokens*, similar to language modeling tasks. Specifically, our UPS's decoder auto-regressively produces a sequence of output tokens, such that the output sequence can potentially be of different lengths to meet the requirements of multiple tasks. Additionally, these output tokens can be interpreted as text embeddings, which are a powerful and general representation that can be mapped into various predictions as required. Moreover, to mitigate the potential destructive interference [60, 90] brought by the heterogeneity between different tasks, we propose a dynamic routing mechanism for our UPS that facilitates parameter sharing between tasks.

In summary, our contributions are as follows:

- We propose a Unified Pose Sequence (UPS) model that can tackle several popular pose-based tasks through a single unified framework. UPS simultaneously tackles multiple tasks without task-specific designs or branches by modeling the output as a sequence of tokens, enabling it to handle different output formats.
- On four popular pose-based tasks (3D action recognition, 2D action recognition, 3D pose estimation and 3D early action prediction), UPS achieves good performance that is comparable to state-of-the-art methods.

2. Related Work

Pose-based Tasks. Due to the practicality and effectiveness of pose sequences in capturing human motion and behaviour [58, 80], how to better leverage upon them to perform various tasks has become an increasingly popular area of research. There are a few important tasks in particular that have received a lot of attention. *2D and 3D Action Recognition* [12, 13, 20, 21, 36, 38, 44, 48, 66, 67, 88] is where we predict the action class of the input 2D and 3D pose sequence respectively. In *3D Pose Estimation* [24, 41, 45, 84, 95, 96], we predict the 3D coordinates of a human's joints, with the input being either RGB images [53, 70] or 2D poses [24, 41, 45, 84, 95, 96]. In this work, we use 2D pose sequences as input. Besides, in *3D Early Action Prediction* [23, 27, 35, 39, 76–78], we would like to predict the action performed by the subject, after observing only the front parts of each pose sequence. However, these existing methods often require *task-specific* architectures, and the performing of multiple pose-based tasks with a single model is not well explored. Hence, in this work, we seek a unified model to tackle these tasks simultaneously.

Sequence Modeling. Sequence modeling is an important concept in the field of NLP, particularly for the generation of a sentence as a sequence of words [57, 73]. In general, a Transformer is used in an auto-regressive manner [57, 73], taking previous tokens as input to predict the next token, and thus generating a sequence of tokens. Recently, such sequence modeling has also been explored for some vision-language tasks [8,14,25,75,97]. Different from previous works, we explore the crucial challenge of unifying several popular pose-based tasks. Not only do these skeleton-based tasks often require different task-specific designs to successfully tackle, they also require vastly different output formats (e.g., video-level classification vs joint-level coordinates) and input formats (e.g., 2D vs 3D pose).

Multi-task Learning. A related field is multi-task learning [15, 93], where models are trained to perform multiple tasks simultaneously. In general, there are several approaches to multi-task learning, including multi-task architecture designs [16, 50, 62, 94], optimization methods [22, 90], and learning of task relationships [3, 17, 26, 79]. However, in existing works on multi-task learning, each task still requires dedicated task-specific branches, thus extending new tasks in this setting will require additional sets of parameters, e.g., task-specific heads. Differently, our UPS unifies multiple skeleton-based tasks into one single model by integrating all output formats into a language-based format, and can conveniently handle various tasks without needing any modification to the model architecture.

Language Models for Vision Tasks. Language models have often been applied to facilitate vision-based tasks. For instance, language is used as input in text-to-image generation [61, 85] and visual grounding [32, 83, 89], while language is an output for the image captioning [9, 56] and visual question answering [2, 82] tasks. Here, we propose a novel paradigm integrating language and pose into a unified model to handle various pose and action tasks.

3. Method

3.1. Overview

In order to unify diverse pose-based human behavior understanding tasks (e.g., action recognition, 3D pose estimation, early action prediction) with a single model, we propose a novel Unified Pose Sequence model as shown in Fig. 1. A major obstacle we face is that different tasks require different output formats. For instance, the action recognition task requires class predictions, while the 3D pose estimation task requires 3D locations of human joints.

Therefore, to jointly represent the output sequences from our UPS and heterogeneous ground-truth formats from different tasks, we utilize sequence modeling [8] to unify different target data formats and avoid multiple task-specific output heads. Specifically, we tokenize the text-based class labels and coordinate-based joint locations following the standard language modeling setup and establish a *unified vocabulary*. Then, given the UPS output token sequences,

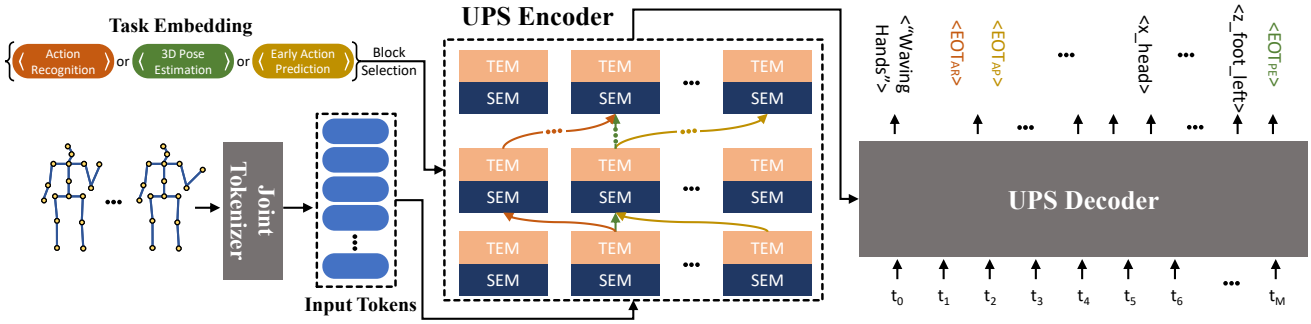

Figure 1. Illustration of the proposed UPS model. Our UPS consists of three major parts, i.e., a joint tokenizer, a UPS encoder and a UPS decoder. The joint tokenizer takes a pose sequence (in coordinates format) as input and produces an input token sequence. Then, the input token sequence is sent to the dynamic UPS encoder together with a task embedding indicating which task the UPS model is tackling. The UPS encoder can adaptively determine which sets of parameters should be shared for different tasks, conditioned on the input task embedding. Next, the encoded token sequence is sent to the UPS decoder to produce unified output sequence in an auto-regressive manner. The UPS decoder will stop generating outputs if any of the three ending tokens (i.e., EOT_{AR}, EOT_{PE}, and EOT_{AP}) is encountered.

we map the output token back to the task-required formats via a vocabulary lookup. Furthermore, to mitigate the potential destructive interference issue [60, 90] caused by simultaneously learning various heterogeneous tasks with a single model, we include a dynamic design in our UPS encoder to adaptively learn which sets of parameters should be shared by different tasks.

3.2. Unified Outputs Modeling

In this subsection, we describe in detail how we unify the tokenization of heterogeneous target data formats for different tasks, e.g., action classes in text format (action recognition and early action prediction) and 3D human joint locations in coordinates format (3D pose estimation) into a unified category-joint look-up vocabulary. This allows us to avoid designing multiple task-specific heads.

Action Token. In order to handle the heterogeneity between action classes and human joint coordinates, unlike traditional representations, here we handle action category labels purely in *text format*. Then the category labels in text format (e.g., "walking", "squatting down") are sent to an off-the-shelf language model (e.g., RoBERTa [47]) to extract text features as our *action token* $s_{cls} \in \mathbb{R}^d$, where d denotes the token size. Here we have n_{cls} discrete *action tokens*, where n_{cls} is the total number of action categories.

Pose Token Sequence. To further unify pose sequences (given in coordinates format), we leverage sequence modeling [8] to tokenize ground-truth 3D human joint locations. Specifically, we assume that we have a target pose sequence $P_{\text{TAR}} \in \mathbb{R}^{N \times J \times V}$ in 3D space, where N, J and V are sequence length, number of joints, and number of dimensions respectively. We denote the j-th joint at the n-th frame as $p_{n,j}$, which is at the location $(x_{n,j}, y_{n,j}, z_{n,j})$.

Next, we would like to represent each possible $(x_{n,j}, y_{n,j}, z_{n,j})$ coordinate as an item in the look-up vocabulary. However, this can be difficult because the ground truth coordinates are often given as real numbers, and it can be infeasible to represent this in a look-up vocabulary (with finite entries). Thus, we quantize each dimension (i.e., X-dim, Y-dim and Z-dim) into n_{bins} quantized bins. Note that after quantization, we have $3 \cdot n_{bins}$ discrete bins for X-dim, Y-dim and Z-dim in total.

Then, we can represent each bin by a text-prompted token extracted by the same language model used for *action token* (e.g., RoBERTa [47]). Here, we take the X-dim as an example: we describe the first location on the X-dim as "The first horizontal coordinate" and so forth, where the last location on the X-dim can be queried by "The $\{n_{bin}$-$th\}$ horizontal coordinate". Then we send these descriptions to the pre-trained language model and extract $3 \cdot n_{bins}$ discrete joint tokens. In such a way, an arbitrary joint $p_{n,j}$ can be denoted using 3 discrete tokens $s^X_{n,j}$, $s^Y_{n,j}$ and $s^Z_{n,j} \in \mathbb{R}^d$ along each dimension. Therefore, the target coordinate-format pose sequence $P_{\text{TAR}} \in \mathbb{R}^{N \times J \times V}$ can be further discretely tokenized into a *pose token sequence* S_{TAR}, i.e., $\{s^X_{1,1}, s^Y_{1,1}, s^Z_{1,1}, \ldots s^X_{N,J}, s^Y_{N,J}, s^Z_{N,J}\}$.

EOTs: End of Task Tokens. Another challenge we face is that the different tasks can require output sequences of different lengths. For example, for action recognition we only require one output token to represent the class prediction, while for 3D pose estimation we need to produce a longer sequence of J joint coordinates. To adaptively produce the output token sequence as required while avoiding the usage of multiple task-specific output heads, we introduce *end of task* tokens (EOTs) to indicate when to stop the decoding process for the UPS model. Similar to *joint tokens*, we leverage text prompts to produce these tokens, e.g., the description "action recognition ends here" is sent to the pretrained language model to generate the ending token for action recognition task EOT_{AR}. Here we define 3 types of

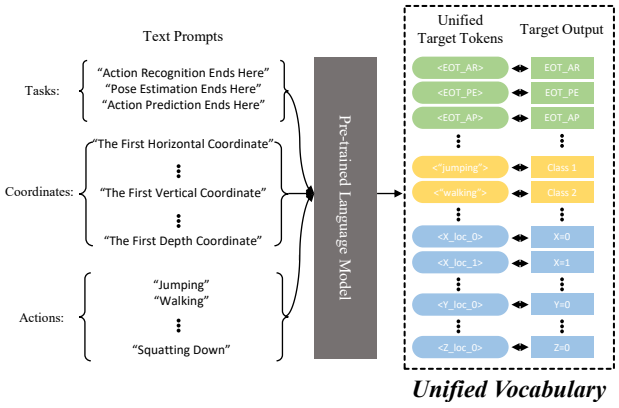

Figure 2. Illustration of the proposed unified vocabulary. To unify heterogeneous target data formats, we leverage sequence modeling and use text descriptions to represent (1) action category labels, (2) joint coordinate values and (3) task-ending indicators. These text descriptions are sent to the off-the-shelf RoBERTa [47] to extract text features as our target tokens. Then, our unified vocabulary is established naturally by matching each target output (across various data formats) with the corresponding extracted target tokens.

EOTs: EOT_{AR}, EOT_{PE}, and EOT_{AP} for action recognition, 3D pose estimation and early action prediction respectively.

Unified Vocabulary. We establish a unified vocabulary containing all tokenized action categories, quantized bins and EOTs as shown in Fig. 2. As we can see, the unified vocabulary Ω_{VOB} represents the mapping relationships between original target formats and unified tokens. The size of the vocabulary is $(n_{cls} + 3 \cdot n_{bins} + 3)$, i.e., $\Omega_{\text{VOB}} \in \mathbb{R}^{(n_{cls}+3\cdot n_{bins}+3)\times d}$.

Task Embedding. Another issue we face is: how does our model know which task it is tackling, or which sequence to output? In other words, during forward inference, when we specify that we are tackling a certain task (out of a pool of tasks), we expect our unified model to produce the output sequence that we require. To tackle this issue, a straightforward option would be to always produce a long sequence containing the output sequences for all tasks, but this is an inefficient approach that is difficult to scale. Instead, a more efficient and elegant approach is for our unified model to learn to flexibly switch around different output formats, and only produce the output sequence that corresponds to a given task.

To achieve this, we introduce *task embeddings* τ as an additional input to the model. Specifically, a task embedding is introduced for each task (i.e., $\tau_m \in \mathbb{R}^q$ is introduced for the m-th task), and is sent as input to the model when we require output sequences for the m-th task. The task embeddings $\{\tau_m\}_{m=1}^{M}$ can be optimized in an end-to-end manner. These task embeddings play an important role in our routing mechanism, which is described in Sec. 3.4.

3.3. Unified Pose Sequence (UPS) Model

In this section, we describe the architecture of our UPS in more detail. Our UPS architecture needs to be task-agnostic, and thus utilizes simple components that have been shown to be effective in pose-based tasks. As shown in Fig. 1, the proposed UPS model comprises of 3 components which are: a joint tokenizer, a dynamic UPS encoder and a UPS decoder. Here, we let the original input pose sequences in coordinate formats be $P_{\text{IN}} \in \mathbb{R}^{N\times J\times V}$, where N, J and V denotes number of frames, number of human joints, and number of dimensions.

Joint Coordinates Tokenizer. To tokenize the input pose sequence P_{IN} (which is in the coordinates format), we introduce a joint coordinates tokenizer. which is comprised of three TCN-GCN layers [11]. The joint tokenizer takes $P_{\text{IN}} \in \mathbb{R}^{N\times J\times V}$ as input, and produces input token sequence $S_{\text{IN}} \in \mathbb{R}^{N\times J\times d}$.

UPS Encoder. Our UPS encoder consists of $L_{encoder}$ stacks of SEM-TEM blocks, where each SEM-TEM block is a SEM module followed by a TEM module (as described in more detail below). The UPS encoder takes S_{IN} as inputs and produce encoded hidden token features $S_{\text{EN}} \in \mathbb{R}^{N\times J\times d}$. We remark that, as shown in Fig. 1, we design a routing mechanism and send an additional task token to the UPS encoder, to adaptively determine which subsets of parameters should be shared for different tasks. Details of the routing mechanism are outlined in Sec. 3.4. Below, we describe the SEM and TEM in detail.

(1) Spatial Encoding Module (SEM). To obtain representative topology information among different human joints, we use the GCN block [11, 44, 67] as a basic component for our SEM. Each SEM is comprised of two basic GCN blocks, where the input tokens $X_{\text{IN}} \in \mathbb{R}^{N\times J\times d}$ are fed to the GCN blocks sequentially to produce tokens $X_{\text{LAT}} \in \mathbb{R}^{N\times J\times d}$ for the following TEM.

(2) Temporal Encoding Module (TEM). The TEM performs includes a Multi-Head Self-Attention (MHSA) layer followed by a 2-layer MLP. Importantly, the MHSA is conducted between "frame-level" tokens in order to efficiently encode temporal information over a long sequence of frames. Specifically, we first reshape $X_{\text{LAT}} \in \mathbb{R}^{N\times J\times d}$ as $X_{\text{LAT}} \in \mathbb{R}^{N\times J*d}$, where we now have N "frame-level" tokens of length $J*d$. The input X_{LAT} is sent into the TEM, and we encode the temporal relationships across frames, resulting in an output $X_{\text{OUT}} \in \mathbb{R}^{N\times J*d}$. X_{OUT} is then reshaped to $X'_{\text{IN}} \in \mathbb{R}^{N\times J\times d}$ to be sent into the next module.

UPS Decoder. Our decoder produces the prediction of the next token in an auto-regressive manner, while aggregating information from the input tokens as well as the partial sequence of output tokens that have already been produced.

Here, our decoder consists of $L_{decoder}$ basic tranformer blocks [19]. At inference, our decoder takes in the tokens corresponding to the input and sequentially produce unified

output tokens until the EOT token is encountered. Details on training and testing are elaborated in Sec. 3.5.

3.4. Routing Mechanism

Our proposed UPS architecture can tackle several popular skeleton-based tasks (e.g., action recognition, 3D pose estimation and early action prediction) with a unified model and output format. However, the various tasks described can require very different types of knowledge, and simultaneously tackling these tasks altogether can be challenging. In particular, using the exact same set of parameters to tackle multiple different tasks can lead to destructive interference [60, 90], and a lowered performance. Thus, we further extend our UPS encoder with a dynamic routing mechanism. Our dynamic routing mechanism allows tasks to either share blocks of parameters or use separate sets of parameters, depending on which one is more beneficial for performance. This encourages knowledge sharing, while mitigates the destructive interference issue.

Firstly, we introduce H parallel blocks in each layer of the UPS encoder, where each of the H blocks in each layer has the same architecture. Thus, our UPS encoder will consist of a stack of L layers consisting of H blocks each. Furthermore, we introduce block embeddings $B_{l,h} \in \mathbb{R}^q$ for the h-th block in the l-th layer $\theta_{l,h}$. Then, during forward inference, a Block Selection step is conducted to select the most suitable block.

In the Block Selection step, we make use of our task embedding τ that is defined for each task, and use them to selectively activate blocks at each layer to perform our dynamic routing. We note that such a design is unlike previous works [97] that use their task embeddings as input tokens that are directly processed along with P_{IN} in SEM and TEM.

We compute the dot product between the task embedding and the block embeddings to calculate scores to determine which block to activate. Specifically, given a task embedding τ, we perform the following steps at a layer l to select the most suitable block among $\{\theta_{l,h}\}_{h=1}^{H}$, as follows:

$$s_{l,h} = B_{l,h} \cdot \tau, \;\; h \in [1, H] \tag{1}$$

$$m_l = GumbelSoftmax(\{s_{l,h}\}_{h=1}^{H}) \tag{2}$$

where the selected block for the l-th layer is θ_{l,m_l}. In other words, the block θ_{l,m_l} is chosen because its block embedding B_{l,m_l} is the closest (and hence most suitable) to the task embedding $\tau \in \mathbb{R}^q$. Importantly, this mechanism leads to the samples of a task sharing the same route, while different tasks can potentially share or not share the same route, depending on the learned task and block embeddings. Note that, as the Argmax operation is non-differentiable, we use the Gumbel-Softmax operation [30] so that the entire model can be trained in an end-to-end manner.

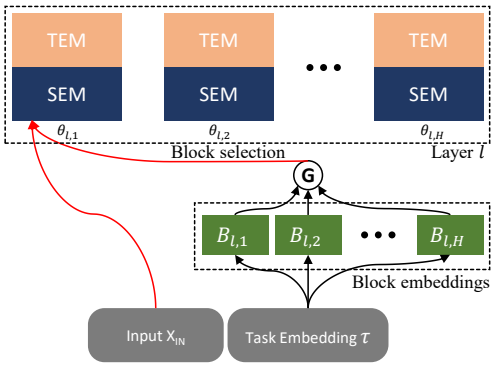

Figure 3. Task embeddings τ are learned to dynamically select the optimal blocks to use during training. At each l-th layer of the encoder, there are H blocks $\{\theta_{l,h}\}_{h=1}^{H}$ to choose from (indicated in beige and blue), with corresponding block embeddings $\{B_{l,h}\}_{h=1}^{H}$ (indicated in green). To select the most suitable block, we compute the dot products between task embedding τ and block embeddings $\{B_{l,h}\}_{h=1}^{H}$ and send them into the Gumbel-Softmax operator (indicated by Ⓖ). By optimizing the embeddings τ and $\{B_{l,h}\}_{h=1}^{H}$ during training, our dynamic routing mechanism can alleviate the issue of destructive interference and improve the sharing of knowledge.

3.5. Training and Testing.

Training Loss. We generate our ground truth sequences using the techniques described in Sec. 3.2. We also use the negative log-likelihood loss to optimize our sequence prediction capabilities, following previous works [8, 14] on language modeling. Specifically, the loss is formulated as:

$$\mathcal{L} = -\frac{1}{K}\sum_{k=1}^{K} logP_{\phi}(y_k|y_{<k}, X) \tag{3}$$

where K is the length of the ground truth sequence, ϕ refers to all trainable parameters (i.e. $\phi = \{\theta, B, \tau\}$), X is the input pose sequence, y_k refers to the k-th output token and $y_{<k}$ refers to all output tokens before the k-th output token. Intuitively, we are training our model ϕ to accurately predict the next token, with only access to the inputs and previously predicted tokens. We note that all tasks can be optimized through this loss.

UPS Training. Here, we rotate the training among the four different tasks with every iteration. Specifically, we cycle through the tasks in this manner: 3D pose estimation $\rightarrow$ 2D action recognition $\rightarrow$ 3D early action prediction $\rightarrow$ 3D action recognition. We also train a corresponding task embedding for each task.

Inference. After obtaining the trained UPS model, we test the single unified model on all the tasks. For each task, we perform model inference according to the optimized routes learned by the corresponding task token.

4. Implementation Details

We leverage RoBERTa$_{\text{Base}}$ [47] as the pre-trained language model to extract word embeddings from text prompts. For the UPS decoder vocabulary, it holds action tokens, joint coordinate tokens, and EOT tokens, and thus it has a size of $(n_{cls} + 3 \cdot n_{bins} + 3)$. To obtain representative features from the human topology, a GCN-based [11, 88] tokenizer is utilized. For UPS encoder and decoder, both of them consist of 3 stacked SEM-TEM blocks, i.e., $L_{encoder} = L_{decoder} = 3$. In each layer of the UPS encoder, by setting $H = 2$, our dynamic design has 2 parallel SEM-TEM blocks. For all block embeddings and task embeddings, we set $q = 256$, and randomly initialize them. All experiments are conducted on 8 Nvidia V100 GPUs, and the batch size is set as 1,024. We use AdamW [49] optimizer with weight decay of $5e-4$. The initial learning rate is set to $1e-2$ and gradually decays to 0.

5. Experiments

We conduct experiments on four tasks with the same unified model. The tasks are: 3D action recognition, 2D action recognition, 3D pose estimation, and 3D early action prediction. We experiment on NTU RGB+D 60 (NTU60) [64] and NTU RGB+D 120 (NTU120) [42] datasets for 3D action recognition and 3D early action prediction, Kinetics 400 [33] dataset for 2D action recognition and Human3.6M [28] dataset for 3D pose estimation.

We conduct experiments on the following variants: (1) *UPS$_{separate}$*, which is optimized separately on each task. (2) *UPS*, which represents our full model; it is trained on all tasks at the same time and then fine-tuned on each task based on our task embedding. When we train on all tasks at the same time, our dynamic routing mechanism is leveraged in each layer to encourage different tasks to either share common knowledge by selecting same set of parameters, or to mitigate the destructive interference issue by using separate sets of parameters.

5.1. 3D Action Recognition

In 3D action recognition, we are given a 3D pose sequence, and want to predict its action class.

Dataset. NTU RGB+D 60 [64] is a large dataset that has been widely used for 3D action recognition. It consists of about 56k RGB+D sequences from 60 activity classes. NTU RGB+D 120 [42] is an extension of [64], and is currently the largest dataset for 3D action analysis. It is a challenging dataset that contains more than 114k pose sequences across 120 activity classes. We follow the standard evaluation protocol of previous works [42, 64] to evaluate the Cross-Subject (xsub) and Cross-View (xview) protocols for NTU60, and the Cross-Subject (xsub) and Cross-Setup (xset) protocols for NTU120.

Table 1. Performance comparison (%) for 3D action recognition on NTU RGB+D 60 and NTU RGB+D 120 datasets. We follow the setting of [42, 64].

Methods	NTU60		NTU120	
	xsub	xview	xsub	xset
ST-GCN [88]	81.5	88.3	70.7	73.2
2s-AGCN [67]	88.5	95.1	82.2	84.1
Shift-GCN [12]	90.7	96.5	85.9	87.6
MS-G3D [48]	91.5	96.2	86.9	88.4
DSTA-Net [68]	91.5	96.4	86.6	89.0
CTR-GCN [11]	92.4	96.8	88.9	90.6
PoseConv3D [21]	**94.1**	**97.1**	86.9	90.3
InfoGCN [13]	93.0	**97.1**	**89.8**	**91.2**
UPS$_{\text{separate}}$	89.6	93.1	85.1	87.8
UPS	92.6	97.0	89.3	91.1

Table 2. Performance comparison (%) for 2D action recognition on Kinetics 400 dataset. We follow the setting of [88] and report Top-1 and Top-5 recognition accuracy.

Methods	Top-1 (%)	Top-5 (%)
ST-GCN [88]	30.7	52.8
2s-AGCN [67]	36.1	58.7
DGNN [66]	36.9	59.6
GCN-NAS [55]	37.1	60.1
MS-AAGCN [69]	37.8	61.0
Sybio-GNN [37]	37.2	58.1
UPS$_{\text{separate}}$	36.2	59.4
UPS	**40.5**	**63.3**

Results. Following previous works [13, 42, 48], we employ the Top-1 classification accuracy metric. As shown in Tab. 1, our UPS model achieves good performance that is comparable to the state-of-the-art, demonstrating the efficacy of our method for 3D action recognition. Across all evaluation protocols on NTU60 and NTU120, we observe that by sharing one single model, UPS achieves better performances compared to UPS$_{\text{separate}}$. This demonstrates the efficacy of our method in incorporating diverse tasks into one model.

5.2. 2D Action Recognition

The 2D skeleton action recognition task is where we predict the action class of a 2D pose sequence.

Dataset. Kinetics 400 [33] is a widely used dataset that contains 400 action classes. It consists of more than 306k video clips. Following previous works [88], we extract 2D pose sequences using the OpenPose [6] toolbox. We follow the train-test split of previous works [37, 67, 88].

Results. Following previous works [88], we employ the Top-1 and Top-5 accuracy metrics. Note that, to unify the input formats for our joint tokenizer, for 2D skeletons we manually pad the third axis (Z-dim) with all zeros. As shown in Tab. 2, even when trained separately, our

Table 3. MPJPE results (mm) for 3D pose estimation on Human3.6M. We follow the evaluation setting of [65].

Methods	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg
Pavllo *et al.* [54]	45.2	46.7	43.3	45.6	48.1	55.1	44.6	44.3	57.3	65.8	47.1	44.0	49.0	32.8	33.9	46.8
Lin *et al.* [40]	42.5	44.8	42.6	44.2	48.5	57.1	42.6	41.4	56.5	64.5	47.4	43.0	48.1	33.0	35.1	46.6
Cai *et al.* [5]	44.6	47.4	45.6	48.8	50.8	59.0	47.2	43.9	57.9	61.9	49.7	46.6	51.3	37.1	39.4	48.8
Xu *et al.* [81]	37.4	43.5	42.7	42.7	46.6	59.7	41.3	45.1	52.7	60.2	45.8	43.1	47.7	33.7	37.1	45.6
Wang *et al.* [74]	41.3	43.9	44.0	42.2	48.0	57.1	42.2	43.2	57.3	61.3	47.0	43.5	47.0	32.6	31.8	45.6
Liu *et al.* [46]	41.8	44.8	41.1	44.9	47.4	54.1	43.4	42.2	56.2	63.6	45.3	43.5	45.3	31.3	32.2	45.1
Zeng *et al.* [91]	46.6	47.1	43.9	41.6	45.8	49.6	46.5	40.0	53.4	61.1	46.1	42.6	43.1	31.5	32.6	44.8
Zheng *et al.* [96]	41.5	44.8	39.8	42.5	46.5	51.6	42.1	42.0	53.3	60.7	45.5	43.3	46.1	31.8	32.2	44.3
Chen *et al.* [7]	41.4	43.5	40.1	42.9	46.6	51.9	41.7	42.3	53.9	60.2	45.4	41.7	46.0	31.5	32.7	44.1
Shan *et al.* [65]	38.4	42.1	39.8	40.2	45.2	48.9	40.4	38.3	53.8	57.3	43.9	41.6	42.2	29.3	29.3	42.1
$\text{UPS}_{\text{separate}}$	39.4	44.2	38.0	42.5	43.6	52.5	40.9	49.2	53.6	70.5	43.5	45.3	48.1	30.0	31.9	44.9
UPS	37.5	39.2	36.9	40.6	39.3	46.8	39.0	41.7	50.6	63.5	40.4	37.8	44.2	26.7	29.1	**40.8**

Table 4. P-MPJPE results (mm) for 3D pose estimation on Human3.6M. We follow the evaluation setting of [65].

Methods	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg
Liu *et al.* [40]	32.5	35.3	34.3	36.2	37.8	43.0	33.0	32.2	45.7	51.8	38.4	32.8	37.5	25.8	28.9	36.8
Pavllo *et al.* [54]	34.1	36.1	34.4	37.2	36.4	42.2	34.4	33.6	45.0	52.5	37.4	33.8	37.8	25.6	27.3	36.5
Cai *et al.* [5]	35.7	37.8	36.9	40.7	39.6	45.2	37.4	34.5	46.9	50.1	40.5	36.1	41.0	29.6	33.2	39.0
Xu *et al.* [81]	31.0	34.8	34.7	34.4	36.2	43.9	31.6	33.5	42.3	49.0	37.1	33.0	39.1	26.9	31.9	36.2
Liu *et al.* [46]	32.3	35.2	33.3	35.8	35.9	41.5	33.2	32.7	44.6	50.9	37.0	32.4	37.0	25.2	27.2	35.6
Wang *et al.* [74]	32.9	35.2	35.6	34.4	36.4	42.7	31.2	32.5	45.6	50.2	37.3	32.8	36.3	26.0	23.9	35.5
Chen *et al.* [7]	33.1	35.3	33.4	35.9	36.1	41.7	32.8	33.3	42.6	49.4	37.0	32.7	36.5	25.5	27.9	35.6
Zheng *et al.* [96]	32.5	34.8	32.6	34.6	35.3	39.5	32.1	32.0	42.8	48.5	34.8	32.4	35.3	24.5	26.0	34.6
Shan *et al.* [65]	31.3	35.2	32.9	33.9	35.4	39.3	32.5	31.5	44.6	48.2	36.3	32.9	34.4	23.8	23.9	34.4
$\text{UPS}_{\text{separate}}$	32.4	35.6	31.4	35.3	34.3	40.7	32.9	38.7	44.6	54.1	36.9	33.5	40.6	25.9	28.0	36.3
UPS	30.3	32.2	30.8	33.1	31.1	35.2	30.3	32.1	39.4	49.6	32.9	29.2	33.9	21.6	24.5	**32.5**

$\text{UPS}_{\text{separate}}$ obtains good recognition accuracy. For instance, as compared to Sybio-GNN [37], our basic $\text{UPS}_{\text{separate}}$ achieves a 1.3% higher Top-5 recognition accuracy. By unifying all tasks together, our full unified UPS model outperforms $\text{UPS}_{\text{separate}}$ by a large margin, and achieves state-of-the-art performance compared to all previous approaches on both Top-1 and Top-5 accuracy. This suggests that by integrating different heterogeneous tasks together, our UPS model can take advantage of each task to further benefit the learning process for 2D skeleton-based action recognition.

5.3. 3D Pose Estimation

In 3D pose estimation, we take 2D pose sequences as inputs and predict the corresponding 3D joint coordinates.

Dataset. Human3.6M [28] contains 3.6 million human poses, and is one of the largest motion capture datasets. In this dataset, skeleton data of the subjects performing various activities are captured via motion capture. We follow the train-test split of prior works [5, 54, 65], and use 5 subjects for training and 2 subjects for testing. Following previous works [5, 46, 54, 65, 96], we use CPN [10] to extract 2D keypoints, and train our model on the detected 2D pose sequences.

Results. Following previous works [5, 65, 91, 96], we report the mean per joint position error (MPJPE) and the Procrustes MPJPE (P-MPJPE) for each action. MPJPE calculates the average Euclidean distance between the predicted joint positions and the ground truth positions. P-MPJPE computes the predicted results after the predicted 3D poses are aligned to the ground truth via a rigid transformation. As shown in Tab. 3 and Tab. 4, our UPS model achieves good performance on both metrics, and obtains lower error as compared to the separately-trained $\text{UPS}_{\text{separate}}$, demonstrating the efficacy of our method for 3D pose estimation.

Table 5. Performance comparison (%) of 3D early action prediction on NTU60 dataset. We follow the evaluation setting of [39, 52, 78].

Methods	Observation Ratios on NTU60		
	20%	40%	60%
Jain *et al.* [29]	7.07	18.98	44.55
Ke *et al.* [34]	8.34	26.97	56.78
Weng *et al.* [78]	35.56	54.63	67.08
Aliakbarian *et al.* [63]	27.41	59.26	72.43
Wang *et al.* [77]	35.85	58.45	73.86
Pang *et al.* [52]	33.30	56.94	74.50
Tran *et al.* [72]	24.60	57.70	76.90
Ke *et al.* [35]	32.12	63.82	77.02
Li *et al.* [39]	42.39	72.24	82.99
Foo *et al.* [23]	**53.98**	74.34	85.03
$\text{UPS}_{\text{separate}}$	50.11	69.84	82.59
UPS	53.25	**75.06**	**85.35**

Table 6. Performance comparison (%) of 3D early action prediction on NTU120 dataset. We follow the previous work [23] and evaluate our UPS using xsub protocol.

Methods	Observation Ratios on NTU120		
	20%	40%	60%
Foo *et al.* [23]	31.73	45.67	67.08
$\text{UPS}_{\text{separate}}$	30.46	45.91	66.76
UPS	**33.35**	**48.12**	**69.95**

5.4. Early Action Prediction

In early action prediction, our goal is to correctly predict action classes before the action are fully executed.

Dataset. We use the NTU RGB+D 60 [64] and NTU

RGB+D 120 [42] datasets. We follow the existing works [23, 35, 39] and evaluate our UPS on xsub protocol .

Results. We evaluate the prediction performance of our UPS when only 20%, 40% and 60% of frames are observed, and the results are shown in Tab. 5. Our UPS achieves promising prediction results compared to other state-of-the-art approaches. It is noteworthy that all the existing approaches are specifically designed for early action prediction task while our UPS is able to tackle different tasks at the same time.

6. Ablation Study

To show the efficacy of our proposed UPS, we conduct extensive ablation experiments on three tasks, i.e., 3D action recognition, 3D early action prediction and 3D pose estimation. For 3D action recognition and 3D early action prediction, we conduct experiments on the xsub protocol of NTU RGB+D 120 dataset and report Top-1 accuracy. As for 3D pose estimation, the ablation experiments are conducted on the Human 3.6M dataset (using CPN to extract 2D pose) and we report the MPJPE metric.

Impact of depth of UPS encoder. We conduct ablation studies on the impact of UPS encoder's depth ($L_{encoder}$) in Tab. 7. We find that, when we increase the depth above 3, the performance does not improve further for all three tasks.

Impact of number of blocks in a layer (H). In Tab. 8 we ablate the decision regarding our setting of H. We find that, when we increase the H above 2, to 5, the results do not improve further. We also note that setting $H = 2$ provides a significantly improved performance from $H = 1$, which is equivalent to not having dynamic routing.

Impact of Routing Mechanism. We evaluate the efficacy of our proposed routing mechanism qualitatively and quantitatively. The qualitative visualization is shown in Fig. 4, which show that our tasks can share blocks or use separate blocks in each layer. As we can see, different tasks tend to share different blocks in different layers. The pose estimation and action prediction tasks tend to share the same block in the earlier two layers. However, for the last two layers, the action recognition and action prediction tasks tend to share parameters. This is possibly because the action recognition and action prediction tasks both require action classes as outputs. We also quantitatively evaluate the impact of the dynamic routing mechanism in Tab. 9. We find that our UPS consistently outperforms the variant without dynamic routing (*UPS w/o DR*), which demonstrates the efficacy of the dynamic routing mechanism.

7. Conclusion

In this paper, we propose a novel Unified Pose Sequence Modeling approach to unify three different pose-based human behavior understanding tasks, which are action recognition, 3D pose estimation and early action prediction. We leverage sequence modeling and text prompts to unify text-based activity categories and coordinate-based human joints into one single model. We also propose a dynamic routing mechanism to encourage different tasks to share common knowledge and avoid unwanted interference by adaptively sharing different subsets of parameters.

Table 7. Evaluation of depth of UPS encoder.

Depth	Action Recognition ↑	Pose Estimation ↓	Early Action Prediction ↑		
			20%	40%	60%
1	88.0	42.0	52.05	74.14	84.29
3	89.3	40.8	53.25	75.06	85.35
5	89.2	40.8	53.21	75.07	85.22

Table 8. Evaluation of number of blocks in each layer.

Blocks	Action Recognition ↑	Pose Estimation ↓	Early Action Prediction ↑		
			20%	40%	60%
1	88.6	41.7	51.80	73.26	83.92
2	89.3	40.8	53.25	75.06	85.35
5	89.1	40.9	53.24	75.07	85.33

Table 9. Evaluation of dynamic routing (DR) mechanism.

Setting	Action Recognition ↑	Pose Estimation ↓	Early Action Prediction ↑		
			20%	40%	60%
UPS w/o DR	87.0	42.8	53.02	73.47	84.15
UPS	89.3	40.8	53.25	75.06	85.35

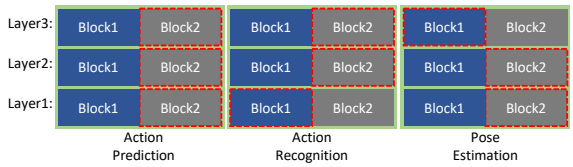

Figure 4. Qualitative visualization of block selection. Overall, there are three layers in the UPS encoder, and each layer has two blocks. Here, for each task, we indicate the selected blocks in each layer with a red box. We can see that the action recognition and the action prediction tend to share blocks in the later layers, i.e., the last two layers, while the pose estimation and the action prediction tend to share blocks in the earlier two layers.

Acknowledgments

This work is supported by MOE AcRF Tier 2 (Proposal ID: T2EP20222-0035), National Research Foundation Singapore under its AI Singapore Programme (AISG-100E-2020-065), and SUTD SKI Project (SKI 2021_02_06). This work is also supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215.

References

[1] Sharath Chandra Akkaladevi and Christoph Heindl. Action recognition for human robot interaction in industrial applications. In *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*, pages 94–99. IEEE, 2015. 1

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 2

[3] Joachim Bingel and Anders Søgaard. Identifying beneficial task relations for multi-task learning in deep neural networks. *arXiv preprint arXiv:1702.08303*, 2017. 2

[4] Erkan Bostanci, Nadia Kanwal, and Adrian F Clark. Augmented reality applications for cultural heritage using kinect. *Human-centric Computing and Information Sciences*, 5(1):1–18, 2015. 1

[5] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2272–2281, 2019. 7

[6] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 6

[7] Tianlang Chen, Chen Fang, Xiaohui Shen, Yiheng Zhu, Zhili Chen, and Jiebo Luo. Anatomy-aware 3d human pose estimation with bone-based pose decomposition. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):198–209, 2021. 7

[8] Ting Chen, Saurabh Saxena, Lala Li, David J. Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. In *International Conference on Learning Representations*, 2022. 2, 3, 5

[9] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 2

[10] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018. 7

[11] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13359–13368, 2021. 1, 4, 6

[12] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 183–192, 2020. 1, 2, 6

[13] Hyung-gun Chi, Myoung Hoon Ha, Seunggeun Chi, Sang Wan Lee, Qixing Huang, and Karthik Ramani. Infogcn: Representation learning for human skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20186–20196, 2022. 2, 6

[14] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1931–1942. PMLR, 18–24 Jul 2021. 2, 5

[15] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020. 2

[16] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3150–3158, 2016. 2

[17] Jinliang Deng, Xiusi Chen, Renhe Jiang, Xuan Song, and Ivor W Tsang. A multi-view multi-task learning framework for multi-variate time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 2022. 2

[18] Chhavi Dhiman and Dinesh Kumar Vishwakarma. A review of state-of-the-art techniques for abnormal human activity recognition. *Engineering Applications of Artificial Intelligence*, 77:21–45, 2019. 1

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4

[20] Haodong Duan, Jiaqi Wang, Kai Chen, and Dahua Lin. Dg-stgcn: Dynamic spatial-temporal modeling for skeleton-based action recognition. *arXiv preprint arXiv:2210.05895*, 2022. 2

[21] Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2969–2978, 2022. 2, 6

[22] Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd annual meeting of the Association for Computational Linguistics and the 7th international joint conference on natural language processing (volume 2: short papers)*, pages 845–850, 2015. 2

[23] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. Era: Expert retrieval and assembly for early action prediction. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pages 670–688. Springer, 2022. 1, 2, 7, 8

[24] Jia Gong, Lin Geng Foo, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. Diffpose: Toward more reliable 3d pose estimation. In *Proceedings of the IEEE/CVF Confer-*

ence on Computer Vision and Pattern Recognition (CVPR), June 2023. 2

[25] Tanmay Gupta, Amita Kamath, Aniruddha Kembhavi, and Derek Hoiem. Towards general purpose vision systems: An end-to-end task-agnostic vision-language architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16399–16409, 2022. 1, 2

[26] Jingrui He and Rick Lawrence. A graphbased framework for multi-task multi-view learning. In *ICML*, pages 25–32, 2011. 2

[27] Jian-Fang Hu, Wei-Shi Zheng, Lianyang Ma, Gang Wang, and Jianhuang Lai. Real-time rgb-d activity prediction by soft regression. In *European Conference on Computer Vision*, pages 280–296. Springer, 2016. 2

[28] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 6, 7

[29] Ashesh Jain, Avi Singh, Hema S Koppula, Shane Soh, and Ashutosh Saxena. Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3118–3125. IEEE, 2016. 7

[30] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 5

[31] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 14(2):201–211, 1973. 1

[32] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021. 2

[33] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6

[34] Qiuhong Ke, Mohammed Bennamoun, Senjian An, Ferdous Sohel, and Farid Boussaid. A new representation of skeleton sequences for 3d action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3288–3297, 2017. 7

[35] Qiuhong Ke, Mohammed Bennamoun, Hossein Rahmani, Senjian An, Ferdous Sohel, and Farid Boussaid. Learning latent global network for skeleton-based action prediction. *IEEE Transactions on Image Processing*, 29:959–970, 2019. 1, 2, 7, 8

[36] Jungho Lee, Minhyeok Lee, Dogyoon Lee, and Sangyoon Lee. Hierarchically decomposed graph convolutional networks for skeleton-based action recognition. *arXiv preprint arXiv:2208.10741*, 2022. 2

[37] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Symbiotic graph neural networks for 3d skeleton-based human action recognition and motion prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6):3316–3333, 2022. 6, 7

[38] Tianjiao Li, Lin Geng Foo, Ping Hu, Xindi Shang, Hossein Rahmani, Zehuan Yuan, and Jun Liu. Token boosting for robust self-supervised visual transformer pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 2

[39] Tianjiao Li, Jun Liu, Wei Zhang, and Lingyu Duan. Hardnet: Hardness-aware discrimination network for 3d early activity prediction. In *European Conference on Computer Vision*, pages 420–436. Springer, 2020. 1, 2, 7, 8

[40] Jiahao Lin and Gim Hee Lee. Trajectory space factorization for deep video-based 3d human pose estimation. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2019. 7

[41] Jun Liu, Henghui Ding, Amir Shahroudy, Ling-Yu Duan, Xudong Jiang, Gang Wang, and Alex C Kot. Feature boosting network for 3d pose estimation. *IEEE transactions on pattern analysis and machine intelligence*, 42(2):494–501, 2019. 1, 2

[42] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. 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, 2020. 6, 8

[43] Jun Liu, Amir Shahroudy, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ssnet: scale selection network for online 3d action prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8349–8358, 2018. 1

[44] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European conference on computer vision*, pages 816–833. Springer, 2016. 1, 2, 4

[45] Kenkun Liu, Rongqi Ding, Zhiming Zou, Le Wang, and Wei Tang. A comprehensive study of weight sharing in graph networks for 3d human pose estimation. In *European Conference on Computer Vision*, pages 318–334. Springer, 2020. 1, 2

[46] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. Attention mechanism exploits temporal contexts: Real-time 3d human pose reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5064–5073, 2020. 7

[47] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 3, 4, 6

[48] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 143–152, 2020. 1, 2, 6

[49] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[50] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3994–4003, 2016. 2

[51] Erik Murphy-Chutorian and Mohan Manubhai Trivedi. Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness. *IEEE Transactions on intelligent transportation systems*, 11(2):300–311, 2010. 1

[52] Guoliang Pang, Xionghui Wang, Jianfang Hu, Qing Zhang, and Wei-Shi Zheng. Dbdnet: Learning bi-directional dynamics for early action prediction. In *IJCAI*, pages 897–903, 2019. 7

[53] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7025–7034, 2017. 2

[54] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7753–7762, 2019. 7

[55] Wei Peng, Xiaopeng Hong, Haoyu Chen, and Guoying Zhao. Learning graph convolutional network for skeleton-based human action recognition by neural searching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2669–2676, 2020. 6

[56] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 2

[57] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training, 2018. 2

[58] Bin Ren, Mengyuan Liu, Runwei Ding, and Hong Liu. A survey on 3d skeleton-based action recognition using learning method. *arXiv preprint arXiv:2002.05907*, 2020. 1, 2

[59] Alina Roitberg, Alexander Perzylo, Nikhil Somani, Manuel Giuliani, Markus Rickert, and Alois Knoll. Human activity recognition in the context of industrial human-robot interaction. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, pages 1–10. IEEE, 2014. 1

[60] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in Neural Information Processing Systems*, 32, 2019. 2, 3, 5

[61] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2

[62] Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017. 2

[63] Mohammad Sadegh Aliakbarian, Fatemeh Sadat Saleh, Mathieu Salzmann, Basura Fernando, Lars Petersson, and Lars Andersson. Encouraging lstms to anticipate actions very early. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 280–289, 2017. 7

[64] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016. 6, 7

[65] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. P-stmo: Pre-trained spatial temporal many-to-one model for 3d human pose estimation. In *European Conference on Computer Vision*, 2022. 7

[66] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7912–7921, 2019. 1, 2, 6

[67] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12026–12035, 2019. 1, 2, 4, 6

[68] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Decoupled spatial-temporal attention network for skeleton-based action-gesture recognition. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 6

[69] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with multi-stream adaptive graph convolutional networks. *IEEE Transactions on Image Processing*, 29:9532–9545, 2020. 1, 6

[70] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018. 2

[71] Theodoros Theodoridis and Huosheng Hu. Action classification of 3d human models using dynamic anns for mobile robot surveillance. In *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 371–376. IEEE, 2007. 1

[72] Vinh Tran, Niranjan Balasubramanian, and Minh Hoai. Progressive knowledge distillation for early action recognition. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 2583–2587. IEEE, 2021. 7

[73] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[74] Jingbo Wang, Sijie Yan, Yuanjun Xiong, and Dahua Lin. Motion guided 3d pose estimation from videos. In *European Conference on Computer Vision*, pages 764–780. Springer, 2020. 7

[75] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 2

[76] Wenqian Wang, Faliang Chang, Chunsheng Liu, Guangxin Li, and Bin Wang. Ga-net: a guidance aware network for skeleton-based early activity recognition. *IEEE Transactions on Multimedia*, 2021. 2

[77] Xionghui Wang, Jian-Fang Hu, Jian-Huang Lai, Jianguo Zhang, and Wei-Shi Zheng. Progressive teacher-student learning for early action prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3556–3565, 2019. 1, 2, 7

[78] Junwu Weng, Xudong Jiang, Wei-Long Zheng, and Junsong Yuan. Early action recognition with category exclusion using policy-based reinforcement learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(12):4626–4638, 2020. 1, 2, 7

[79] Yi-Feng Wu, De-Chuan Zhan, and Yuan Jiang. Dmtmv: a unified learning framework for deep multi-task multi-view learning. In *2018 IEEE international conference on big knowledge (ICBK)*, pages 49–56. IEEE, 2018. 2

[80] Yuling Xing and Jia Zhu. Deep learning-based action recognition with 3d skeleton: A survey, 2021. 1, 2

[81] Jingwei Xu, Zhenbo Yu, Bingbing Ni, Jiancheng Yang, Xiaokang Yang, and Wenjun Zhang. Deep kinematics analysis for monocular 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on computer vision and Pattern recognition*, pages 899–908, 2020. 7

[82] Li Xu, He Huang, and Jun Liu. Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9878–9888, 2021. 2

[83] Li Xu, Mark He Huang, Xindi Shang, Zehuan Yuan, Ying Sun, and Jun Liu. Meta compositional referring expression segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 2

[84] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3d human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16105–16114, 2021. 1, 2

[85] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018. 2

[86] Rui Yan, Jinhui Tang, Xiangbo Shu, Zechao Li, and Qi Tian. Participation-contributed temporal dynamic model for group activity recognition. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1292–1300, 2018. 1

[87] Rui Yan, Lingxi Xie, Jinhui Tang, Xiangbo Shu, and Qi Tian. Higcin: Hierarchical graph-based cross inference network for group activity recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 1

[88] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018. 1, 2, 6

[89] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016. 2

[90] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020. 2, 3, 5

[91] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020. 7

[92] Junhao Zhang, Yali Wang, Zhipeng Zhou, Tianyu Luan, Zhe Wang, and Yu Qiao. Learning dynamical human-joint affinity for 3d pose estimation in videos. *IEEE Transactions on Image Processing*, 30:7914–7925, 2021. 1

[93] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(12):5586–5609, 2022. 2

[94] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 94–108. Springer, 2014. 2

[95] Weixi Zhao, Weiqiang Wang, and Yunjie Tian. Graformer: Graph-oriented transformer for 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20438–20447, June 2022. 1, 2

[96] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3d human pose estimation with spatial and temporal transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11656–11665, October 2021. 1, 2, 7

[97] Chaoyang Zhu, Yiyi Zhou, Yunhang Shen, Gen Luo, Xingjia Pan, Mingbao Lin, Chao Chen, Liujuan Cao, Xiaoshuai Sun, and Rongrong Ji. Seqtr: A simple yet universal network for visual grounding, 2022. 2, 5