MaskCLR: Attention-Guided Contrastive Learning for Robust Action Representation Learning

Mohamed Abdelfattah* Mariam Hassan Alexandre Alahi

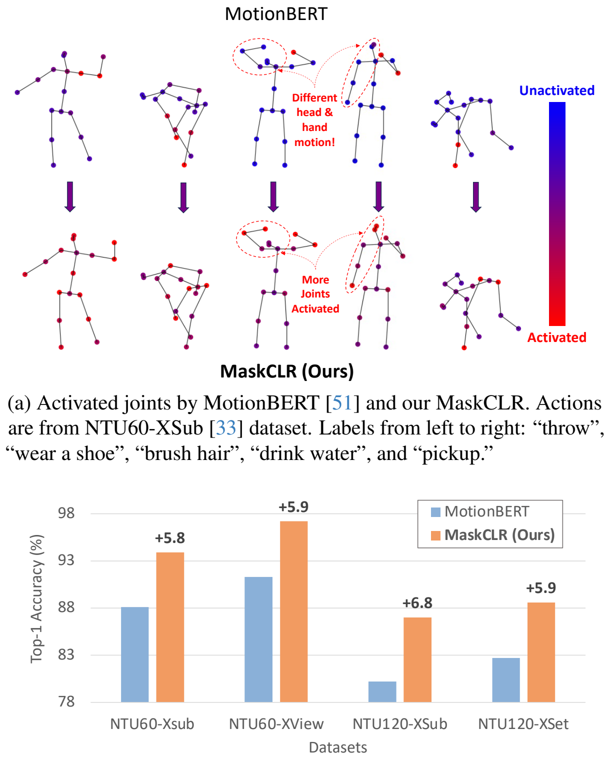
École Polytechnique Fédérale de Lausanne (EPFL)
firstname.lastname@epfl.ch

Abstract

*Current transformer-based skeletal action recognition models tend to focus on a limited set of joints and low-level motion patterns to predict action classes. This results in significant performance degradation under small skeleton perturbations or changing the pose estimator between training and testing. In this work, we introduce **MaskCLR**, a new **Mask**ed **C**ontrastive **L**earning approach for **R**obust skeletal action recognition. We propose an Attention-Guided Probabilistic Masking strategy to occlude the most important joints and encourage the model to explore a larger set of discriminative joints. Furthermore, we propose a Multi-Level Contrastive Learning paradigm to enforce the representations of standard and occluded skeletons to be class-discriminative, i.e., more compact within each class and more dispersed across different classes. Our approach helps the model capture the high-level action semantics instead of low-level joint variations, and can be conveniently incorporated into transformer-based models. Without loss of generality, we combine MaskCLR with three transformer backbones: the vanilla transformer, DSTFormer, and STTFormer. Extensive experiments on NTU60, NTU120, and Kinetics400 show that MaskCLR consistently outperforms previous state-of-the-art methods on standard and perturbed skeletons from different pose estimators, showing improved accuracy, generalization, and robustness. Project website: https://maskclr.github.io.*

1. Introduction

A skeleton is a representation of the human body structure that typically consists of a set of keypoints or joints, each associated with a specific body part. Compared to RGB-based action recognition, which focuses on extracting feature representations from RGB frames [1, 38, 41] and/or optical flow [36], skeleton-based approaches [44, 48] rely only on skeleton data. Transformers [39] have been proposed to encode the skeleton information for action recognition using Multi-Head Self Attention (MHSA) blocks [51]. Generally, MHSA blocks assign higher weights, *i.e.,* attention, to the most important joints/input regions that characterize every action to distinguish between different classes. For example, the hand joints in the action "throw" are assigned the highest weights while the rest of joints remain relatively unactivated. Motivated by this observation, we ask: *Is it possible to*



(a) Activated joints by MotionBERT [51] and our MaskCLR. Actions are from NTU60-XSub [33] dataset. Labels from left to right: "throw", "wear a shoe", "brush hair", "drink water", and "pickup."

(b) Accuracy under Gaussian noise $\mathcal{N}(0,\ 0.002^2)$.

Figure 1. Our proposed approach (MaskCLR) (a) uses a bigger set of discriminative joints to recognize actions and hence, (b) is more robust to noisy skeletons compared to baseline MotionBERT [51].

[*] Corresponding author.

exploit the information carried in the unactivated joints to aid in action classification?

To answer this question, in Figure 1a (top row), we visualize the activated joints of different samples from NTU60-XSub [33] dataset according to their attention weights, which are learned by State-Of-The-Art (SOTA) transformer-based MotionBERT [51]. From the visualizations, we observe that the model focuses on a limited set of discriminative joints to recognize the actions. Therefore, we argue that the model (1) misses action semantics (2) misclassifies the action if such joints are slightly perturbed and (3) ignores other joints which might be informative in action classification. For instance, in "brush hair" and "drink water," the unactivated joints carry useful information about head and hand motion, which is significantly different (see Figure 1a).

Consequently, such methods fall short in the following aspects: (1) *Robustness against skeleton perturbations*: the accuracy of existing methods is substantially affected by action-preserving levels of perturbations. For example, a small shift in joint coordinates often leads to a completely different prediction. (2) *Generalization to pose source*: Changing the pose estimator used to extract the skeletons between training and testing results in a considerable drop in accuracy. This shows that such methods only model the distribution of the predicted joints from the specific pose estimator used for training data extraction, but fail to handle any distribution shift from using a different pose estimator at test time.

In this paper, we introduce **MaskCLR**, a novel masked contrastive learning framework that improves the robustness, accuracy, and generalization of transformer-based methods. First, instead of using only a few joints, we propose an **Attention-Guided Probabilistic Masking (AGPM)** strategy to mostly occlude the activated joints and re-feed the resulting skeletons to the model. This strategy aims at *forcing* the model to explore a bigger set of informative joints out of the unactivated ones. Further, we propose a **Multi-Level Contrastive Learning (MLCL)** approach, which consists of two flavours of contrastive losses: sample- and class-level contrastive losses. At the sample level, we maximize the similarity between the embeddings of standard and masked skeletons in the feature space. At the class level, we take advantage of the cross-sequence global context by contrasting the class-wise average features of standard and masked skeletons, thus forming a class-discriminative feature space.

MaskCLR directly addresses the aforementioned limitations of existing methods. Our AGPM strategy helps the model learn the holistic motion patterns of multiple joints (Figure 1a), which mitigates the effect of action-preserving perturbations such as Gaussian noise (Figure 1b). Furthermore, our MLCL paradigm captures the high-level action semantics, enhancing the model robustness to distribution shifts arising from different pose estimators. To the best of our knowledge, MaskCLR is the first approach that improves the robustness and generalization of transformer-based skeletal action recognition. Notably, MaskCLR only requires an extra amount of training computation, but does not change the model size or inference time. To summarize, our key contributions in MaskCLR are threefold:

- First, we propose an Attention-Guided Probabilistic Masking strategy aimed at expanding the set of activated joints. Our objective is to recognize the combined joint motion patterns instead of focusing on a small set of joints.
- Next, we introduce a Multi-Level Contrastive Learning paradigm to leverage the rich semantic information in skeleton sequences sharing the same class. Our approach results in a better clustered feature space which boosts the overall model performance.
- Finally, we apply MaskCLR on three transformer backbones, and we demonstrate through extensive experimental results its superiority on three popular benchmarks (NTU60 [33], NTU120 [25], and Kinetics400 [16]).

2. Related Works

2.1. Skeleton-Based Action Recognition

The main objective behind skeleton-based action recognition is to classify a sequence of human keypoints into a set of action categories. Convolutional Neural Networks (CNNs) [4, 26] and Recurrent Neural Networks (RNNs) [7, 24] were among the earliest adopted deep-learning methods to model the spatiotemporal correlations in the skeletons but the performance was suboptimal because the topological structure of the skeletons was not well explored. Significant performance gains were obtained by employing Graph Neural Networks (GCNs) as a feature extractor on heuristically designed fixed graphs, which was first introduced in ST-GCN [48]. Since then, numerous methods have emerged to improve the accuracy and robustness of GCNs, including the usage of hierarchically decomposed graphs [19], Spatio-Temporal Curves [20], Koompan pooling [43], masked sequence reconstruction [45], and text-based action labels [10]. The SOTA on most benchmarks is PoseConv3D [9], which re-introduced 3D-CNNs for action recognition by projecting skeletons into stacked 3D Heatmaps. Transformers have also been adopted for action recognition. MotionBERT [51] performs 2D-to-3D pose lifting to learn motion representations. STTFormer [31] uses spatio-temporal tuples self-attention to capture the relationship of different joints. FG-STFormer [11] couples spatiotemporal focal and global transformers for action modelling. However, these methods (1) lack robustness against perturbed skeletons which are fairly common in real world applications, and (2) cannot handle the distribution shift in poses from a different pose estimator at test time. Additionally, transformer-based methods (3) give higher weights to a small set of joints without leveraging the information carried by the other joints, and (4) focus only on learning local graph representations but neglect the rich semantic information shared between skeletons of the same classes.

2.2. Contrastive Learning

The core idea behind contrastive learning is to pull together representations of similar inputs (positive pairs) while pushing apart that of dissimilar ones (negative pairs) in the feature space. It has been shown to contribute for substantial performance gains, especially in self-supervised representation learning [2, 14, 42]. Positive pairs are conventionally obtained by augmenting the standard input into two different views, while negative ones are obtained either through random sampling or hard mining techniques [15, 17, 32]. In skeleton-based action recognition, such frameworks have been adopted in the pre-training stage. In CrossCLR [21], the positive pairs are sampled in the data space by cross-modal knowledge. AimCLR [13] uses extreme augmentations to boost the effect of contrastive learning. ActCLR [22] uses the average motion across all sequences in the dataset as a static anchor for contrastive learning.

Our method differs from these approaches as follows: (1) The previous methods sample positive pairs by using fixed sample-wide augmentations that are invariant to the internal semantics of the action. In contrast, we employ an *adaptive masking strategy* by occluding the most activated joints, which vary based on the sample and action. (2) The previous methods employ contrastive learning at the sample level only. Instead, we contrast the semantic-level *class* representations, thus exploiting the context from the complementary individual and class aggregations. (3) While previous methods employ contrastive learning in the pre-training stage, MaskCLR is incorporated in the *fully-supervised* setting, thus requiring no extra pre-training cost.

3. Method

In this section, we introduce MaskCLR, our novel approach to enhance the accuracy, robustness, and generalization of transformer-based skeletal action recognition methods. MaskCLR consists of an Attention-Guided Probabilistic Masking (AGPM) strategy (Sec 3.2) combined with a Multi-Level Contrastive Learning (MLCL) approach (Sec 3.3 & Sec 3.4). As shown in Figure 2, our approach consists of two pathways: a standard pathway, which receives standard skeletons as input, and a masked one, which receives mostly the less activated joints from the standard pathway.

3.1. Preliminary

We leverage the Multi-Head Self-Attention (MHSA) backbone of transformer-based models [30, 51] to compute the joint-wise attention weights over the spatiotemporal dimensions. First, an input 2D skeleton sequence x of T frames and J joints is fed to a Fully Connected (FC) network to get the high-dimensional feature $\mathbf{F} \in \mathbb{R}^{T \times J \times C_f}$ of C_f channels. We then apply the transformer encoder g_θ for N times on $\mathbf{F}$ before passing the output to an FC network to get the feature representation $\mathbf{R} \in \mathbb{R}^{T \times J \times C_r}$ of C_r channels. Each MHSA block is composed of h heads defined as

$$head^i = softmax\left(\frac{\mathbf{Q}^i(\mathbf{K}^i)^t}{\sqrt{d_K}}\right)\mathbf{V}^i, \tag{1}$$

where $i \in 1, ..., h$ denotes the attention head. Self-attention is utilized to calculate the query $\mathbf{Q}$, key $\mathbf{K}$, and value $\mathbf{V}$ from input features $\mathbf{F}$, where d_K is the dimension of $\mathbf{K}$. We compute the mean of the attention maps across the self-attention heads to get the aggregated attention scores $\mathbf{A}$:

$$\mathbf{A} = \frac{1}{h}\sum_{i=1}^{h} softmax\left(\frac{\mathbf{Q}^i(\mathbf{K}^i)^t}{\sqrt{d_K}}\right). \tag{2}$$

Only the last MHSA block is used to compute the most activated joints since it inherits the information learned from the previous layers. In the supplementary material, we study the effect of using the attention filters from the other layers.

3.2. Attention-Guided Probabilistic Masking

The computed attention scores $\mathbf{A}$ (visualized in Figure 1a) serve as an empirical semantic richness prior to guide the masking strategy. Our aim is to mask the most activated joints to alleviate the dependency on them towards the exploration of a bigger set of informative joints (Figure 1a bottom). Therefore, we convert the attention scores into a probability distribution which reflects the probability that each joint feature is masked:

$$\pi = softmax(\mathbf{A}/\tau_{prop}), \tag{3}$$

where τ_{prop} is a temperature hyperparameter. Essentially, τ_{prop} controls the level of sharpness in the output probabilities. A lower temperature (less than 1) sharpens the distribution, making it more peaky and focusing on the most activated joints, and vice versa. Therefore, we set $\tau_{prop} < 1$ to direct the masking towards the most activated joints, and we adopt the Gumble max trick [12] to probabilistically guide the masking strategy:

$$\begin{gathered} \mathtt{K} = \delta \times T \times J, \\ r = -\log(-\log \varepsilon),\, \varepsilon \in U[0,1]^{T \times J}, \\ \mathtt{mask\_inds} = \mathtt{Top\text{-}K\text{-}indices}(\log \pi + r)\,, \end{gathered} \tag{4}$$

where $\delta \in [0,1]$ is a predefined fraction of joints to be masked and $U[0,1]$ is a uniform distribution. Hence, `mask_inds` is the indices of the joint features that are replaced with learnable mask tokens to get $\mathbf{F}_m$ in the masked pathway (see Figure 2). In this way, the more activated joints are more likely to be masked as illustrated in Figure 3, thus encouraging the model to explore more discriminative joints.

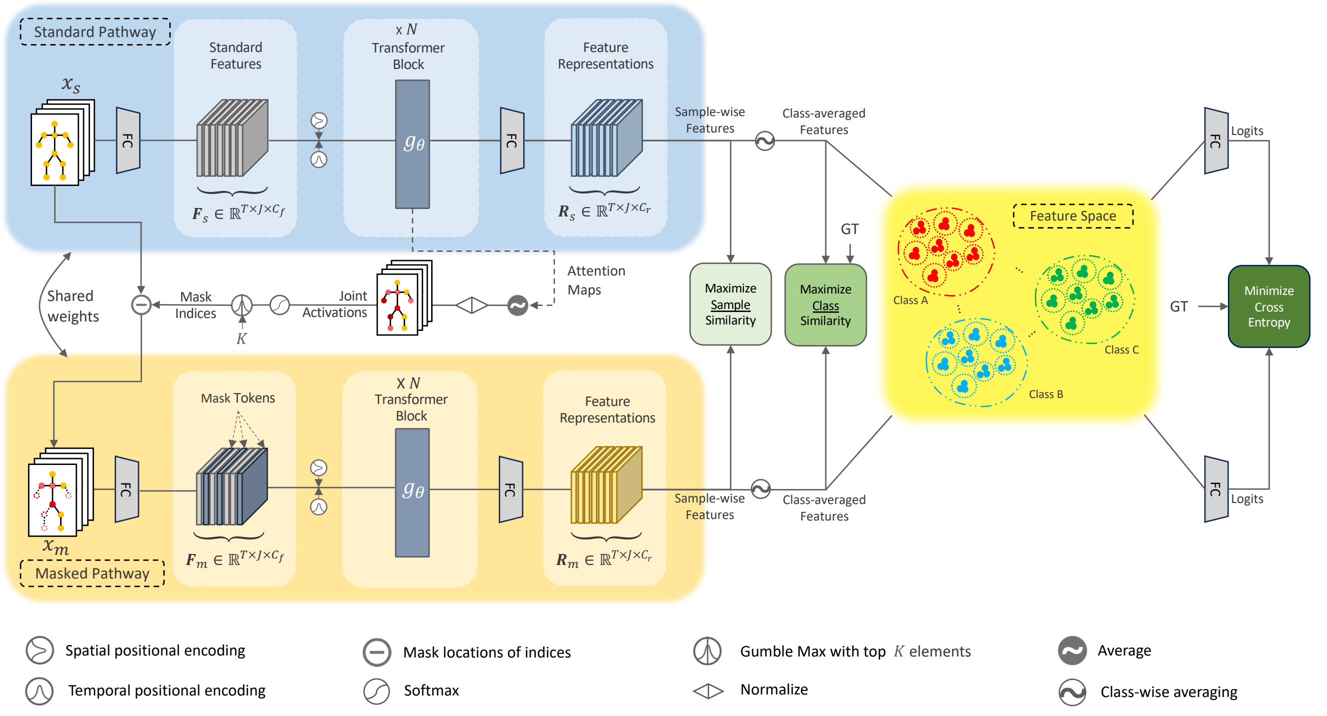


Figure 2. **Overview of MaskCLR.** Our approach consists of two (*standard* and *masked*) pathways that share the same weights. The standard pathway takes standard input skeletons while the masked pathway receives mostly the less activated joints from the standard pathway. Initially, the standard pathway is trained alone using the cross-entropy loss. The masked pathway, subsequently, comes into play to encourage the model to explore more discriminative joints. Using sample contrastive loss, we maximize the agreement of feature representations from the two pathways for the same skeleton sequence and vice versa. Additionally, to exploit the high semantic consistency between same-class skeleton sequences, we maximize the similarity between the class-wise average representations from the two pathways using class contrastive loss. Ultimately, the two contrastive losses contribute to the formation of a disentangled feature space, effectively improving the accuracy, robustness, and generalization of the model. At test time, only the standard pathway is used.

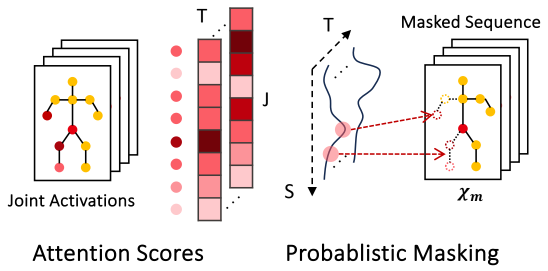


Figure 3. The attention scores from the standard pathway serve as an empirical semantic richness prior to guide the masking strategy. Hence, the most activated joints are more likely to be masked, enabling the model to explore more discriminative joints.

3.3. Sample Contrastive loss

We feed the more challenging masked features $\mathbf{F}_m$ to the masked pathway to get the masked representations $\mathbf{R}_m$. Our target is to achieve a model that can learn from the less activated joints, which carry useful information about the body pose that could inform the action prediction (see Figure 1a). To that end, we adopt sample contrastive loss to maximize the similarity between the standard $\mathbf{R}_s^i$ and masked $\mathbf{R}_m^i$ representations which correspond to the same skeleton sequence i. More specifically, for a batch of size B, the positive pairs are $\mathbf{R}_s^i$ and $\mathbf{R}_m^i$ while the negative ones are the rest of $B-1$ pairs, $\mathbf{R}_s^i$ and $\mathbf{R}_s^k$, $k \neq i$. Since skeletons extracted from different videos, forming the negative pairs, have different content, the similarity of their representations should be minimized in the latent space. Therefore, we apply the sample contrastive loss $\mathcal{L}_{sc}$:

$$\mathcal{L}_{sc}(\mathbf{R}_s^i,\mathbf{R}_m^i) = - \log \left[\frac{e^{s(\mathbf{R}_s^i,\mathbf{R}_m^i)}}{e^{s(\mathbf{R}_s^i,\mathbf{R}_m^i)} + \sum_{k=1}^{B} \mathbb{1}_{k\neq i} e^{s(\mathbf{R}_s^i,\mathbf{R}_s^k)}} \right], \quad (5)$$

where $\mathbb{1}$ is an indicator function that evaluates to 1 for skeletons corresponding to a different sample $k \neq i$, s is the cosine similarity $s(u,v) = \frac{u^T v}{\|u\|_2 \|v\|_2} \,/\, \tau_{sim}$, and τ_{sim} is the temperature hyperparameter.

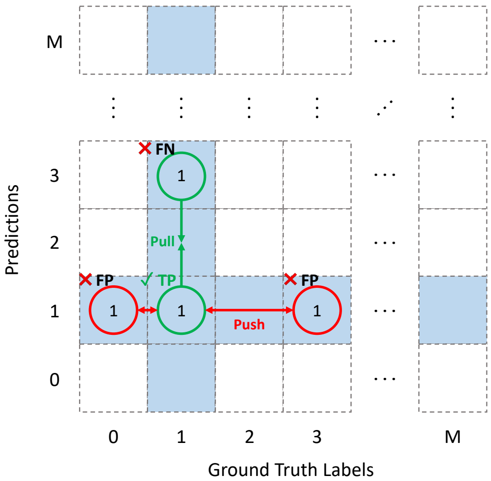


Figure 4. Class Contrastive Loss. Given an example label $l = 1$, we minimize the distance between representations of correctly predicted True Positive (TP) and misclassified False Negative (FN) samples sharing the same label l and maximize that between TP and other False Positive (FP) samples, which are misclassified as l.

3.4. Class Contrastive loss

Used alone, the sample contrastive loss function encourages representations of different skeleton sequences to be pushed apart even if they belong to the same action class. This, in turn, could result in overlooking the high-level semantics that characterize every action. Therefore, we need to maximize the similarity between skeleton sequences sharing the same class. Additionally, we aim at improving performance on confusing samples, *i.e.,* False Negatives (FN) and False Positives (FP), which often come from semantically similar actions like “reading” and “writing.” More specifically, as shown in Figure 4, our objective is to maximize the agreement between the representations of correctly predicted, *i.e.,* True Positive (TP), and mispredicted, *i.e.,* False Negative (FN), samples that share the same class while minimizing that between TP and FP ones, which belong to different classes. To that end, we define the class representation $\mathbf{C}^l$ of class l, as the Exponential Moving Average (EMA) of confident TP samples:

$$\mathbf{C}^l = \gamma \cdot \mathbf{C}^l + (1-\gamma) \cdot \frac{1}{|D^l_{s,TP}|} \sum_{i \in D^l_{s,TP}} \mathbf{R}^i_s \, , \tag{6}$$

where γ is the momentum term, $D^l_{s,TP}$ is the set of confident TP samples of class l from the standard pathway. We choose the confident TP samples to update the class representation because they have better semantic consistency. Ideally, in every batch, newly arrived samples of class l should be similar to their class representation $\mathbf{C}^l$ and dissimilar to the representations of other classes $\mathbf{C}^k$, $k \neq l$. Further, we want to take advantage of the confident TP samples to help clear the confusion in the FN and FP samples. Therefore, we average the confident and confusing samples as follows:

$$\mu^l_{TP} = \frac{1}{|D^l_{z,TP}|} \sum_{i \in D^l_{z,TP}} \mathbf{R}^i_z \, , \tag{7}$$

$$\mu^l_{FN} = \frac{1}{|D^l_{z,FN}|} \sum_{i \in D^l_{z,FN}} \mathbf{R}^i_z \, , \tag{8}$$

$$\mu^l_{FP} = \frac{1}{|D^l_{z,FP}|} \sum_{i \in D^l_{z,FP}} \mathbf{R}^i_z \, , \tag{9}$$

where μ^l_{TP}, μ^l_{FN} and μ^l_{FP} denote the mean representations of TP, FN, and FP respectively for label l, $D^l_{z,TP}, D^l_{z,FN}$, and $D^l_{z,FP}$ denote the sets of and TP, FN, and FP from the two pathways, $\mathbf{R}^i_z$ denotes the feature representations from any of the two pathways, and $z \in \{s, m\}$. We pull closer μ^l_{TP}, μ^l_{FN} and push away μ^l_{TP}, μ^l_{FP} (see Figure 4) by minimizing the penalty term:

$$\omega^l = s(\mu^l_{TP}, \mu^l_{FP}) - s(\mu^l_{TP}, \mu^l_{FN}) + 2. \tag{10}$$

Hence, ω^l hits its minimum 0 when the similarity between the TP and FP converges to -1 and that between TP and FN converges to 1. We add the constant term 2 to force ω^l to be non-negative value. We set $s(.,.)$ to 0 if any of its input samples are not available in the batch. The penalty term ω^l, inspired by SegFormer [40], contributes to reducing the ambiguity in the confusing samples and is, therefore, added to the class contrastive loss $\mathcal{L}_{cc}$ as follows:

$$\mathcal{L}_{cc}(\mathbf{C}^l, \mu^l_{TP}) = \\ -\log \left[\frac{e^{s(\mathbf{C}^l, \mu^l_{TP}) - \omega^l}}{e^{s(\mathbf{C}^l, \mu^l_{TP}) - \omega^l} + \sum_{k=1}^{M} \mathbb{1}_{k \neq l} e^{s(\mathbf{C}^k, \mu^l_{TP})}} \right] , \tag{11}$$

where M is the number of classes in the dataset. Intuitively, $\mathcal{L}_{cc}$ encourages feature representations to be class-dissociated, with ω giving more focus to push away representations of confusing classes. Finally, The overall loss function used to train our model is,

$$\mathcal{L} = \mathcal{L}_{ce} + \alpha \mathcal{L}_{sc} + \beta \mathcal{L}_{cc} \, , \tag{12}$$

where $\mathcal{L}_{ce}$ is the average cross entropy loss from the two pathways, and α and β are the weights assigned to sample and class contrastive losses respectively.

4. Experiments

4.1. Datasets

We use the **NTU RGB+D** [25, 33] and **Kinetics400** [16] datasets in our experiments. To obtain 2D poses, we employ three pose estimators (pre-trained on MS COCO [23]) of different AP scores: **ViTPose** (SOTA) [46] (High Quality, HQ), **HRNet** [37] (Medium Quality, MQ), and **OpenPifPaf** [18] (Low Quality, LQ). We apply the same post-processing across the three versions (outlier removal, pose tracking, etc). Further details about the datasets and pose estimators are provided in the supplementary material. We report the Top-1 accuracy for all datasets.

4.2. Implementation Details

We validate our approach on three transformer backbones: the vanilla transformer [6], DSTFormer [51], and STTFormer [31]. We set the depth $N = 5$, $C_f = 512$, $C_r = 512$, $h = 8$, and fix temporal sampling at $T = 243$. Different temporal lengths could be handled at test time due to the flexibility of the transformer backbone. For contrastive losses, we set $\alpha = 0.9$, $\beta = 0.1$, $\gamma = 0.9$, $\delta = 0.7$ and $\tau_{prop} = \tau_{sim} = 0.7$. The classification head is an MLP with hidden dimension = 2048, drop out rate $p = 0.5$, BatchNorm, and ReLU activation. We train our model for 600 epochs, where we first use only $\mathcal{L}_{ce}$ to train the standard pathway for 300 epochs. For the next 300 epochs, we add the masked pathway and train with the combined loss (Eq. 12). We train with backbone learning rate 0.0001, MLP learning rate 0.001, and batch size 32 using AdamW [28] optimizer. We conduct our experiments with 8 A100 GPUs.

4.3. Comparison with state-of-the-art Methods

Accuracy on standard skeletons. In Table 1, we compare the accuracy of MaskCLR to SOTA supervised methods, or self-supervised methods after fine-tuning. MaskCLR outperforms previous methods on **4 out of 5** benchmarks, and outperforms **all** baseline methods sharing the same backbone on **all** benchmarks. For NTU60-XView, MaskCLR is tied with MAMP [29] with the vanilla transformer backbone, yet improves the accuracy of the STTFormer and DSTFormer backbones. For Kinetics400, MaskCLR surpasses MotionBERT [51] by a margin of **5.8** percentage points. This shows that MaskCLR improves the accuracy at no pre-training cost and without increasing the model size.

Generalization to pose source. To demonstrate the generalization of our method to the type of pose estimator, we evaluate our model on the skeletons extracted by different pose estimators of different quality levels. For a fair comparison, we train all models on MQ skeletons and we evaluate on LQ and HQ ones (see Table 2). MaskCLR with DSTFormer backbone [51] consistently outperforms previous methods on **all** benchmarks under HQ and LQ poses. Compared to MotionBERT [51], using the same backbone, we improve generalization to skeleton source by **26.6** percentage points for NTU120-XSub VitPose Skeletons.

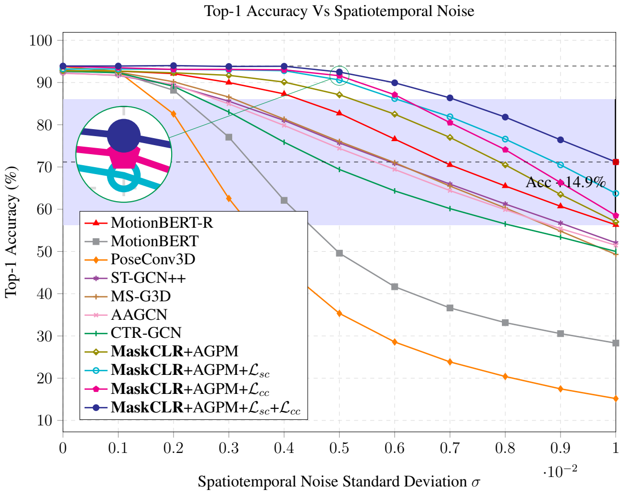


Figure 5. Top 1 accuracy on NTU60-XSub against spatiotemporal noise levels. While the performance of current methods drops rapidly with noise, adding each of AGPM, $\mathcal{L}_{sc}$, and $\mathcal{L}_{cc}$ significantly improves robustness. When combined, MaskCLR shows the lowest drop in accuracy compared to previous SOTA methods.

Robustness against skeleton perturbations. We compare the robustness of our method to existing methods under Gaussian noise, part occlusion, and joint occlusion. For noisy skeletons, we introduce Gaussian noise $X \sim \mathcal{N}(0, \sigma^2)$ to all joints, incrementally increasing σ from 0 to 0.01 with step size = 0.001. In the supplementary material, we provide visualizations of noisy skeletons. As shown in Figure 5, MaskCLR is superior to existing methods under noisy skeletons. Additionally, MaskCLR surpasses MotionBERT-R, which is trained with the same DSTFormer backbone [51] and 15% random masking, by **14.9** points at noise $\sigma = 0.01$. In part occlusion, we separately remove five body parts {head, left_arm, right_arm, left_leg, right_leg}. In joint occlusion, we randomly mask 15%, 30%, 45%, and 60% of joints. As shown in Figure 6, MaskCLR substantially outperforms existing methods under part and joint occlusion. We provide more results in the supplementary material.

Robustness against perturbed skeletons from different pose estimators. We perturb all LQ, MQ, and HQ skeletons with 50% frame masking and spatiotemporal Gaussian noise at $\sigma = 0.002$. While the performances of PoseConv3D [9] and MotionBERT [51] significantly degrade, MaskCLR shows the smallest drop in accuracy (see Table 3). Using the same DSTFormer backbone, MaskCLR achieves an absolute performance improvement of 9.1% on NTU60-XView ViTPose skeletons compared to MotionBERT. This shows the superiority of our approach in generalization and robustness under perturbed skeletons from different pose estimators.

Table 1. MaskCLR outperforms or closely competes with previous SOTA. Numbers in green reflect improvement over the second best method using the same backbone.

Method	Backbone	NTU60-XSub	NTU60-XView	NTU120-XSub	NTU120-XSet	Kinetics400
ST-GCN [48]	GCN	81.5	88.3	70.7	73.2	30.7
ActCLR [22]	GCN	85.8	91.2	79.4	80.9	-
ST-GCN++ [8]	GCN	89.3	95.6	83.2	85.6	-
AAGCN [35]	GCN	89.7	95.7	80.2	86.3	-
DGNN [34]	GCN	89.9	96.1	-	-	36.9
FR-GCN [50]	GCN	90.3	95.3	85.5	88.1	-
CTR-GCN [3]	GCN	90.6	96.9	82.2	84.5	-
MS-G3D [27]	GCN	92.2	96.6	87.2	89.0	45.1
FG-STFormer [11]	FG-STFormer	92.6	96.7	89.0	90.6	-
FGCN [49]	GCN	90.2	96.3	85.4	87.4	-
Koompan Pool. [43]	GCN	92.9	96.8	**90.0**	**91.3**	-
InfoGCN [5]	GCN	93.0	**97.1**	85.1	86.3	-
PoseConv3D [9]	3D-CNN	**93.7**	96.6	86.0	89.6	**46.0**
AimCLR [13]	STTFormer	83.9	90.4	74.6	77.2	-
CrosSCLR [21]	STTFormer	84.6	90.5	75.0	77.9	-
SkeletonMAE [47]	STTFormer	86.6	92.9	76.8	79.1	-
MaskCLR (Ours)	STTFormer	**90.1** (↑3.5)	**95.4** (↑2.5)	**79.0** (↑2.2)	**80.5** (↑1.4)	-
SkeletonMAE [47]	Transformer	88.5	94.7	87.0	88.9	-
MAMP [29]	Transformer	93.1	**97.5**	90.0	91.3	-
MaskCLR (Ours)	Transformer	**93.5** (↑0.4)	**97.5**	**90.5** (↑0.5)	**91.9** (↑0.6)	-
MotionBERT [51]	DSTFormer	92.8	97.1	84.8	86.4	38.8
MaskCLR (Ours)	DSTFormer	**93.9** (↑1.1)	**97.3** (↑ 0.2)	**87.4** (↑2.6)	**89.5** (↑3.1)	**44.7** (↑5.8)

Table 2. Top-1 accuracy when changing the pose estimator between training and testing. Numbers in green reflect improvement over MotionBERT [51] with the same DSTFormer backbone. Right arrows indicate changing the pose estimator from train → test.

Method	NTU60		NTU120	
	XSub	XView	XSub	XSet
HRNet [37] (MQ) → PifPaf [18] (LQ)				
ST-GCN++ [8]	65.3	72.4	68.8	71.2
MS-G3D [27]	51.7	57.3	54.9	55.8
AAGCN [35]	51.4	60.1	51.5	62.7
CTR-GCN [3]	50.4	58.2	57.6	58.2
PoseConv3D [9]	83.2	87.3	80.2	82.9
MotionBERT [51]	90.4	91.9	73.5	77.6
MaskCLR	**93.4** (↑ 3.0)	**97.2** (↑ 5.3)	**87.1** (↑ 13.6)	**86.5** (↑ 8.9)
HRNet [37] (MQ) → ViTPose [46] (HQ)				
ST-GCN++ [8]	73.2	83.0	66.4	69.6
MS-G3D [27]	82.5	91.0	72.1	74.5
AAGCN [35]	79.6	90.2	66.6	72.0
CTR-GCN [3]	77.7	84.4	65.5	67.7
PoseConv3D [9]	73.7	79.8	65.2	70.1
MotionBERT [51]	71.0	85.8	51.7	63.9
MaskCLR	**91.5** (↑ 20.5)	**96.3** (↑ 10.5)	**78.3** (↑ 26.6)	**79.8** (↑ 15.9)

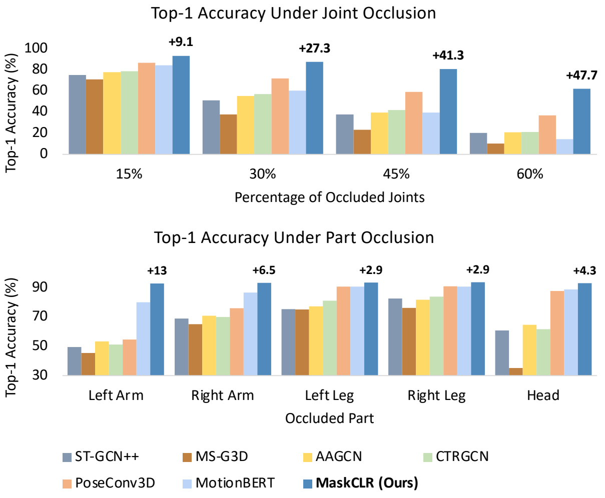


Figure 6. Top-1 accuracy under joint and part occlusion. **Bold** numbers reflect improvement over MotionBERT [51] with the same DSTFormer backbone.

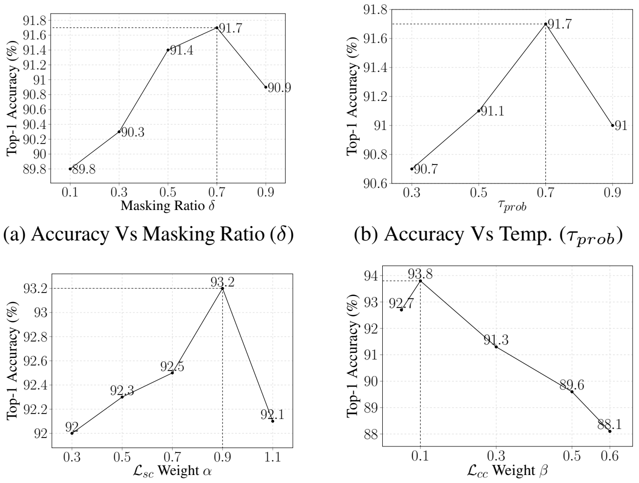
5. Ablation Studies

Next, we perform ablation experiments on NTU60-XSub dataset with DSTFormer [51] backbone to better understand the effect of the different components in our framework. Further ablations are provided in the supplementary material.

AGPM vs Random Masking (RM). We investigate the effectiveness of our Attention-Guided Probabilistic Masking (AGPM) strategy against random masking, which is commonly used in previous work [22, 51]. We randomly mask 15%, 30%, and 45% of joints as input to the masked pathway. As in previous findings [51], marginal differences are observed between different masking ratios $<$ 50% of the joints, with the highest accuracy being 89.7% at 15% RM (see Table 4). In comparison, we experiment with our AGPM approach by varying δ between 0.1-0.9 (step size = 0.2) (see Figure 7a). We observe that the highest accuracy of 91.7% is achieved at $\delta = 0.7$, which is **2.0** percentage points higher than the 15% RM. Higher masking ratio (≥ 0.5 RM or > 0.7 AGPM) result in rapid drop in accuracy (as in

Table 3. Drop in accuracy under skeleton perturbations. Numbers in green and red reflect improvement and decline compared to MotionBERT [51] with the same DSTFormer backbone, respectively.

Pose Estimator	Method	NTU60 XSub	NTU60 XView	NTU120 XSub	NTU120 XSet
Gaussian Noise ($\sigma = 0.002$)					
HRNet [37]	PoseConv3D	12.0	16.6	12.9	13.8
	MotionBERT	4.7	8.4	3.3	**0.1**
	MaskCLR	**0.1** (↑ 4.6)	**0.2** (↑ 8.2)	**0.2** (↑ 3.1)	1.1 (↓ 1.0)
PifPaf [18]	PoseConv3D	15.3	19.9	16.6	16.7
	MotionBERT	8.3	6.4	2.4	7.0
	MaskCLR	**0.1** (↑ 8.2)	**0.2** (↑ 6.2)	**0.1** (↑ 2.3)	**0.0** (↑ 7.0)
ViTPose [46]	PoseConv3D	9.2	13.7	2.8	10.9
	MotionBERT	2.9	9.4	1.9	5.0
	MaskCLR	**0.1** (↑ 2.8)	**0.4** (↑ 9.0)	**0.7** (↑ 0.2)	**0.1** (↑ 4.9)
50% Frame Masking					
HRNet [37]	PoseConv3D	1.9	3.1	6.2	8.4
	MotionBERT	1.7	1.6	1.2	4.4
	MaskCLR	**1.6** (↑ 0.1)	**1.0** (↑ 0.6)	**0.8** (↑ 0.4)	**4.3** (↑ 0.1)
PifPaf [18]	PoseConv3D	4.0	5.0	7.9	10.0
	MotionBERT	2.0	2.6	1.2	4.7
	MaskCLR	**1.6** (↑ 0.4)	**1.0** (↑ 1.6)	**1.1** (↑ 1.1)	**2.4** (↑ 2.3)
ViTPose [46]	PoseConv3D	2.2	2.7	9.0	7.9
	MotionBERT	**0.9**	10.2	3.1	2.6
	MaskCLR	1.1 (↓ 0.2)	**1.1** (↑ 9.1)	**2.3** (↑ 0.8)	**2.1** (↑ 0.5)



(a) Accuracy Vs Masking Ratio (δ)

(b) Accuracy Vs Temp. (τ_{prob})

(c) Accuracy Vs $\mathcal{L}_{sc}$ Weight (α)

(d) Accuracy Vs $\mathcal{L}_{cc}$ Weight (β)

Figure 7. Effect of hyperparameters on NTU60-XSub.

previous findings [29, 51]). One possible explanation is that when masking is too high, it results in very few cues in the skeletons sequence, leading the model to overfit. Furthermore, we study the effect of the temperature hyperparameter τ_{prop} on the model performance at $\delta = 0.7$ after 100 epochs. We keep $\tau_{prop} < 1$ to increase the probability of masking the most activated joints (as described in sec. 3.2). We find that $\tau_{prop} = 0.7$ results in the best accuracy of 91.7% (Figure 7b). Higher values result in similar behavior as RM since it alleviates the focus on masking the most activated joints.

Effect of Hyperparameters. We analyze the effect of $\mathcal{L}_{sc}$ weight α (Figure 7c) and $\mathcal{L}_{cc}$ weight β (Figure 7d) on the overall model performance. At $\delta = 0.7$, we experiment with adding $\mathcal{L}_{sc}$ and $\mathcal{L}_{cc}$ separately, achieving a top accuracy of 93.2% and 93.8% at $\alpha = 0.9$ and $\beta = 0.1$ respectively. Higher contrastive loss weights often lead to slower convergence or unstable training losses.

Table 4. Ablation Experiments on NTU60-XSub.

Masking	Mask ratio	$\mathcal{L}_{ce}$	$\mathcal{L}_{sc}$	$\mathcal{L}_{cc}$	Accuracy
No Mask	0	✓			88.7
RM	0.15	✓			**89.7**
RM	0.30	✓			89.6
RM	0.45	✓			89.4
RM	0.15	✓	✓		91.1
RM	0.15	✓		✓	91.9
RM	0.15	✓	✓	✓	**92.0**
AGPM	0.5	✓			91.4
AGPM	0.7	✓			**91.7**
AGPM	0.9	✓			90.9
AGPM	0.7	✓	✓		93.2
AGPM	0.7	✓		✓	93.8
AGPM	0.7	✓	✓	✓	**93.9**

Ablation on contrastive losses. We experiment with separately and collectively applying $\mathcal{L}_{sc}$ and $\mathcal{L}_{cc}$ with RM and AGPM (Table 4.) While each loss individually contributes to a performance gain, using the two losses together results in **2.3** and **2.2** improvement in percentage points with RM and AGPM respectively. Figure 5 shows that combining the two losses improves robustness against noise.

6. Conclusion

In this paper, we introduce MaskCLR, a new training paradigm for robust skeleton-based action recognition. Concretely, MaskCLR encodes more information from input joints through the attention-guided probabilistic masking of the most activated nodes. Further, a multi-level contrastive learning framework is proposed to contrast skeleton representations at the sample and class levels, forming a class-dissociated feature space that enhances the model accuracy, robustness to perturbations, and generalization to pose estimators. We demonstrate the effectiveness of our method on three transformer backbones, three benchmarks, and three pose estimators, significantly outperforming existing works on standard and perturbed skeletons.

7. Acknowledgement

This research is supported by the Swiss National Science Foundation through a Sinergia grant for the interdisciplinary project *Narratives from the Long Tail: Transforming Access to Audiovisual Archives*, grant number CRSII5_198632, see https://www.futurecinema.live/project/%7D for a project description. We deeply thank Taylor Mordan for his insightful feedback.

References

[1] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 1

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 3

[3] 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

[4] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3218–3226, 2015. 2

[5] Siyi Chi, Zihao Li, Yichen Zhang, Ziwei Liu, Le Zhang, and Chunhua Shen. Infogcn: Representation learning for human skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 7

[6] 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. 6

[7] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015. 2

[8] Haodong Duan, Jiaqi Wang, Kai Chen, and Dahua Lin. Pyskl: Towards good practices for skeleton action recognition. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 7351–7354, 2022. 7

[9] 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, 7

[10] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. Unified pose sequence modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13019–13030, 2023. 2

[11] Zhimin Gao, Peitao Wang, Pei Lv, Xiaoheng Jiang, Qidong Liu, Pichao Wang, Mingliang Xu, and Wanqing Li. Focal and global spatial-temporal transformer for skeleton-based action recognition. In *Proceedings of the Asian Conference on Computer Vision*, pages 382–398, 2022. 2, 7

[12] Emil Julius Gumbel. *Statistical theory of extreme values and some practical applications: a series of lectures*. US Government Printing Office, 1948. 3

[13] Tianyu Guo, Hong Liu, Zhan Chen, Mengyuan Liu, Tao Wang, and Runwei Ding. Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 762–770, 2022. 3, 7

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 3

[15] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *Advances in Neural Information Processing Systems*, 33:21798–21809, 2020. 3

[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. 2, 6

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020. 3

[18] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):13498–13511, 2021. 6, 7, 8

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

[20] Jungho Lee, Minhyeok Lee, Suhwan Cho, Sungmin Woo, Sungjun Jang, and Sangyoun Lee. Leveraging spatio-temporal dependency for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10255–10264, 2023. 2

[21] Linguo Li, Minsi Wang, Bingbing Ni, Hang Wang, Jiancheng Yang, and Wenjun Zhang. 3d human action representation learning via cross-view consistency pursuit. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4741–4750, 2021. 3, 7

[22] Lilang Lin, Jiahang Zhang, and Jiaying Liu. Actionlet-dependent contrastive learning for unsupervised skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3, 7

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 6

[24] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C Kot. Global context-aware attention lstm networks for 3d action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1647–1656, 2017. 2

[25] 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, 2019. 2, 6

[26] Mengyuan Liu, Hong Liu, and Chen Chen. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 68:346–362, 2017. 2

[27] 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. 7

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

[29] Yunyao Mao, Jiajun Deng, Wengang Zhou, Yao Fang, Wanli Ouyang, and Houqiang Li. Masked motion predictors are strong 3d action representation learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10181–10191, 2023. 6, 7, 8

[30] Chiara Plizzari, Marco Cannici, and Matteo Matteucci. Spatial temporal transformer network for skeleton-based action recognition. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part III*, pages 694–701. Springer, 2021. 3

[31] Helei Qiu, Biao Hou, Bo Ren, and Xiaohua Zhang. Spatio-temporal tuples transformer for skeleton-based action recognition. *arXiv preprint arXiv:2201.02849*, 2022. 2, 6

[32] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020. 3

[33] 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. 1, 2, 6

[34] 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. 7

[35] 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

[36] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014. 1

[37] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 6, 7, 8

[38] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1

[39] 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. 1

[40] Hualiang Wang, Huanpeng Chu, FU Siming, Zuozhu Liu, and Haoji Hu. Renovate yourself: Calibrating feature representation of misclassified pixels for semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2450–2458, 2022. 5

[41] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016. 1

[42] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7303–7313, 2021. 3

[43] Xinghan Wang, Xin Xu, and Yadong Mu. Neural koopman pooling: Control-inspired temporal dynamics encoding for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10597–10607, 2023. 2, 7

[44] Philippe Weinzaepfel and Grégory Rogez. Mimetics: Towards understanding human actions out of context. *International Journal of Computer Vision*, 129(5):1675–1690, 2021. 1

[45] Wenhan Wu, Jiaxin Wang, Le Zhang, and Chunhua Shen. Skeletonmae: Graph-based masked autoencoder for skeleton sequence pre-training. *arXiv preprint arXiv:2209.02399*, 2022. 2

[46] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 6, 7, 8

[47] Hong Yan, Yang Liu, Yushen Wei, Zhen Li, Guanbin Li, and Liang Lin. Skeletonmae: graph-based masked autoencoder for skeleton sequence pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5606–5618, 2023. 7

[48] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 1, 2, 7

[49] Hao Yang, Dan Yan, Li Zhang, Yunda Sun, Dong Li, and Stephen J Maybank. Feedback graph convolutional network for skeleton-based action recognition. *IEEE Transactions on Image Processing*, 31:164–175, 2021. 7

[50] Huanyu Zhou, Qingjie Liu, and Yunhong Wang. Learning discriminative representations for skeleton based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 7

[51] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 1, 2, 3, 6, 7, 8