

Part-aware Unified Representation of Language and Skeleton for Zero-shot Action Recognition

Anqi Zhu¹, QiuHong Ke^{2,*}, Mingming Gong¹, James Bailey¹

¹The University of Melbourne; ²Monash University

Parkville VIC 3052, Australia; Wellington Road, Clayton VIC 3800, Australia

azzh1@student.unimelb.edu.au, qiuhong.ke@monash.edu, {mingming.gong, baileyj}@unimelb.edu.au

Abstract

While remarkable progress has been made on supervised skeleton-based action recognition, the challenge of zero-shot recognition remains relatively unexplored. In this paper, we argue that relying solely on aligning label-level semantics and global skeleton features is insufficient to effectively transfer locally consistent visual knowledge from seen to unseen classes. To address this limitation, we introduce Part-aware Unified Representation between Language and Skeleton (PURLS) to explore visual-semantic alignment at both local and global scales. PURLS introduces a new prompting module and a novel partitioning module to generate aligned textual and visual representations across different levels. The former leverages a pre-trained GPT-3 to infer refined descriptions of the global and local (body-part-based and temporal-interval-based) movements from the original action labels. The latter employs an adaptive sampling strategy to group visual features from all body joint movements that are semantically relevant to a given description. Our approach is evaluated on various skeleton/language backbones and three large-scale datasets, i.e., NTU-RGB+D 60, NTU-RGB+D 120, and a newly curated dataset Kinetics-skeleton 200. The results showcase the universality and superior performance of PURLS, surpassing prior skeleton-based solutions and standard baselines from other domains. The source codes can be accessed at <https://github.com/azzh1/PURLS>.

1. Introduction

Human action recognition (HAR) is an important topic in computer vision. As actions are the primary bridge for establishing communications between people and the outside world, HAR is used in many application domains, such as virtual reality [1, 37], automated driving [20, 44], video retrieval [49], and robotics [4, 39]. The visual input modal-

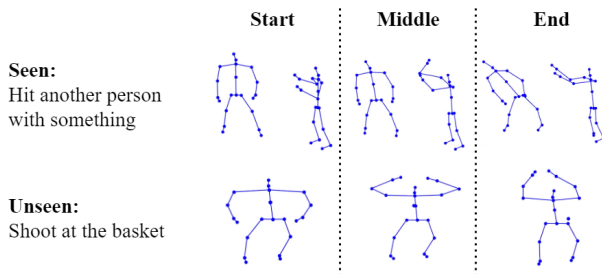


Figure 1. Examples of a seen class ('Hit another person with something') and an unseen class ('Shoot at the basket') from NTU-RGB+D 120 [21]. While humans can quickly identify their similar hand movements and use this knowledge to distinguish the new class from other unseen classes, label-based global feature learning does not facilitate the direct transfer of such local knowledge.

ity can vary, covering RGB videos, depth image sequences, point clouds, and skeleton sequences [34]. During its early stages and even until today, the advancement of HAR has mainly been driven by RGB-based solutions due to their natural data abundance [23, 35]. With the rise of pose prediction and depth-sensing technologies [6, 21, 30], 3-D skeleton sequences are becoming competitive substitutes that can reach high-accuracy prediction while cutting down computation, preserving profile privacy, and being robust by excluding background or color noises from action subjects. Due to these advantages, skeleton-based action recognition is attracting increasing attention in recent years [24, 28, 47].

While remarkable progress has been made in this area, most existing research [5, 11, 19, 22, 31, 32] focuses on recognizing actions in a fully supervised manner, i.e., their designs require annotated data of all action classes for model training. Nevertheless, gathering labeled data for every potential action class is impractical, especially for rare or perilous actions. Zero-shot learning (ZSL) is a research direction that aims to address this issue. Previous ZSL approaches have focused on training models to align label embeddings with visual encoding outputs that are globally av-

*Corresponding author

eraged from all skeleton features [15, 40, 48]. However, as illustrated in Fig. 1, actions that are globally dissimilar (*e.g.* Hit another person with something vs. Shoot at the basket) may still exhibit similar local visual movements. The understanding of these shared movements should remain transferable across seen and unseen classes to enhance the prior knowledge for recognizing new actions. On the other hand, global semantic alignment concentrates on the cross-modal consistency of overall body actions and does not encapsulate refined learning on such local visual concepts. This constrains the generalization capability of the learned representation when applied to unseen classes.

To overcome this issue, we present Part-aware Unified Representation of Language and Skeleton (PURLS), a novel framework that facilitates cross-modal semantic alignment at both global and local levels for prior knowledge exploitation. On the linguistic side, we start by enhancing the semantics of the original action labels using large language models. Specifically, we design a prompting module that employs GPT-3 [2] to generate detailed descriptions for the original actions and their spatially/temporally divisible local movements (*i.e.*, across human body parts/averaged temporal intervals). We then utilize a pre-trained text encoder from CLIP [26] to extract their textual features. For the visual aspect, a straightforward approach is to manually decompose a skeleton sequence into the corresponding global/local movement for a particular description and take the average features from the allocated body/temporal joints to perform alignment. Yet, this simple method limits the visual representation by strictly collecting features from static settings and thus may not always provide the most suitable alignment objects. Hence, we introduce a unique partitioning module that adaptively finds the weights for each joint feature to correlate with the given descriptions. This eventually leads the model to provide a more semantically relevant visual representation for alignment. During training, PURLS learns to project the closest visual \rightarrow textual manifolds at both global and local scales, ensuring semantic consistency with all descriptions in a balanced manner. This enables the projection layer to still distill part-aware knowledge when PURLS conducts prediction by only mapping global visual representations to label-level semantics during testing.

Considering that PURLS is built upon feature-level operations, we test our model using multiple skeleton/language backbones, and compare the results with previous skeleton-based ZSL solutions and classic ZSL benchmarks from other domains. The experiments follow the existing evaluation setups on *NTU-RGB+D 60* [30] and *NTU-RGB+D 120* [21], as well as additional setups with gradually increased unseen classes and decreased seen classes. We also evaluate the model’s performance on a new dataset setting, *Kinetics-skeleton 200*. Our results demonstrate that PURLS achieves

state-of-the-art performance in all experiments and exhibits robust universality and generalizability.

To summarize, our contributions are as follows:

- We propose PURLS, a new framework for the exploration and alignment of global and local visual concepts with enriched semantic information for zero-shot action recognition on skeleton sequences.
- PURLS offers an adaptive weight learning approach for partitioning spatial/temporal local visual representations to support local knowledge transfer from seen to unseen actions.
- PURLS is steadily compatible with different feature extraction backbones, and achieves state-of-the-art performance on three of the public large-scale datasets for skeleton recognition.

2. Related Work

2.1. Zero-shot Learning (ZSL)

Zero-shot Learning (ZSL) relies on training a model with samples from seen classes and their belonging class auxiliary information (*e.g.*, text descriptions, pre-trained attribute features) to develop its recognition ability for unseen categories (assuming that their auxiliary information is also available). The goal of the training is to enable the model to establish a generalizable and meaningful connection between the new visual features and prior semantic knowledge for the unseen classes. The basic methods begin with embedding-based models [12, 18, 45], which directly construct a universal visual-to-textual projection to find the nearest label neighbors by cosine distances. Kodirov [17] pioneered using auto-encoders in ZSL, where the training target is to encode images into semantic space and then decode them back to visual signals. When recognizing unseen classes, the model can either use the pre-trained encoder to project image features to label semantics or decode language embeddings to visual dimensions as class prototypes. Butcher [3] enabled linear metric learning with a cross-modality alignment by creating a shared embedding space transformed from both visual and label encodings. From another perspective, mimicking human’s learning habits, unseen subjects can be regarded as a new mix of visual concept components seen in training samples. [10] provided a generative approach in which the embedding alignment is established between disentangled local visual features and attribute-based text vectors. The disentanglement is realized by filtering out latent class-invariant features and verifying the decoding capacity of the remaining. In [8], a learnable attention module adaptively discovers the corresponding visual representation for each attribute. Applying similar intuition to RGB-based zero-shot action recognition, JigsawNet [25] recognized unseen actions by decomposing inputs in an unsupervised manner into atomic action prototypes that are

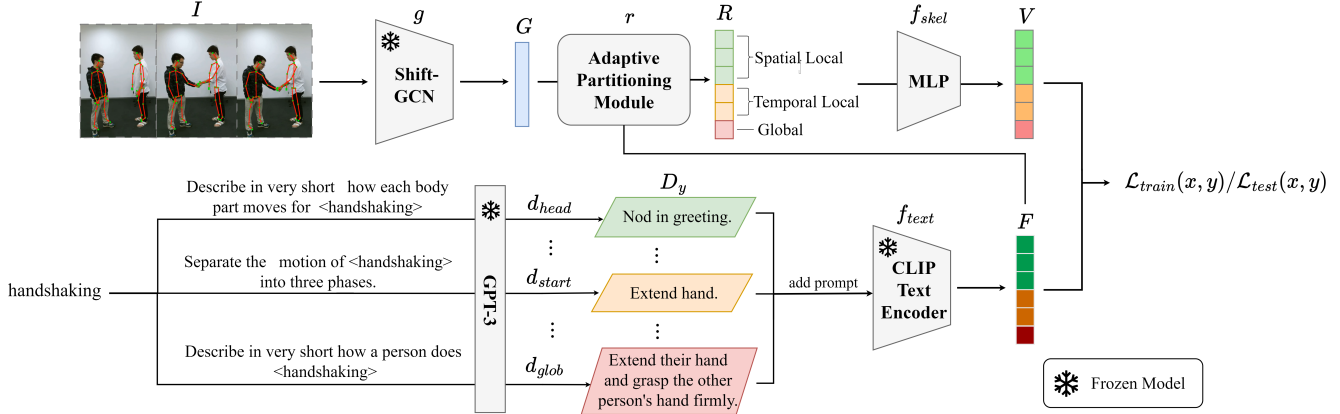


Figure 2. Architecture diagram for PURLS. The matching action label is sent to GPT-3 [2] to obtain detailed descriptions for its global/local body movements, whose textual features are generated by a pre-trained language encoder of CLIP [26]. The visual features of the input skeleton sequence I can be extracted from an arbitrary skeleton backbone g (e.g., Shift-GCN [11]) pre-trained on the seen classes. The output G is then fed to the partitioning module r to group the joint-level features into global and spatially/temporally-local representations in an adaptive manner, which are later projected with their corresponding description embeddings.

pre-memorized from seen classes.

Compared to solutions that match global visual features and label semantics, predictions based on local representations or attributes usually achieve more precise and robust performance due to their wider knowledge transfer. However, these local-based solutions are developed for pixel-format inputs or disentangled global features, which are incompatible with the irregular graph-format feature of skeleton sequences. PURLS is the first paper to implement an automatic knowledge extraction and transfer for locally decomposable visual concepts hidden in skeleton kinematics.

2.2. ZSL in Skeleton-based Action Recognition

While diverse techniques have been developed for ZSL with RGB format inputs, limited research has been conducted on approaches against skeleton sequences. In 2019, [40] proposed two standard methodologies adapted from traditional domains. These include a common-space metric learning using a Relation-Net framework and a visual \rightarrow semantic embedding-based classification using a DeViSE [14] model. Instead of learning to align every incoming pair of visual feature vectors and corresponding labels, [48] elucidated a more generalizable zero-shot prediction by learning to estimate and maximize the mutual information between overall visual and semantic distributions. In addition, the work designed a temporal rank loss to help the model capture more refined temporal information from frame-level visual features. SynSE-ZSL [15] was the first related work that considered skeleton-based local semantic matching. It learned a dual-modal feature representation by training the model to generate pseudo visual and linguistic samples from the opponent modality. The authors highlighted that action labels are often constituted by duplicated

verb and noun phrases, and the visual patterns of most verbs are repetitively learnable from multiple seen classes. Therefore, discriminating the knowledge transfer of verbs from label-level semantics can effectively improve the model’s generalization ability on unseen classes. We argue that a similar intuition can also be applied from the visual input side, in which PURLS mines spatially and temporally local visual concepts and uses language models to infer their aligned semantics from original labels.

2.3. Multi-modal Representation Learning

Multi-modal representation learning is highly relevant to ZSL, where different modalities can be mutually transformed and interpreted for information exchange. For the learning between image and language, CLIP[26] has provided a powerful backbone that utilizes contrastive learning to pre-train massive visual concepts from web data. The success of CLIP has constructed a universal representation manifold that captures shared semantics between RGB inputs and texts, which is widely used as a backbone reference for other downstream recognition tasks or ZSL baselines. In 3-D understanding, [46] enabled representation alignment for point cloud data by converting inputs into depth map images that fit the encoding format of CLIP. On the other hand, [42] proposed ULIP, which directly unified the projected embeddings of images, texts, and point cloud values. Their experiments showed that distilling the knowledge from the matching language-image manifold can effectively overcome the generalization shortage in the original modality. In skeleton learning, [41] explored importing comprehensive contrastive learning between static body-part-based skeleton features and their corresponding movement descriptions induced from original labels by GPT. Yet,

their method relies on data-driven training and focuses on refining supervised recognition. In this paper, we provide PURLS to support adaptive knowledge transfer from seen to unseen classes according to a more powerful manifold alignment against local movements extractable either spatially or temporally.

3. Proposed Approach

While distinct human actions may differ holistically, they often share similar local movements. The conventional training approach of directly aligning seen action labels with the overall representations of skeleton sequences fails to capture the semantic information of such local movements, thereby limiting the efficacy of zero-shot action recognition. To overcome this, PURLS adopts a two-step strategy. As shown in Fig. 2, it first focuses on the nuanced descriptions of each action label, considering global, spatially local, and temporally local perspectives. Subsequently, it uses a unique adaptive partitioning module to generate and align the visual representations from the corresponding skeleton joint features with every derived description. In this section, we first list our problem definition and introduce how we generate descriptions for both global and local movements from the original action labels. Following that, we expound on the process of partitioning the skeleton sequences for optimal feature alignment.

3.1. Problem Definition

Suppose $\mathcal{D}_{tr} = \{(x_{tr}^{sc}, y_{tr}^{sc})\}$ to be the set of N_{tr} training samples from available seen classes \mathcal{Y}^{sc} . A skeleton sequence $x_{tr}^{sc} \in \mathbb{R}^{L \times J \times M \times 3}$ records the 3-D locations of J body joints per actor in L frames. M is the maximum actor number per sequence. $y_{tr}^{sc} \in \mathcal{Y}^{sc}$ is the corresponding action label belonging to the seen class label set. Similarly, we let $\mathcal{D}_{te} = \{(x_{te}^{uc}, y_{te}^{uc})\}$ denote the set of N_{te} testing samples from the unseen classes \mathcal{Y}^{uc} . Under a standard ZSL setting, we have $\mathcal{Y}^{sc} \cap \mathcal{Y}^{uc} = \phi$. Training with only seen class samples, we expect the model to learn an extensive alignment of feature representations between the visual and language modalities, whose knowledge is efficiently transferrable to predict $\hat{y} \in \mathcal{Y}^{uc}$ during evaluation.

3.2. Creating Description-based Text Features

Inspired by human learning habits, we regard an action as a specific combination of local body movements that can be spatially or temporally decomposed. In addition to the label-level semantics, these local movements can also be independently learned as individual visual concepts transferable across different classes. To intelligently extract such underlying semantics, we adopt GPT-3 to produce textual descriptions for these movements at different scales. Tab. 1 and Tab. 2 show the questions and example answers we used for generating local and global descriptions to enrich the

original labels. For local movements, we design to generate detailed descriptions that are individually performed either by P ($P = 4$) body parts (*i.e.*, ‘head’, ‘hands’, ‘torso’, and ‘legs’) or in Z ($Z = 3$) contiguous temporal intervals (*e.g.*, ‘start’, ‘middle’, and ‘end’). To format the generated answers for each local part, we wrap the designed questions in a fixed prompt template as ‘Using the following format, <QUESTION>: <LOCAL PART 1> would: ...; <LOCAL PART 2> would ...; ...; <LOCAL PART H> would: ...’ where $H \in \{P, Z\}$ and <LOCAL PART i > refers to the corresponding local part name in P body parts ($i \in [0, P)$) or Z intervals ($i \in [0, Z)$). For the global semantic, we request GPT-3 to provide descriptions that augment the original label names. Note that one can also prepare different questions and generate multiple descriptions to calculate averaged text embeddings for later alignment. However, we consider that this does not lead to the key improvement in the later ZSL experiments, so we maintain using one question for each type of generation. After acquiring the targeted answers, we calculate their text embeddings using a pre-trained CLIP [26] text encoder f_{text} after converting them into standard prompts as “a (cropped/trimmed) video of [DESCRIPTION]”. For a given original label of $y \in \mathcal{Y}^{sc} \cup \mathcal{Y}^{uc}$, after GPT-3 produces its relevant descriptions $D_y = \{d_{head}, d_{hands}, \dots, d_{start}, \dots, d_{end}, d_{glob}\}$, we have $F = \parallel_d^{D_y} f_{text}(d) \in \mathbb{R}^{(P+Z+1) \times m}$ in which m refers to the text embedding dimension size and $(P + Z + 1)$ denotes the concatenation of P body-part-based, Z interval-based, and one global-based semantic encodings.

3.3. Partitioning Skeleton Feature Representations

Following [15], we first conduct the same padding and normalization pre-process from [11] to get the standard input $I \in \mathbb{R}^{L \times J \times M \times 3}$ of a raw skeleton sequence x and then adopt a pre-trained Shift-GCN [11] to extract its visual features $G = g(I) \in \mathbb{R}^{S \times n}$, where $S = L' \times J$. n is the skeleton encoding dimension and L' is the temporal feature dimension size after I being convoluted in g . To simplify the calculation, we average the features for M performers. To align with the output from the language branch, a partitioning module is further applied to extract the corresponding local and global representations from G .

A straightforward method to generate spatially local representations involves manually breaking down the skeleton joints into body parts as shown in Fig. 3. The feature of each body part can then be derived by averaging the features of the joints inside itself over the whole sequence. For temporal partitioning, one can averagely divide G along temporal dimensions into Z consecutive segments. The representation for each segment can then be computed by averaging the features of all body joints over the segment. The global representation can be achieved by averaging the features of all body joints over the whole sequence. We refer

Action	Question: Describe in very short how each body part moves for <Action>.			
	Head	Hands	Torso	Legs
Hit another person with something	Turn towards the other person.	Grip the object tightly and thrust it forward.	Twist and turn to generate momentum for the strike.	Stomp the ground to provide additional force for the strike.
Shoot at the basket	Turn and look up towards the basket.	Grip the basket and release it.	Twist and extend to generate power for the shot.	Bend slightly and propel slightly upward.

Table 1. Example body-part-based descriptions generated by GPT-3. The refined explanations correlate similar head and hand movements between ‘hit another person with something’ and ‘shoot at the basket’.

Action	Question: Separate the motion of <Action> into three phases.			Question: Describe in very short how a person does <Action>.
	Start	Middle	End	
Hit another person with something	Raise arm.	Swing arm.	Strike other person.	Swing their arm and strike the other person with the object.
Shoot at the basket	Raise arm.	Throw ball.	Aim at basket.	Raise their arm and throw the ball towards the basket.

Table 2. Example temporal-interval-based and global descriptions generated by GPT-3. The refined explanations correlate similar starting global postures between ‘hit another person with something’ and ‘shoot at the basket’.

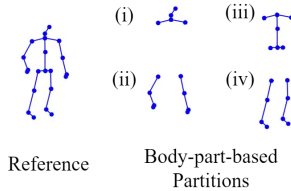


Figure 3. Spatial partitioning scheme for decomposing body joints into four body parts: (i) Head, (ii) Hands, (iii) Torso, (iv) Legs.

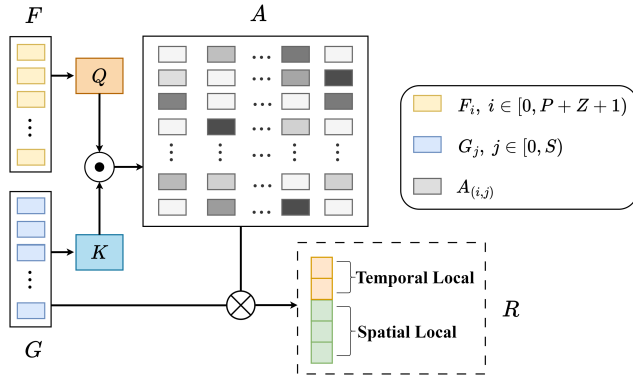


Figure 4. Illustration of how the adaptive partitioning module samples local visual representations.

to this method as **static partitioning**. While simple, these pre-defined partitions often exhibit instability in presenting the visual information that matches their corresponding descriptions. Below, we discuss our reasoning behind this observation and present our solution to resolve the issue.

Adaptive partitioning: Static partitioning extracts local movements from fixed allocated node features. This requires considerable manual examination of potential training datasets to determine the most suitable division prin-

ciples. Furthermore, local recognition can benefit from detecting its combinative context postures in other body parts or intervals. For example, the leg-lifting movement for walking can be more robustly recognized by simultaneously acknowledging an arm-swinging movement. Therefore, a more flexible approach for generating a local representation is to adaptively sample all description-relevant node features from G . Fig. 4 presents our cross-attention-based adaptive partitioning module. To represent a particular body part or interval, we identify the nodes semantically related to its description in terms of spatial and temporal dimensions and then contribute their visual information based on a correlation weight. Specifically, with the text embedding F and the visual output G , we define $Q = FW_Q$ as the language queries, and $K = GW_K$ as the visual keys. $W_Q \in \mathbb{R}^{m \times h}$ and $W_K \in \mathbb{R}^{n \times h}$ are learnable linear transformation matrices, where h is the projection dimension size. The module estimates an attention matrix $A \in \mathbb{R}^{(P+Z+1) \times S}$ by applying a cross product between Q and K , followed by a normalization and a softmax process as

$$A = \text{softmax}((Q \times K^T)/\sqrt{h}). \quad (1)$$

Intuitively, denoting the i -th row of A as $A_i \in \mathbb{R}^{1 \times S}$, it calculates the respective semantic relevance for all S nodes against the i -th description d_i ($i \in [0, P + Z + 1]$). Hence, the paired visual representation $R_i \in \mathbb{R}^n$ for d_i can be computed as a weighted sum from all node features, where the weights are defined by A_i . Promoting this to a matrix-level calculation, we can compute the paired visual representations $R \in \mathbb{R}^{(P+Z+1) \times n}$ for all $P + Z + 1$ descriptions as

$$R = AG. \quad (2)$$

3.4. Aligning dual-modal representations

A pre-trained CLIP model [26] understands a wide range of visual concepts shared among natural language and images. To realize part-aware matching between skeleton sequences and texts, we design PURLS to evenly map each visual representation $R_i \in \mathbb{R}^n$ (the i -th row of R) to an aligned distribution of its description encoding $F_i \in \mathbb{R}^m$ (the i -th row of F). As shown in Fig. 2, we construct an MLP layer f_{skel} to project each R_i to the textual embedding space as $V_i \in \mathbb{R}^m$. Then, we conduct contrastive learning between V_i and F_i as follows:

$$\mathcal{L}(V_i, F_i) = -\frac{1}{2} \log \frac{\exp(\frac{V_i F_i}{\tau})}{\sum_o \mathcal{Y}^{sc} \exp(\frac{V_i F_i^o}{\tau})} - \frac{1}{2} \log \frac{\exp(\frac{V_i F_i}{\tau})}{\sum_w \exp(\frac{V_i^w F_i}{\tau})}, \quad (3)$$

where F_i^o refers to the text embedding of the i -th description from other o (negative) seen action classes, and V_i^w is the i -th projected visual embedding from other w (negative) skeleton samples in the same batch. The temperature parameter τ controls the training gradient. The overall training loss of PURLS is a weighted sum of alignment losses for all local and global representations:

$$\mathcal{L}_{train}(x, y) = \sum_{i=0}^{P+Z} \alpha_i \mathcal{L}(V_i, F_i), \quad (4)$$

in which the weights α_i are either set to $1/(P+Z+1)$ or learnable. During the training, f_{skel} controls the visual \rightarrow textual mapping for all representations and thus learns to adaptively balance the semantic alignments for local and global projections. This helps f_{skel} to distill local-aware interaction knowledge when it encodes for global representations. Therefore, at the testing stage, given an input x_{te} , we can simplify the inference and predict \hat{y}_{te} that yields the lowest alignment loss directly on its global representation:

$$\mathcal{L}_{test}(x_{te}, y) = -\frac{1}{2} \log \frac{\exp(\frac{V_k^{te} F_k}{\tau})}{\sum_o \mathcal{Y}^{uc} \exp(\frac{V_k^{te} F_k^o}{\tau})}, \quad k = P+Z, \quad (5)$$

$$\hat{y}_{te} = \arg \min_{y \in \mathcal{Y}^{uc}} \mathcal{L}_{test}(x_{te}, y). \quad (6)$$

4. Experiments

4.1. Datasets

NTU-RGB+D 60 [30] contains 56,880 skeleton sequence samples of 60 actions, with 40 individual subjects captured from 80 distinct camera viewpoints. Each sample provides a temporal sequence of the 3-D location coordinates for 25 human body joints per performer. The maximum performer number is 2, and the coordinate values are padded

as 0 when the corresponding performer is unavailable (e.g. single-person actions). We use the two splits suggested by [15] - a 55/5 split (with 55 seen classes and 5 randomly chosen unseen classes) and a 48/12 split. We then create two more difficult splits of 40/20 and 30/30 to further challenge the generalization ability of our solutions on more unseen classes with less available training.

NTU-RGB+D 120 [21] is an enlarged dataset based on **NTU-RGB+D 60** and includes 60 additional action classes. It contains 114,480 samples for 120 actions performed by 106 individual subjects captured from 155 distinct camera viewpoints. Analogous to the above, we use two existing splits of 110/10 and 96/24 in [15] and two new splits of 80/40 and 60/60 for our evaluation setups.

Kinetics-skeleton 200 is a customized subset containing samples from the first 200 classes of the *Kinetics-skeleton 400* dataset [43]. When reviewing the ZSL experimental setups in other domains, we find that the existing protocols for skeleton understanding are very limited, as only the *NTU-RGB+D* series provides standard ZSL benchmarks. This motivates us to establish initial benchmarks on other common action datasets. *Kinetics-skeleton 400* includes the skeleton sequences extracted from the samples of 400 human action classes in *Kinetics 400* [16]. Its classes range from daily activities to complex actions. Each category contains at least 400 YouTube video clips, from which the skeleton data is extracted using OpenPose [6] in [43]. During the experiments, we observed that the ZSL accuracy gradually diminished as the number of unseen classes increased to a certain large number. This is probably because the difficulty of the task eventually surpasses the feature extraction capacity of the pre-trained visual backbones, which is not a research focus of our paper. Therefore, we limit our learning scenarios to cover under 200 classes. Similarly, we create four splits of 180/20, 160/40, 140/60, and 120/80 for our setups. For the full experiment results on the complete *Kinetics-skeleton 400* and other small datasets (*NW-UCLA* [38], *UTD-MHAD* [7], and *UWA3D II* [27]), please refer to our supplementary materials.

4.2. Implementation Details

To prepare the skeleton backbone, we follow [15] and only use seen class samples to pre-train feature extraction for the setup of each split. The visual features are realized by the 256-dimensional penultimate layer feature from Shift-GCN [11] ($n = 256$). We use the GPT-3 DaVinci-003 model and the questions in Tab. 1 and Tab. 2 to generate detailed descriptions for the original action labels. The textual features are then realized by the 512-dimensional encoding output from CLIP [26] equipped with the frozen weight of ViT-B/32 ($m = 512$). For the architectural details of PURLS, we always set $P = 4$, $Z = 3$, $W_Q \in \mathbb{R}^{512 \times 150}$, $W_K \in \mathbb{R}^{256 \times 150}$ where $h = 150$. f_{skel} is a 2-layer MLP in which

Model	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)				Kinetics-skeleton 200 (Acc %)			
	55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60	180/20	160/40	140/60	120/80
ReViSE [36]	75.37	26.44	24.26	14.81	57.92	37.96	19.47	8.27	24.95	13.28	8.14	6.23
DeViSE [14]	77.61	35.80	26.91	18.45	61.52	40.91	19.50	12.19	22.22	12.32	7.97	5.65
JPoSE [40]	64.82	28.75	20.05	12.39	51.93	32.44	13.71	7.65	-	-	-	-
CADA-VAE [29]	76.84	28.96	16.21	11.51	59.53	35.77	10.55	5.67	-	-	-	-
SynSE [15]	75.81	33.30	19.85	12.00	62.69	38.70	13.64	7.73	-	-	-	-
SMIE [48]	77.98	40.18	-	-	65.74	45.30	-	-	-	-	-	-
Global	64.69	35.46	27.15	16.29	66.96	44.27	21.31	14.12	25.96	15.85	10.23	7.77
PURLS	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63	32.22	22.56	12.01	11.75

Table 3. Zero-shot action recognition results (%) on *NTU-RGB+D 60*, *NTU-RGB+D 120* and *Kinetics-skeleton 200*. Experiments for JPoSE, CADA-VAE, and SynSE on *Kinetic-skeleton 200* are omitted because their pre-trained text features from their work are not consistent with other customized approaches. The experiments for SMIE is excerpted from its original paper.

the size of each hidden layer is 512. In the experiments on *NTU-RGB+D 60* and *NTU-RGB+D 120*, we set $L = 300$, $J = 25$, $M = 2$. For *Kinetic-skeletons 200*, we adopt the same data input configs from [43], where $L = 300$, $J = 18$, $M = 2$.

For training details, the model is optimized by an Adam optimizer with a learning rate of $1e - 4$ and a batch size of 256. The training epoch number is set to 300 but allows early stops if the training accuracy does not improve in the latest 20 epochs. All of our experiments are conducted using PyTorch on one A100 GPU.

For experiment details, since few previous works are available for skeleton-based ZSL, we implemented some classic baselines used in RGB-based classification from scratch and also referred to the existing skeleton ZSL solutions from [15] and [48]. These include visual-to-language embedding models (DeViSE [14], JPoSE [40]), common-space embedding models (ReViSE [36]), generative solutions (CADA-VAE [29], SynSE [15]) and contrastive learning (SMIE [48]). Additionally, we have another baseline that only learns from the global feature alignment with the label-level encoding from CLIP. We mark this method as ‘Global’ in all of our evaluation tables. While JPoSE, CADA-VAE, SynSE, and SMIE have their original linguistic feature configurations, the text features used in other customized baselines are uniformly encoded by the same CLIP we use for PURLS. The results for ReViSE and DeViSE are better than their records in the previous papers [15, 48] as they use better language models for text embedding.

4.3. Results & Analysis

Tab. 3 presents the classification results using all mentioned baselines and PURLS under the given setups. The learning difficulty increases in the order of *NTU-RGB+D 60*, *NTU-RGB+D 120*, and *Kinetics-skeleton 200*. Under the same dataset, the setups become more challenging with the decrease of seen classes and the increase of unseen classes.

Our method gives the highest performing predictions in every experimental setting. We observe that all previous

Encoder	Descriptor	Model	NTU-RGBD 60 (Acc %)			
			55/5	48/12	40/20	30/30
AA [33]	GPT3	Global	62.79	28.09	25.66	13.86
AA [33]	GPT3	PURLS	76.75	32.39	31.00	21.86
CTR [9]	GPT3	Global	65.16	34.56	26.12	15.92
CTR [9]	GPT3	PURLS	79.97	39.42	32.26	24.59
DG [32]	GPT3	Global	64.28	34.04	27.63	16.71
DG [32]	GPT3	PURLS	80.41	41.06	33.77	25.12
PoseC3D [13]	GPT3	Global	63.45	35.71	27.88	20.66
PoseC3D [13]	GPT3	PURLS	81.14	41.60	34.47	28.11
Shift	GPT3	Global	64.69	35.46	27.15	16.29
Shift	GPT3	PURLS	79.23	40.99	31.05	23.52
Shift	GPT3.5	Global	66.49	38.01	26.31	17.35
Shift	GPT3.5	PURLS	79.17	40.98	30.07	19.95
Shift	GPT4	Global	64.71	40.76	25.68	20.58
Shift	GPT4	PURLS	81.53	41.90	27.28	21.45

Table 4. Ablation study on *NTU-RGB+D 60* (%) for examining the universality of PURLS by replacing the skeleton encoder backbone or the action descriptor.

Partitioning Strategy	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)			
	55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60
Global (Original)	64.69	35.46	27.15	16.29	66.96	44.27	21.31	14.12
Global (GPT-3)	78.50	33.47	29.21	22.27	64.89	47.15	25.16	17.46
Static	76.46	33.03	29.57	22.00	67.62	46.83	26.98	18.03
Adaptive	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63

Table 5. Ablation study (%) on *NTU-RGB+D 60* and *NTU-RGB+D 120* for using different alignment learning with/without partitioning strategies, including direct global feature alignment to label or global description semantics, and PURLS with static/adaptive partitioning.

baselines experience different levels of generalization deterioration when the ratio of seen classes reduces to a certain degree. Meanwhile, PURLS effectively mitigates this issue and consistently maintains its prediction preciseness.

4.4. Ablation Study

We borrowed the setups in the *NTU-RGB+D* series to conduct a careful ablation study on PURLS. We analyzed the method universality with auxiliary benefits from using description-based textual features and incorporating local

α_i	BP	TI	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)			
			55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60
-			78.50	33.47	29.21	22.27	64.89	47.15	25.16	17.46
Average	✓		76.68	37.80	30.92	22.20	68.11	30.93	24.36	18.67
Learnable	✓		76.32	37.62	29.06	21.91	71.73	40.92	23.49	19.13
Average		✓	78.65	38.80	28.14	22.69	55.73	50.67	27.50	17.50
Learnable		✓	77.70	40.69	28.84	22.46	71.26	46.13	24.43	18.57
Average	✓	✓	79.02	39.92	31.00	23.47	73.55	51.38	27.67	18.66
Learnable	✓	✓	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63

Table 6. Ablation study (%) on *NTU-RGB+D 60* and *NTU-RGB+D 120* for (1) using different α_i ($i \in [0, P + Z + 1]$) to sum for L_{train} , (2) adding body-part-based (BP) alignment learning, (3) adding temporal-interval-based (TI) alignment learning. Note that when L_{train} only contains global alignment learning (Row 1), α_i is not applicable.

semantic alignment. This includes the examination of four factors: (1) the universality among different skeleton backbones and description generators, (2) the disparity between using original labels and expanded descriptions for global feature alignment learning, (3) the efficiency of various partitioning strategies when sampling local visual concepts, (4) the respective influence of distilling spatially and temporally local knowledge for global prediction.

Universality: Tab. 4 illustrates the detailed performance of PURLS on *NTU-RGB+D 60* when it uses different skeleton encoders g for visual feature extraction and GPT models for description generation. As a comparison, we also test the replacements on the ‘Global’ baseline (*i.e.*, aligning between the globally averaged skeleton features and label semantics) to verify the improvements brought by our method. For the skeleton backbone, we considered alternatives from several state-of-the-art extractors, including AA-GCN[33], CTR-GCN[9], DG-STGCN[32], and PoseC3D[13]. In particular, PoseC3D is a unique backbone whose output format is a convoluted feature map from its 2-D heatmap input processed from an original skeleton sequence. In this situation, static partitioning is no longer compatible because it cannot pre-define which feature map pixels should belong to a specified body part. On the other hand, PURLS can still easily adapt itself to the new input structure by finding pixel-wise attention weights when generating a global/local visual representation. For the GPT descriptors, we attempted replacements based on model versions, iterating from GPT-3 to GPT-4. The results show that PURLS achieves an absolute advantage over ‘Global’ across all examined settings, revealing that our solution steadily supports a better ZSL ability for most skeleton backbones and language models.

Label Semantics vs. Description Semantics: The first two rows of Tab. 5 present the performance difference of only learning global feature alignment using label semantics or action description semantics. In most scenarios, description-based learning effectively boosts the results

with richer semantic correlations across seen and unseen classes.

Partitioning Strategy: The following two rows of Tab. 5 demonstrate the performance of PURLS using either static or adaptive partitioning. Using static partitioning shows unstable performance improvement compared to only learning global alignment from action descriptions. This is natural since the manual joint assignment may not capture all meaningful local movements in a given action but can also introduce noises. On the other hand, adaptive partitioning can effectively ameliorate these defects.

Local Semantics & Aggregation: In Tab. 6, we further examine the concrete improvement brought by distilling the knowledge of body-part-based and temporal-interval-based local movements from adaptive partitioning. For the aggregation method of different alignment losses, we provided two options to sum the contrastive loss $\mathcal{L}(V_i, F_i)$ for each global/local representation i , including either averaging the weight of each term or applying a learnable weight (see Eq. (4)). According to the results on Row 2-7, we find that the extra alignment losses from spatial and temporal dimensions can bring various prediction accuracy increases. This reveals that the model managed to extract local transferrable knowledge that improves the generalization of predictions for unseen classes. By adaptively distilling action-relevant knowledge from all possible global/local scales in a balanced manner, PURLS can achieve a robust recognition enhancement.

5. Conclusion

We have introduced a novel framework, PURLS, that globally and locally aligns language and skeleton feature representations. We implement this by leveraging label semantic enrichment with large language models, as well as adaptive node feature partitioning on the skeleton structure. This enables PURLS to transfer various visual knowledge from seen classes to unseen classes at different scales. Experimental results demonstrate that PURLS achieves state-of-the-art performance not only in the existing ZSL setups on the *NTU-RGB+D* series, but also in the more challenging setups of a customized dataset *Kinetics-skeleton 200*. Furthermore, PURLS shows powerful generality on different backbones and effectively mitigates the generalization drops when the pre-training on seen classes is severely limited. It holds promise for future related cross-modal tasks by providing flexible feature alignment between natural language and joint-based motion data.

6. Acknowledgement

This research was supported by The University of Melbourne’s Research Computing Services and the Petascale Campus Initiative.

References

- [1] Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1
- [2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. 2, 3
- [3] Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Improving semantic embedding consistency by metric learning for zero-shot classification. *CoRR*, abs/1607.08085, 2016. 2
- [4] Cesar Cadena, Anthony Dick, and Ian Reid. Multi-modal auto-encoders as joint estimators for robotics scene understanding. 2016. 1
- [5] Yi Cao, Chen Liu, Zilong Huang, Yongjian Sheng, and Yongjian Ju. Skeleton-based action recognition with temporal action graph and temporal adaptive graph convolution structure. 80(19):29139–29162, 2021. 1
- [6] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 1, 6
- [7] Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. Utdmhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 168–172, 2015. 6
- [8] Shiming Chen, Ziming Hong, Guo-Sen Xie, Wenhan Yang, Qinmu Peng, Kai Wang, Jian Zhao, and Xinge You. Msdn: Mutually semantic distillation network for zero-shot learning, 2022. 2
- [9] 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. 7, 8
- [10] Zhi Chen, Yadan Luo, Ruihong Qiu, Sen Wang, Zi Huang, Jingjing Li, and Zheng Zhang. Semantics disentangling for generalized zero-shot learning, 2021. 2
- [11] Ke Cheng, Yifan Zhang, Xiangyu He, Weihang Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 180–189, 2020. 1, 3, 4, 6
- [12] Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. Improving zero-shot learning by mitigating the hubness problem, 2015. 2
- [13] Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. *arXiv preprint arXiv:2104.13586*, 2021. 7, 8
- [14] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc' Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2013. 3, 7
- [15] Pranay Gupta, Divyanshu Sharma, and Ravi Kiran Sarvadev-abhatla. Syntactically guided generative embeddings for zero-shot skeleton action recognition. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 439–443, 2021. 2, 3, 4, 6, 7
- [16] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017. 6
- [17] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning, 2017. 2
- [18] Angeliki Lazaridou, Georgiana Dinu, and Marco Baroni. Hubness and pollution: Delving into cross-space mapping for zero-shot learning. pages 270–280, 2015. 2
- [19] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, pages 3595–3603. Computer Vision Foundation / IEEE, 2019. 1
- [20] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Bo Wu, Yifeng Lu, Denny Zhou, Quoc V. Le, Alan Yuille, and Mingxing Tan. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection, 2022. 1
- [21] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. NTU RGBd 120: A large-scale benchmark for 3d human activity understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(10):2684–2701, 2020. 1, 2, 6
- [22] Qiang Nie and Yunhui Liu. View transfer on human skeleton pose: Automatically disentangle the view-variant and view-invariant information for pose representation learning. *IJCV*, 129(1):1–22, 2021. 1
- [23] Behnoosh Parsa, Athma Narayanan, and Behzad Dariush. Spatio-temporal pyramid graph convolutions for human action recognition and postural assessment. pages 1069–1079. IEEE, 2020. 1
- [24] Quang-Tien Pham, Duc-Anh Nguyen, Tien-Thanh Nguyen, Thanh Nam Nguyen, Duy-Tung Nguyen, Dinh-Tan Pham, Thanh-Hai Tran, Thi-Lan Le, and Hai Vu. A study on skeleton-based action recognition and its application to physical exercise recognition. In *The 11th International Symposium on Information and Communication Technology, SoICT 2022, Hanoi, Vietnam, December 1-3, 2022*, pages 239–246. ACM, 2022. 1
- [25] Yijun Qian, Lijun Yu, Wenhe Liu, and Alexander G. Hauptmann. Rethinking zero-shot action recognition: Learning from latent atomic actions. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October*

- 23–27, 2022, *Proceedings, Part IV*, page 104–120, Berlin, Heidelberg, 2022. Springer-Verlag. [2](#)
- [26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. [2](#), [3](#), [4](#), [6](#)
- [27] Hossein Rahmani, Arif Mahmood, Du Huynh, and Ajmal Mian. Histogram of oriented principal components for cross-view action recognition, 2015. [6](#)
- [28] M. Rashmi and Ram Mohana Reddy Guddeti. Skeleton based human action recognition for smart city application using deep learning. In *2020 International Conference on COMMunication Systems & NETWORKS, COMSNETS 2020, Bengaluru, India, January 7-11, 2020*, pages 756–761. IEEE, 2020. [1](#)
- [29] E. Schonfeld, S. Ebrahimi, S. Sinha, T. Darrell, and Z. Akata. Generalized zero- and few-shot learning via aligned variational autoencoders. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8239–8247, Los Alamitos, CA, USA, 2019. IEEE Computer Society. [7](#)
- [30] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1010–1019, 2016. [1](#), [2](#), [6](#)
- [31] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, pages 12026–12035. Computer Vision Foundation / IEEE, 2019. [1](#)
- [32] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *CVPR*, pages 7912–7921. Computer Vision Foundation / IEEE, 2019. [1](#), [7](#), [8](#)
- [33] 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. [7](#), [8](#)
- [34] Zehua Sun, Qiuhong Ke, Hossein Rahmani, Mohammed Bennamoun, Gang Wang, and Jun Liu. Human action recognition from various data modalities: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–20, 2022. [1](#)
- [35] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, pages 6450–6459. Computer Vision Foundation / IEEE Computer Society, 2018. [1](#)
- [36] Yao-Hung Hubert Tsai, Liang-Kang Huang, and Ruslan Salakhutdinov. Learning robust visual-semantic embeddings, 2017. [7](#)
- [37] Thang Vu, Kookhoi Kim, Tung M. Luu, Xuan Thanh Nguyen, and Chang D. Yoo. Softgroup for 3d instance segmentation on point clouds, 2022. [1](#)
- [38] Jiang wang, Xiaohan Nie, Yin Xia, Ying Wu, and Song-Chun Zhu. Cross-view action modeling, learning and recognition, 2014. [6](#)
- [39] Christian Wojek, Stefan Walk, Stefan Roth, and Bernt Schiele. Monocular 3d scene understanding with explicit occlusion reasoning. pages 1993–2000, 2011. [1](#)
- [40] Michael Wray, Gabriela Csurka, Diane Larlus, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 450–459, 2019. [2](#), [3](#), [7](#)
- [41] Wangmeng Xiang, Chao Li, Yuxuan Zhou, Biao Wang, and Lei Zhang. Generative action description prompts for skeleton-based action recognition. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 10242–10251. IEEE, 2023. [3](#)
- [42] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding, 2023. [3](#)
- [43] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition, 2018. [6](#), [7](#)
- [44] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11784–11793, 2021. [1](#)
- [45] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning, 2019. [2](#)
- [46] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8552–8562, 2022. [3](#)
- [47] Yunkai Zhang, Yinghong Tian, Pingyi Wu, and Dongfan Chen. Application of skeleton data and long short-term memory in action recognition of children with autism spectrum disorder. *Sensors*, 21(2):411, 2021. [1](#)
- [48] Yujie Zhou, Wenwen Qiang, Anyi Rao, Ning Lin, Bing Su, and Jiaqi Wang. Zero-shot skeleton-based action recognition via mutual information estimation and maximization. In *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*, pages 5302–5310. ACM, 2023. [2](#), [3](#), [7](#)
- [49] Cunjuan Zhu, Qi Jia, Wei Chen, Yanming Guo, and Yu Liu. Deep learning for video-text retrieval: a review. *International Journal of Multimedia Information Retrieval*, 12(1):3, 2023. [1](#)

Part-aware Unified Representation of Language and Skeleton for Zero-shot Action Recognition

Supplementary Material

In this supplementary material, we present additional information to explain and support our design choices and their concrete effects in each methodology module of PURLS. Additionally, we list the remaining experiment performance that we omitted in the main paper and provide a qualitative visualization of the learning outcome of our solution.

1. Additional Information for Local Semantic Divison

PURLS aligns local visual semantics with global/body-part-based/temporal-interval-based descriptions against a given label. In the submission, we meticulously analyze the spatial and temporal local semantics when they are respectively extracted, resulting in a total of $P + Z + 1$ aligned representations. Alternatively, if we jointly consider spatial-temporal semantics, there will be $P * Z + 1$ aligned representations (*i.e.* finding temporal local movements for each spatial local area). To streamline computational costs, we opt for the approach of separately considering spatial and temporal local semantics.

2. Additional Information for Creating Description-based Text Features

The descriptions for each label from every evaluation dataset are provided in the supplement repository under the path of 'supplement/gpt3_desc.xlsx'. For body-part-based local movements, we generated descriptions for four body areas including 'Head', 'Hands', 'Torso', and 'Legs'. For temporal-interval-based local movements, we generated descriptions for three phases including 'Start', 'Middle', and 'End'. A manual inspection was applied to ensure that all descriptions properly enhance the original label semantics.

2.1. Hyperparameter selection for generating descriptions

In the main submission, we used only one question and one generated answer for each description type when preparing the text encoding inputs for CLIP. Meanwhile, we also attempted to use varying numbers of questions and answers to check their influence on the model performance. In Tab. 1, we present the wrapping prompts for each global/local part and all the alternative questions we designed for each type of description generation (see the design explanation in Sec. 3.2 of our main paper). Tab. 2 records the hyperparameter ablation when we use different numbers of questions

and answers for the generation. When multiple descriptions are generated, we respectively encode each answer and average all output features to calculate the linguistic embeddings later used for semantic alignments. Yet, unlike traditional supervised learning, we observed that including multiple questions and answers does not improve ZSL classification in most cases. Therefore, we only use one question and one answer for each description in our main experiments.

2.2. Preprocessing for CLIP inputs

A pre-trained text encoder from CLIP requires proper prompting on its input to ensure the backbone outputs fit the required downstream tasks. Hence, as shown in Tab. 3, we prepared customized prompting sentences to further standardize each generated description before encoding them.

3. Additional Information for Adaptive Partitioning

3.1. Hyperparameter selection for Local Representations

In the main submission, we choose to align local movements using four body-part-based (BP) partitions and three temporal-interval-based (TI) partitions (*i.e.* $P = 4, Z = 3$). Tab. 4 provides the hyperparameter ablation when we try different numbers of spatial/temporal local representations. In the case of BP, using one body part means directly calculating the global features along the spatial dimension. Using two body parts refers to splitting spatial features into upper and lower body movements. Using six body parts requires further decomposing hand and leg movements into individual single-hand and single-leg movements. In the case of TI, the original sequence is averagely divided into multiple time intervals according to the specified number. The results reveal that having four body parts and three temporal intervals yields the best performance for PURLS.

3.2. Visualization for Adaptive Partitioning

Fig. 1, Fig. 2, Fig. 3, and Fig. 4 visualize the learned weights for acquiring the corresponding visual representations towards the descriptions for the classes 'apply cream on face', 'running on the spot', 'kicking other person', and 'cutting nails'. We analyze the impacts of adaptive partitioning in four scenarios. Two spatial-based scenarios include learning the actions characterized by specific local body movements (*e.g.* 'applying cream on face') or the movement from

Description Type	Questions
Body-part-based	Using the following format, describe in very short how each body part moves / how a person’s body part would act / individual body part action for <Action>.
	Head would: { <i>text</i> }
	Hands would: { <i>text</i> }
	Torso would: { <i>text</i> }
	Legs would: { <i>text</i> }
Temporal-interval-based	Use the following format, separate in very short how a person performs the action of / the motion of / the sub-motions when a person carries out <Action> into three phases.
	1: The person would { <i>text</i> }.
	2: The person would { <i>text</i> }.
	3: The person would { <i>text</i> }.
Global	Using the following format, describe in very short the motion of a person who / the motion of a person carries out / how a person does <Action>.
	The person would { <i>text</i> }.

Table 1. The full version of the designed questions used to generate label-relative descriptions under different scales. The bold texts represent the question alternatives, while {*text*} refers to the blanks for GPT-3 to fill.

# Prompt question	# Description per question	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)			
		55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60
1	1	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63
2	1	75.44	45.64	26.38	24.72	61.87	45.28	25.10	15.76
3	1	76.90	43.63	26.37	24.67	63.00	43.96	25.01	15.73
1	5	78.50	41.57	27.25	22.02	67.59	44.06	26.32	16.39
1	10	77.7	41.99	27.01	22.33	66.51	44.60	26.56	16.96

Table 2. Hyperparameter ablation (%) of applying different numbers of prompt questions & generated descriptions on GPT-3. ‘# Prompt question’ refers to the number of used questions. ‘# Desc per question’ means the number of generated descriptions for each question.

the entire body (*e.g.* ‘running on the spot’). Two temporal-based scenarios include learning the actions that can be broken down into sequentially local movements (*e.g.* ‘kicking other person’), and the actions that involve repetitive movements (*e.g.* ‘cutting nails’). To showcase the different learning focus in a more intuitive manner, we demonstrate the respective feature sampling weight of each body joint on every temporal dimension.

Adaptive partitioning for spatially local movements:

For the actions characterized by specific local body movements (Fig. 1), against body-part-based descriptions (Row 1-4 in each phase), adaptive partitioning effectively highlights the feature sampling from the body joints belonging to the ‘hands’. It also assigns contextual significance to a few body-joint features from the ‘torso’ and ‘legs’ during the middle phase, as some co-movements may exist when raising arms to one’s face. The local representation of hand movements emphasizes these features the most. The representations for other descriptions learn each joint feature in a more averaged manner while still giving the most importance to hand features, as they become the most valuable context features. Against the actions characterized by the entire body movement (Fig. 2), we find the module more averagely samples the body-joint-level features for each description.

Adaptive partitioning for temporally local movements:

For the actions that can be broken down into sequentially local movements (Fig. 3), against temporal-interval-based descriptions (Row 5-7 in each phase), adaptive partitioning emphasizes the overall features in the third phase, which contains the most representative postures of the kicking movement. The representation for the ‘end’ phase assigns the highest weights to these features to represent its local description as ‘Strike other person.’ Meanwhile, multiple hand-related features are collected with higher priority in the first two phases, while some leg-related features are also collected in the second phase. This implies that PURLS also allocates attention to sampling the hand and leg movements for representing the descriptions of ‘Raise leg’ and ‘Extend foot’. Against the actions that repeat temporally local movements (Fig. 4), we observe a relatively balanced distribution of learning focus. In particular, the distribution tends to concentrate slightly more in the first phase. We believe this is because most relevant visual information is already available at the beginning. On the other hand, the ending phase may contain noises from the zero-padding frames.

Description Type	Prompts
Body-part-based	A cropped video of people’s head motions that <Description>.
	A cropped video of people’s hand motions that <Description>.
	A cropped video of people’s torso motions that <Description>.
	A cropped video of people’s leg motions that <Description>.
Temporal-interval-based	A trimmed video of the motion that <Description>.
Global	A video of people’s motion that <Description>.

Table 3. The customized prompts used to wrap each description before it is sent to CLIP.

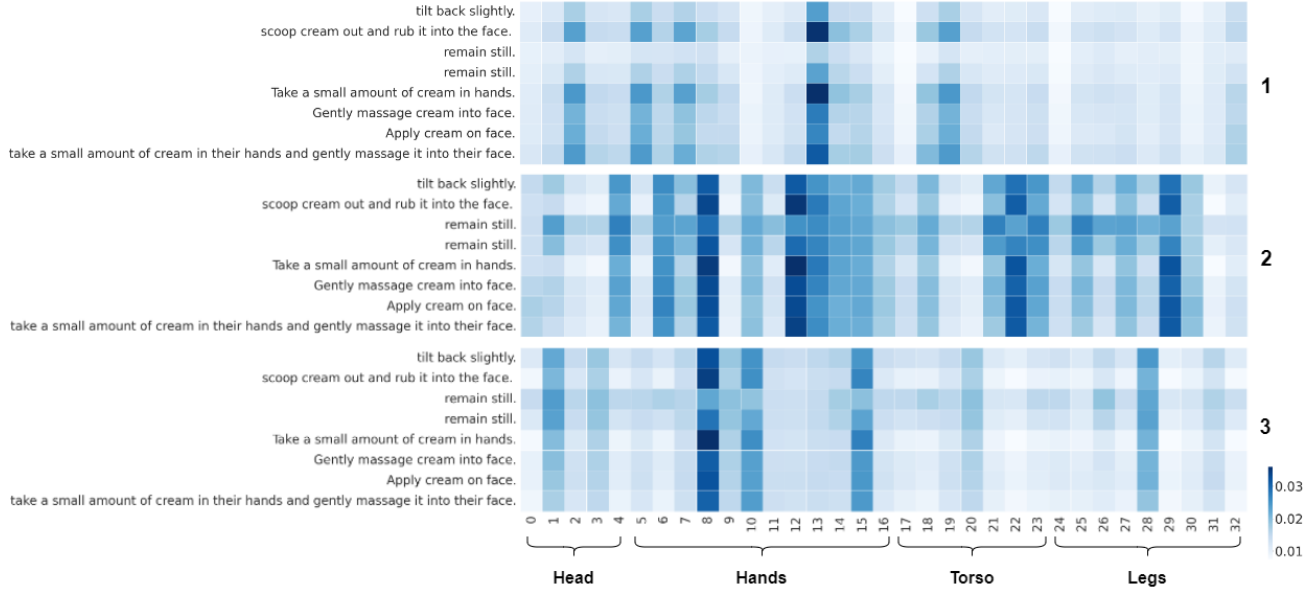


Figure 1. Visualized illustration of the learned adaptive weights used for sampling joint features to generate the visual representation of each description on ‘apply cream on face’. The horizontal axis lists each body joint, and the vertical axis is the convoluted temporal dimension with a length of L' . We trunk the temporal dimension into three groups for the phases of ‘start’ (1), ‘middle’ (2), and ‘end’ (3). And we label the default body part that each body joint usually belongs to according to Fig. 3 in our main paper. For the demonstration in one phase, from top to bottom is the feature weight distribution against the description for body-part-based (‘head’, ‘hands’, ‘torso’, ‘legs’), temporal-interval-base (‘start’, ‘middle’, ‘end’), and global semantics.

BP	TI	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)			
		55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60
1	3	77.70	40.69	28.84	22.46	71.26	46.13	24.43	18.57
2	3	78.31	33.15	30.81	23.03	72.77	45.90	26.26	19.65
4	3	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63
6	3	72.18	37.65	30.10	23.35	62.67	47.81	26.39	18.78
4	1	76.32	37.62	29.06	21.91	71.73	40.92	23.49	19.13
4	3	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63
4	6	74.54	38.89	28.70	23.03	71.00	48.81	26.11	17.71

Table 4. Hyperparameter ablation (%) on *NTU-RGB+D 60* and *NTU-RGB+D 120* of (1) using different numbers of body-part-based (BP) local partitioning, (2) using different numbers of temporal-interval-based (TI) local partitioning.

4. Additional Information for Experiments

4.1. Seen/unseen Split Setups

The specific lists of seen/unseen classes in each experiment split are provided as numpy files in the supplement

repository under the path of ‘supplement/label_splits’. Each split setting belongs to one of the following dataset folders: ‘ntu60’, ‘ntu120’, ‘kinetic200’, ‘kinetic400’, ‘nw-ucla’, ‘utd-mhad’, ‘uwa3dii’. The first three datasets are the ones we focus on in the main paper. For a given split, the file that contains the corresponding unseen class list is named as ‘ru + the number of unseen classes’. Similarly, the file recording the corresponding seen classes is named as ‘rs + the number of seen classes’. For example, for the split with 55 seen classes and 5 unseen classes on the *NTU-RGB+D 60* dataset, the corresponding seen and unseen class lists are recorded in the files of ‘supplement/label_splits/ntu60/rs55.npy’ and ‘supplement/label_splits/ntu60/ru5.npy’.

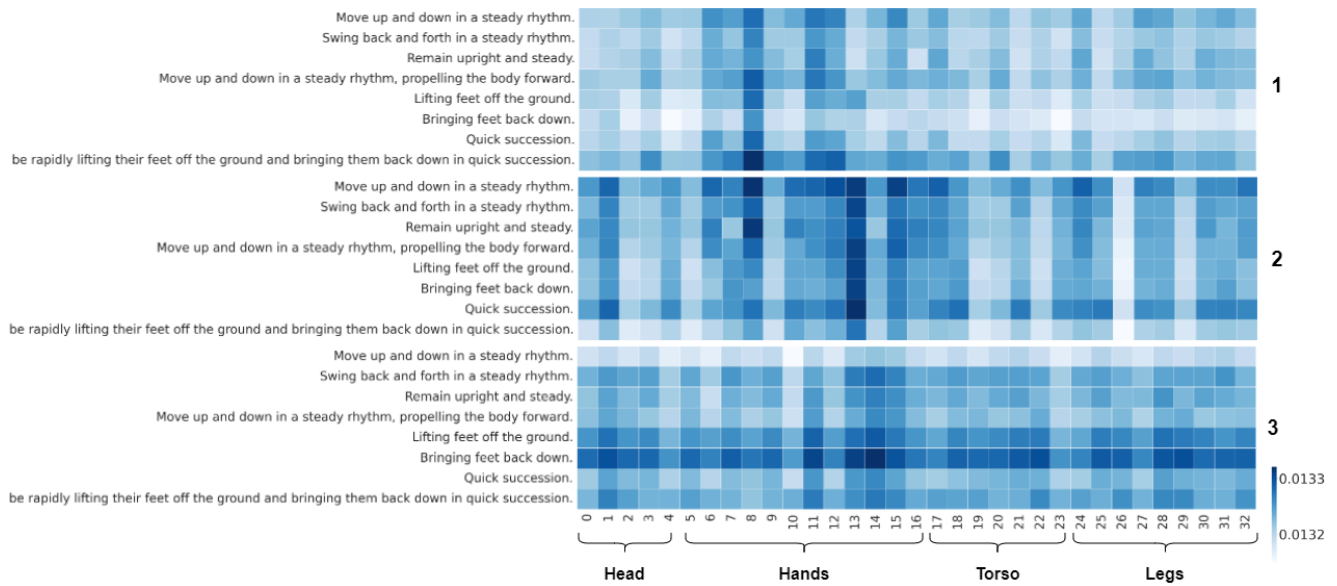


Figure 2. Visualized illustration of the learned adaptive weights for the class ‘running on the spot’.

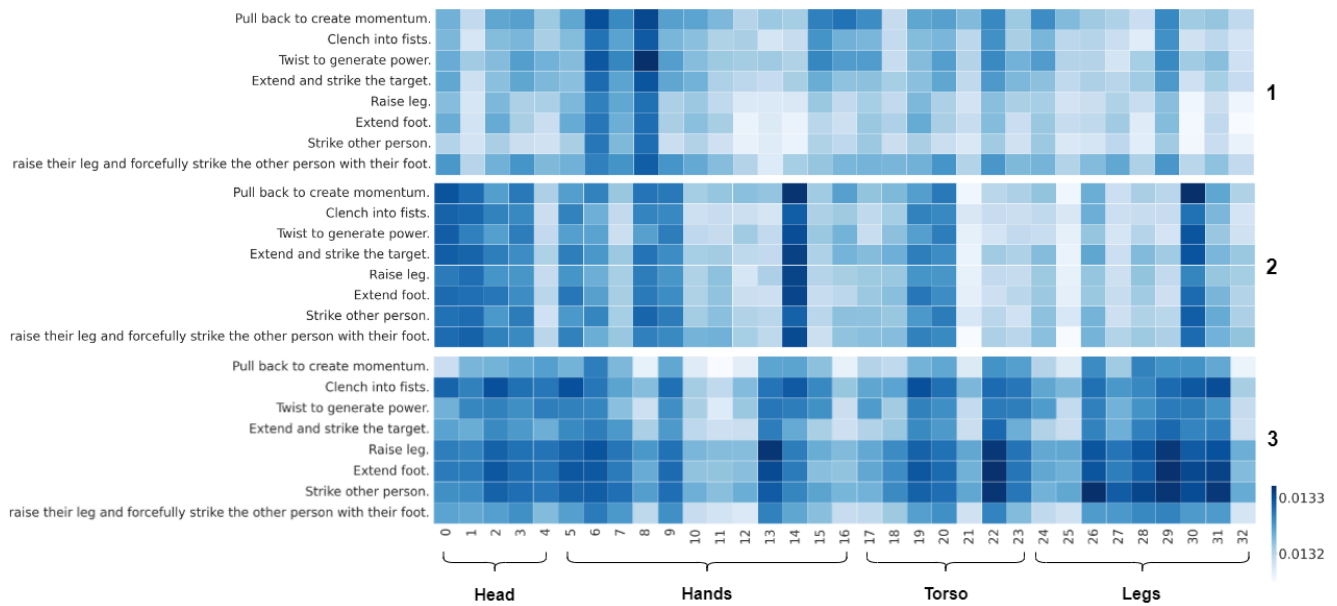


Figure 3. Visualized illustration of the learned adaptive weights for the class ‘kicking other person’.

4.2. Extra Visualization for Experiment Results

Fig. 5, Fig. 6 and Fig. 7 visualize the accuracy gaps and changing curves of every baseline and PURLS when predicting different numbers of unseen classes on *NTU-RGB+D 60*, *NTU-RGB+D 120*, *Kinetic-skeletons 200*. Our method reaches state-of-the-art prediction accuracies in every experimental setting. The performance conclusion on each dataset is consistent with the analysis in Sec. 4.3 of our main paper.

4.3. Full Ablation Results

Tab. 5 and Tab. 6 add extra results for the experiment setups we used for ablation study (see Sec. 4.4 of our main paper) on the *Kinetic-skeletons 200* dataset.

4.4. Other Experiment Results

Kinetic-skeleton 400: We provide Tab. 7 to record the performance results of ReViSE, DeVISE, Global, and PURLS on *Kinetics-skeleton 400* (See the dataset description in

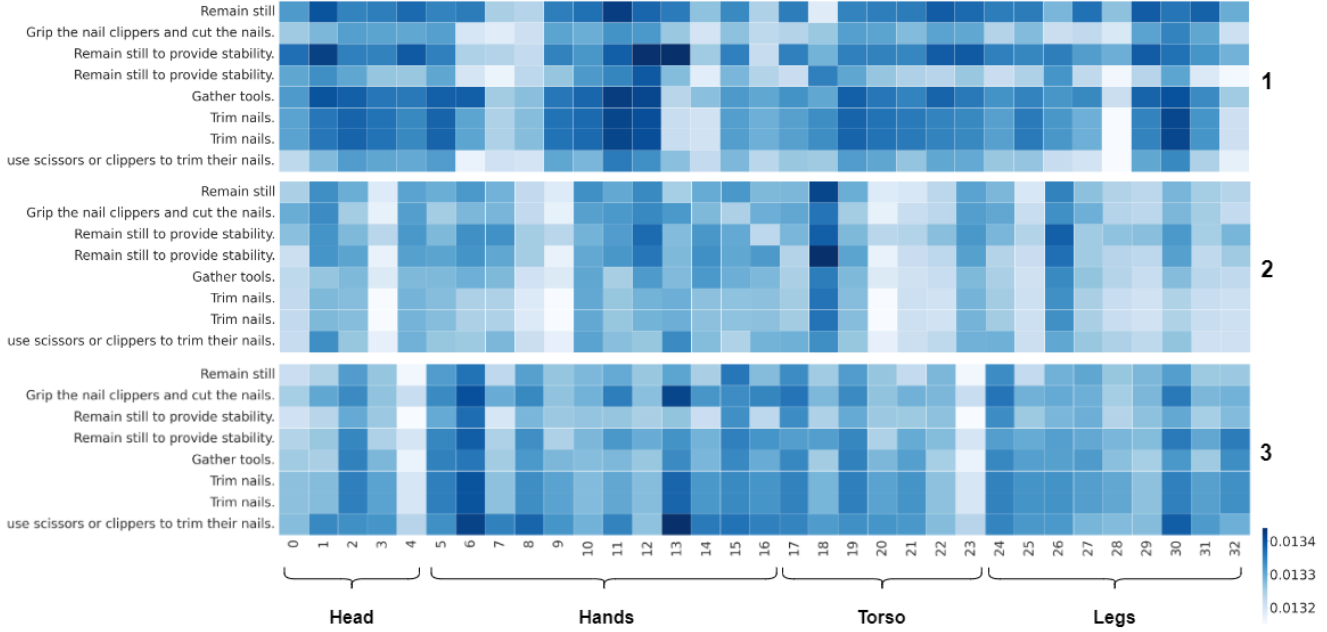


Figure 4. Visualized illustration of the learned adaptive weights for the class ‘cutting nails’.

Partitioning Strategy	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)				Kinetic 200 (Acc %)			
	55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60	180/20	160/40	140/60	120/80
Global (Original)	64.69	35.46	27.15	16.29	66.96	44.27	21.31	14.12	25.96	15.85	10.23	7.77
Global (GPT-3)	78.50	33.47	29.21	22.27	64.89	47.15	25.16	17.46	24.44	14.08	8.31	7.06
Static Partitioning	76.46	33.03	29.57	22.00	67.62	46.83	26.98	18.03	24.04	15.60	8.14	7.74
Adaptive Partitioning	79.23	36.77	31.05	23.52	71.95	52.01	28.38	19.63	32.22	22.56	12.01	11.75

Table 5. Full ablation study (%) on *NTU-RGB+D 60*, *NTU-RGB+D 120*, and *Kinetics-skeleton 200* for using different alignment learning with partitioning strategies, including direct global feature alignment to label or global description semantics, and PURLS with static/adaptive partitioning.

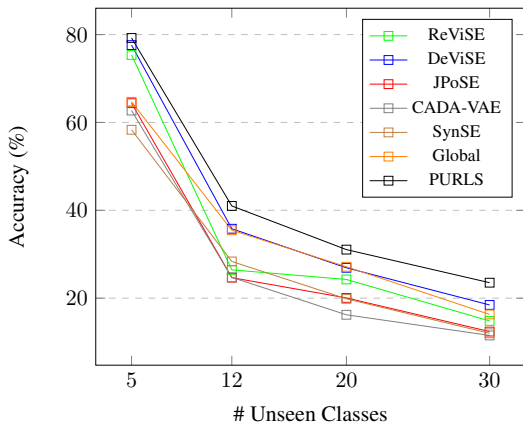


Figure 5. Visualized accuracy variation on *NTU-RGB+D 60*.

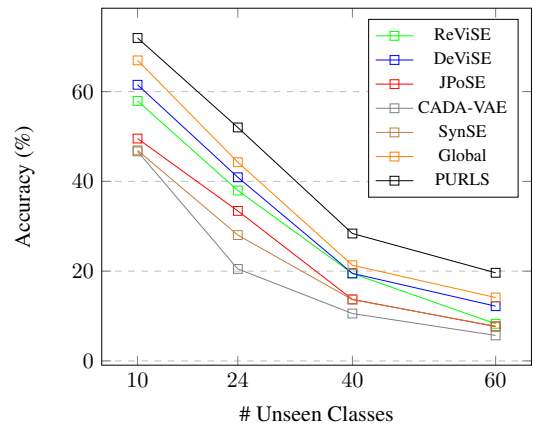


Figure 6. Visualized accuracy variation on *NTU-RGB+D 120*.

Sec. 4.1 of our main paper). Fig. 8 visualizes the accuracy gaps and changing curves of every baseline and PURLS.

We created four splits of 360/40, 320/80, 300/100, and 280/120 for evaluation. These setups are replaced with

α_i	BP	TI	NTU-RGBD 60 (Acc %)				NTU-RGBD 120 (Acc %)				Kinetic 200 (Acc %)			
			55/5	48/12	40/20	30/30	110/10	96/24	80/40	60/60	180/20	160/40	140/60	120/80
-			78.50	33.47	29.21	22.27	64.89	47.15	25.16	17.46	24.44	14.08	8.31	7.06
Average	✓		76.68	37.80	30.92	22.20	68.11	30.93	24.36	18.67	22.32	7.12	3.63	5.60
Learnable	✓		76.32	37.62	29.06	21.91	71.73	40.92	23.49	19.13	22.73	14.79	8.24	7.69
Average		✓	78.65	38.80	28.14	22.69	55.73	50.67	27.50	17.50	21.81	20.04	8.61	6.81
Learnable		✓	77.70	40.69	28.84	22.46	71.26	46.13	24.43	18.57	24.85	18.73	7.97	7.95
Average	✓	✓	79.02	39.92	31.00	23.47	73.55	51.38	27.67	18.66	26.87	21.10	10.03	11.55
Learnable	✓	✓	79.23	40.99	31.05	23.52	71.95	52.01	28.38	19.63	32.22	22.56	12.01	11.75

Table 6. Full ablation study (%) on *NTU-RGB+D 60*, *NTU-RGB+D 120* and *Kinetics-skeleton 200* for (1) using different α_i ($i \in [0, P + Z + 1]$) to sum for L_{train} , (2) adding body-part-based (BP) alignment learning, (3) adding temporal-interval-based (TI) alignment learning.

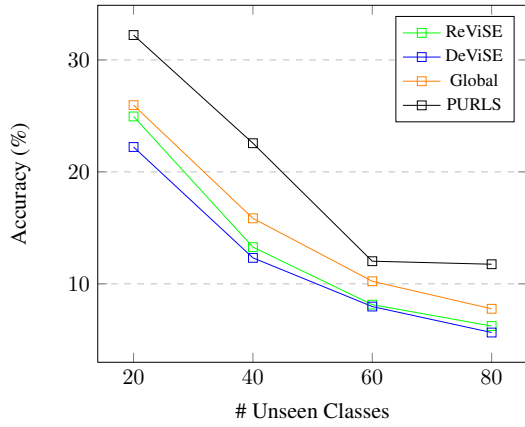


Figure 7. Visualized accuracy variation on *Kinetics-skeleton 200*.

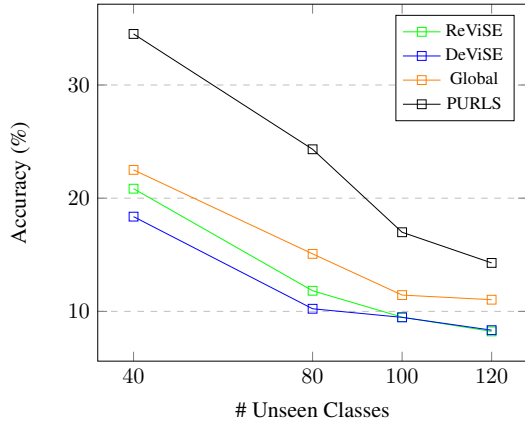


Figure 8. Visualized accuracy variation on *Kinetics-skeleton 400*.

a similar but simpler testing protocol on *Kinetics-skeleton 200*.

Small-scale datasets: Tab. 8 records the performance of ReViSE, DeVISE, Global, and PURLS on multiple small-scale datasets, including *NW-UCLA*, *UTD-MHAD*, and *UWA3D II* (respectively containing 10, 27 and 10 com-

Model	Kinetic 400 (Acc %)			
	360/40	320/80	300/100	280/120
ReViSE	20.84	11.82	9.49	8.23
DeViSE	18.37	10.23	9.47	8.34
Global	22.50	15.08	11.44	11.03
PURLS	34.51	24.32	16.99	14.28

Table 7. Zero-shot action recognition results (%) on *Kinetics-skeleton 400* for PURLS & all available baselines introduced in Sec. 4.2 of our main paper.

Model	NW-UCLA		UTD-MHAD		UWA3D II			
	8/2	5/5	22/5	18/9	14/13	24/6	20/10	15/15
ReViSE	69.12	44.99	29.37	19.86	12.26	30.33	10.91	10.43
DeViSE	72.81	36.02	12.50	13.59	13.94	34.16	12.51	11.41
Global	73.49	46.83	23.00	18.47	14.38	32.18	15.63	11.02
PURLS	75.84	49.47	57.50	31.71	19.23	35.65	18.91	13.98

Table 8. Zero-shot action recognition results (%) on *NW-UCLA*, *UTD-MHAD* and *UWA3D II* for PURLS & all available baselines.

mon daily action classes). As shown in the table, PURLS also outperforms other baselines in these testing environments where a minimal semantic overlap between seen and unseen classes makes zero-shot recognition more challenging.