



Article

Unsupervised Temporal Adaptation in Skeleton-Based Human Action Recognition

Haitao Tian  and Pierre Payeur * 

School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada; htian026@uottawa.ca

* Correspondence: ppayeur@uottawa.ca

Abstract: With deep learning approaches, the fundamental assumption of data availability can be severely compromised when a model trained on a source domain is transposed to a target application domain where data are unlabeled, making supervised fine-tuning mostly impossible. To overcome this limitation, the present work introduces an unsupervised temporal-domain adaptation framework for human action recognition from skeleton-based data that combines Contrastive Prototype Learning (CPL) and Temporal Adaptation Modeling (TAM), with the aim of transferring the knowledge learned from a source domain to an unlabeled target domain. The CPL strategy, inspired by recent success in contrastive learning applied to skeleton data, learns a compact temporal representation from the source domain, from which the TAM strategy leverages the capacity for self-training to adapt the representation to a target application domain using pseudo-labels. The research demonstrates that simultaneously solving CPL and TAM effectively enables the training of a generalizable human action recognition model that is adaptive to both domains and overcomes the requirement of a large volume of labeled skeleton data in the target domain. Experiments are conducted on multiple large-scale human action recognition datasets such as NTU RGB+D, PKU MMD, and Northwestern–UCLA to comprehensively evaluate the effectiveness of the proposed method.

Keywords: unsupervised domain adaptation; human action recognition; contrastive learning; human skeleton data



Citation: Tian, H.; Payeur, P.

Unsupervised Temporal Adaptation in Skeleton-Based Human Action Recognition. *Algorithms* **2024**, *17*, 581. <https://doi.org/10.3390/a17120581>

Academic Editors: Marcin Iwanowski and Bogusław Cyganek

Received: 31 October 2024

Revised: 6 December 2024

Accepted: 11 December 2024

Published: 16 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Skeleton-based human action recognition has been widely exploited for privacy-sensitive computer vision applications, such as human activity analysis [1], interactive gaming [2], and video surveillance [3]. Human actions can be represented by skeleton model videos [4] that encode the trajectories of skeleton joints captured by specialized acquisition systems, offering the merits of being agnostic to the environment and providing compact data representation. Graph Convolution Networks (GCNs) have dominated the recent research [5–7] on vision-based human action recognition by demonstrating their pivotal competence in aggregating action dynamics from spatial–temporal skeleton topologies and realizing end-to-end sequence-wise action classification.

Even though training a powerful GCN model for action recognition from large-scale public datasets [8,9] is promising, the transposition of the resulting model to applications in real-world environments remains challenging. For instance, the data distribution in a target deployment environment can deviate partially from that of the source training domain due to (i) the particular image acquisition configuration adopted or (ii) variability between subjects performing the same action. This consideration refers to the data domain covariate issue [10]. A common strategy consists of conducting fine-tuning rounds with full supervision using data collected in the target image acquisition configuration to reduce data discrepancy between the source domain and the target domain. However, the collection and annotation of a large volume of skeleton data for fine-tuning is extremely time consuming.

This work proposes an unsupervised domain adaptation (UDA) approach for skeleton-based human action recognition. It aims to utilize copious data from a controlled source domain, e.g., public datasets or laboratory environments, to pre-train an initial model. The approach then refines a target-adaptive action recognition model to deploy in a real-world (uncontrolled) domain with a fine-tuning phase using unlabeled data from the latter domain. UDA is achieved in two phases: Contrastive Prototype Learning (CPL) and Temporal Adaptation Modeling (TAM). Specifically, CPL utilizes supervised contrastive learning [11] to model discriminative action prototypes on trimmed data from the source domain. TAM formulates a self-training layer on the top of action prototypes. Since data in the target domain are unlabeled, TAM first leverages the pre-learned action prototypes to generate pseudo-labels on the target domain. Second, using both pseudo-labeled target data and labeled source data, it trains an action classification model that is adaptive to both domains. The contributions of this work consist of the following three elements:

- An original domain adaptation framework allows the action recognition network to initially learn human action knowledge from a labeled data domain, e.g., a public dataset or a controlled laboratory environment, to adapt to a target application domain;
- The proposed method requires only labeled data from the source domain and unlabeled data from the target domain for model adaptation, significantly reducing the effort invested in data annotation in the target domain;
- The proposed method is generalizable to state-of-the-art network architectures for unsupervised domain adaptation in action recognition without additional constraints.

2. Related Work

Skeleton data encode the dynamics of human movement over time by depicting the trajectories of the human body's skeleton joints. In this context, a human action can be represented by a sequence of skeleton frames that is trimmed to cover only the series of body movements that compose a specific action from the beginning to the end. By benchmarking the human action recognition task on a variety of large-scale skeleton-based datasets such as NTU RGB+D [8] and the PKU Multi-Modality Dataset [9], Graph Convolution Networks (GCNs) have attracted significant attention in recent works [5–7]. The concept of a GCN is to construct a graph upon skeleton frames in which each node corresponds to a human body joint and the edges correspond to the spatial connectivity among the joints, allowing the network to interpret the topological features of skeleton data. For instance, Li et al. [12] introduce actional-structural directed graph neural networks (AS-GCNs) to model the temporal dependencies among skeleton joints. Yan et al. [5] define spatial-temporal graph convolutional networks (ST-GCNs) to capture both spatial and temporal information in skeleton data, achieving competitive results in action recognition. Chen et al. [6] propose a framework (CTR-GCN) with the goal of interpreting relationships between joints.

However, the generalization of action recognition models in real environments is still challenging [13] due to the data distribution shift caused by variations in sensory configuration, e.g., camera views, heights, orientations, locations, and variations in data collection [8]. Skeleton contrastive learning [7,14] is an effective way to overcome the need for annotation on large volumes of data while demonstrating pivotal competency in learning invariant representations from unlabeled skeleton data for downstream tasks such as action recognition. These methods involve skeleton augmentation while constructing self-supervised contrastive learning pretext tasks, such as similarity measurement [15], which contrast positive pairs against negative pairs from a pre-defined dynamic dictionary [16,17].

3. Method

3.1. Preliminary

3.1.1. Skeleton-Based Action Recognition

A skeleton sequence, $X \in \mathbb{R}^{T \times V \times 3}$, consists of T frames in the shape of V human body joints, each one being defined in a calibrated camera reference frame with three-dimensional coordinates. A skeleton-based action recognition model F (Equation (1)) can be

decoupled into a feature subnetwork, $\mathcal{M} : \mathbb{R}^{T \times V \times 3} \rightarrow \mathbb{R}^{T \times V \times C_{fea}}$, which parses skeleton joint topologies from the input sequence X via spatial and temporal graph convolutions [5], and a task subnetwork, $\mathcal{C} : \mathbb{R}^{T \times V \times C_{fea}} \rightarrow [0, 1]^{L \times 1}$, which aggregates actions' semantic representations into a SoftMax output space where the multi-class cross-entropy loss is involved in optimization:

$$L_{CE} = -Y \cdot \log \left[\underbrace{\left(\mathcal{C} \circ \mathcal{M} \right)}_F (X) \right] \quad (1)$$

where Y is the sequence-wise action label of X , while $Y \in \mathbb{R}^L$ is defined in a range of L categories.

3.1.2. Domain Adaptation in Skeleton Data

However, end-to-end supervised learning on F requires the collection and annotation of a large volume of skeleton sequences $\{X, Y\}$, which is extremely time-consuming in practice. To circumvent the need for such data, this work proposes Unsupervised Domain Adaptation (UDA) based on two fundamental considerations:

- (1) The action recognition model, $F = \mathcal{C} \circ \mathcal{M}$, is modular such that its subnetworks (\mathcal{M} and \mathcal{C}) can be treated individually and separately, which allows an adaptation framework to optimize them step by step.
- (2) In practical engineering applications, a target deployment environment (target domain \mathcal{D}_t) often presents misaligned data distributions, given the fact that the image acquisition configuration can differ from that of the original data collection domain (source domain \mathcal{D}_s). At the same time, individuals coming from different environments performing the same action may have significant variations in their movement patterns, leading to another form of discrepancy in data distribution. Either data domain shift (i.e., the data distribution variations across \mathcal{D}_s and \mathcal{D}_t) tends to corrupt the model's performance when the model is well trained on \mathcal{D}_s but is evaluated on \mathcal{D}_t .

To this end, UDA considers a source domain, i.e., $\mathcal{D}_s = \{(X_s, y_s)\}$, which is composed of the sequence X_s and its annotation y_s , and a target domain, $\mathcal{D}_t = \{X_t\}$, where the skeleton sequence X_t is unlabeled. UDA aims to utilize a significant quantity of labeled data from \mathcal{D}_s to pretrain an intermediate model (e.g., feature encoder) and then refine a target-adaptive model F by using only unlabeled data from the target environment. By transferring skeleton action knowledge across different domains, UDA effectively avoids learning from scratch and saves the labor of annotating videos in the target domain.

Figure 1 illustrates the proposed framework. **Contrastive Prototype Learning (CPL)** optimizes the subnetwork \mathcal{M} over the labeled data from \mathcal{D}_s . The pre-learned \mathcal{M} is recycled into a successive **Temporal Adaptation Modeling (TAM)** phase that models temporal adaptation knowledge (pseudo-labels) on the unlabeled data from the target domain. Finally, the proposed **CPL** and **TAM** models are submitted to an optimization phase to progressively refine an action classifier.

3.2. Contrastive Prototype Learning (CPL)

In conventional skeleton-based human action recognition, a classification network receives a skeleton sequence X where its feature subnetwork (encoder), $\mathcal{M} : \mathbb{R}^{T \times V \times 3} \rightarrow \mathbb{R}^{T \times V \times C_{fea}}$, is responsible for parsing a spatiotemporal skeleton representation of X into a high-dimensional feature space where hidden states are closely clustered into *action prototypes* such that the task subnetwork (classifier), \mathcal{C} , is able to determine action-wise classification boundaries. Based on these observations, we propose to use Contrastive Prototype Learning (**CLP**) to learn *action prototypes* by optimizing \mathcal{M} using skeleton data from the source domain \mathcal{D}_s .

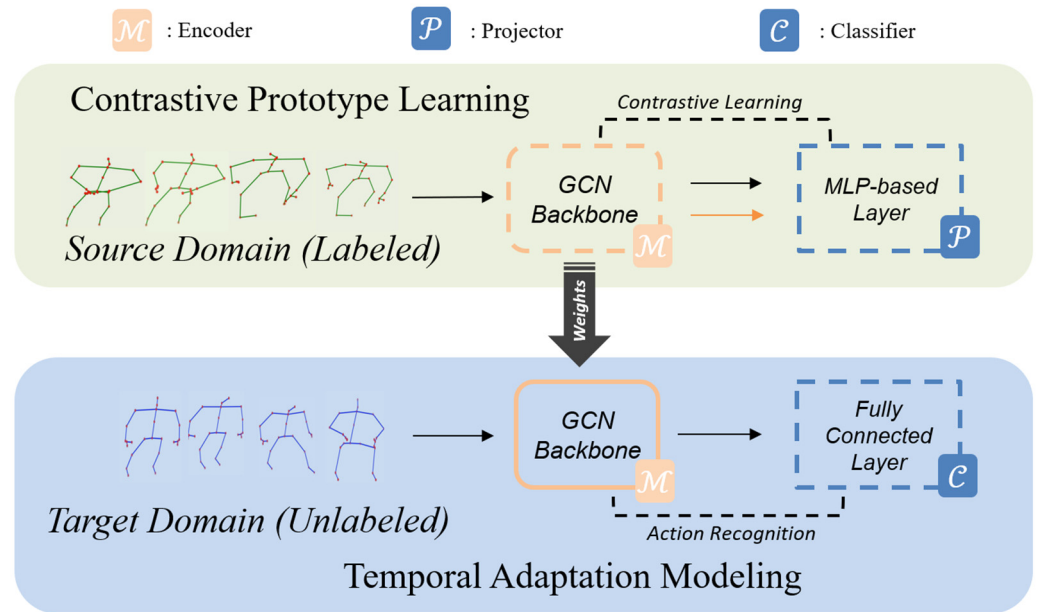


Figure 1. Unsupervised domain adaptation (UDA) framework combining the proposed CPL and TAM strategies. In CPL, the training data (labeled) from the source domain supports supervised representation learning. The learned backbone is reused in TAM for refining over data samples (initially unlabeled) from the target domain. Network embedding is denoted with dotted lines if it is updated during training and with solid lines otherwise.

Specifically, **CPL** utilizes a projection layer, $\mathcal{P} : \mathbb{R}^{T \times V \times C_{fea}} \rightarrow \mathbb{R}^{C_{prj}}$, on top of the feature subnetwork \mathcal{M} to implement contrastive learning. Recent research has utilized a variety of data augmentation schemes [7,14] to operate skeleton-data-based contrastive learning. In contrast, **CPL** opts out of data augmentation and leverages action labels in contrastive learning. Since data labels carry strong supervisory signals, **CPL** helps learn explicit and discriminative action prototypes.

Let $\mathcal{B}_S = \left\{ \left(X_s^{(i)}, y_s^{(i)} \right) \right\}_{i=1}^N$ be a data batch from the source domain \mathcal{D}_S , where $N \ll |\mathcal{D}_S|$. Let $Q = \{1, \dots, N\}$ be the set of instance indices of \mathcal{B}_S . The data batch considers a semantically identical set $P(i)$ with respect to the anchor $X_s^{(i)}$, in which each element $p \in P(i)$ indices a sequence $X_s^{(p)}$ which has the same sequence label as the anchor, i.e., $P(i) = \left\{ p \in A(i) : y^{(p)} = y^{(i)} \right\}$, where $A(i) \in Q \setminus \{i\}$ indicates all elements in Q except for i . We call $X_s^{(p)}$ a *positive* counterpart of the *anchor*. In this way, the representation pair $(z_s^{(i)}, z_s^{(p)})$ consists of a dynamic skeleton dictionary from which **CLP** can learn underlying feature invariances by $z^i = (\mathcal{P} \circ \mathcal{M})(X_s^{(i)})$ and $z^{(p)} = (\mathcal{P} \circ \mathcal{M})(X_s^{(p)})$. In order to map semantically similar sequences to action prototypes, **CLP** uses a SupCon loss [11] to minimize the distance between feature embeddings of *anchor* and *positives*:

$$L_{\text{SupCon}} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z^i \cdot z^{(p)} / \tau)}{\sum_{n \in \mathcal{N}} \exp(z^i \cdot n / \tau)} \quad (2)$$

where τ acts as the temperature parameter and n represents a negative counterpart sampled from a memory bank \mathcal{N} (we follow the same implementation as in [14] on the construction of the memory bank).

3.3. Temporal Adaptation Modeling (TAM)

Upon the optimization of effective action prototypes, a target domain sequence $X_t^{(i)}$ can be encoded as prototypical distributions according to $\mathcal{M}(X_t^{(i)})$. However, since the target domain is unlabeled, it is intractable to directly learn an action classifier \mathcal{C} using end-to-end supervised learning. At the same time, given the domain shift between the source domain and the target domain, directly learning a classifier exclusively on the source domain will lead to a biased classifier that compromises the testing performance on the target domain.

TAM proposes an episodic optimization paradigm that alternates a supervised learning iteration on the source domain and a self-training iteration on the target domain.

3.3.1. Supervised Learning on \mathcal{D}_s

It learns a classifier, $\mathcal{C} : \mathbb{R}^{T \times V \times C_{fea}} \rightarrow [0, 1]^{L \times 1}$, over the data sample $(X_s^{(i)}, Y_s^{(i)}) \in \mathcal{B}_s$. The learning process is optimized by a typical multi-class cross-entropy loss using

$$L_{CE-s} = -Y_s^{(i)} \cdot \log\left(\left(\mathcal{C} \circ \overline{\mathcal{M}}\right)\left(X_s^{(i)}\right)\right) \tag{3}$$

where $\overline{\mathcal{M}}$ denotes the action prototypes learned by Equation (2) which is fixed at this step.

3.3.2. Self-Training on \mathcal{D}_t

After training the model on the labeled source data, it is used to predict pseudo-labels for the unlabeled target domain data. Specifically, let $X_t^{(i)}$ be data sample from the target domain data batch $\mathcal{B}_t = \{X_t^{(i)}\}_{i=1}^N$ and $p_t^{(i,l)}$ be the probabilistic distribution of $X_t^{(i)}$ at category l where $p_t^{(i,l)} = \left(\left(\mathcal{C} \circ \mathcal{M}\right)\left(X_t^{(i)}\right)\right)^{(l)}$, we treat $\hat{Y}_t^{(i)}$ as the pseudo-label of $X_t^{(i)}$, which can be obtained by

$$\hat{Y}_t^{(i)} = \begin{cases} l, & \text{if } l = \arg \max_l p_t^{(i,l)}, \text{ and } p_t^{(i,l)} > \mathcal{T} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where \mathcal{T} denotes the confidence threshold. Only predictions with confidence scores above \mathcal{T} are considered valid pseudo-labels. This minimizes the inclusion of noisy labels. Therefore, we use $\hat{Y}_t^{(i)}$ in the self-training optimization:

$$L_{CE-t} = -\hat{Y}_t^{(i)} \cdot \log\left(\left(\mathcal{C} \circ \overline{\mathcal{M}}\right)\left(X_t^{(i)}\right)\right) \tag{5}$$

The pseudo-labels help the model learn to recognize target domain patterns, gradually improving its performance. Pseudo-labels are refined in each iteration. As the model becomes more accurate, it generates better pseudo-labels, leading to a refinement of the model's optimization.

3.4. Optimization

The proposed UDA network (illustrated in Figure 1) consists of three components: an encoder \mathcal{M} , a projector \mathcal{P} , and a classifier \mathcal{C} . The network training is carried out by the interaction among (2), (3), and (5), which are alternatively optimized (Figure 2) according to the stages below:

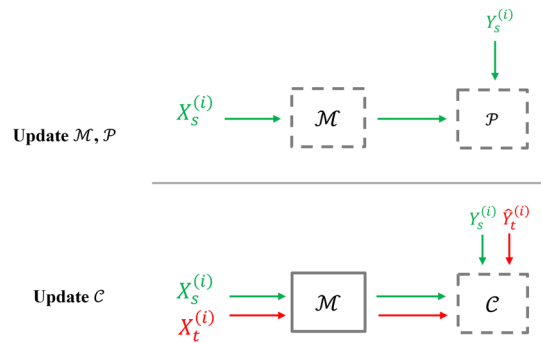


Figure 2. Tensor flow of the training pipeline. Green symbols relate to the source domain, while red ones relate to the target domain.

- **CPL (\mathcal{M}, \mathcal{P}) updating.** The CPL model is initially trained with supervised contrastive learning utilizing labeled source domain data $(X_s^{(i)}, Y_s^{(i)})$. Parameters in \mathcal{M} and \mathcal{P} are updated by minimizing the loss function (2) as follows:

$$\min_{\mathcal{M}, \mathcal{P}} \sum_{(X_s^{(i)}, Y_s^{(i)}) \in B_s} L_{\text{SupCon}}(X_s^{(i)}, Y_s^{(i)}; \mathcal{M}, \mathcal{P}) \tag{6}$$

- **TAM (\mathcal{C}) updating.** The classifier \mathcal{C} is updated in an iterative refinement scheme.
 - (1) Data and domain labels $(X_s^{(i)}, Y_s^{(i)})$ are used to update \mathcal{C} to improve semantic classification. It is achieved by minimizing the loss functions (3) as in (7), where λ is the trade-off weight used to balance the TAM training. At this step, $\overline{\mathcal{M}}$ is fixed.

$$\min_{\mathcal{C}} \left[(1 - \lambda) \sum_{(X_s^{(i)}, Y_s^{(i)}) \in B_s} L_{\text{CE-s}}(X_s^{(i)}, Y_s^{(i)}; \overline{\mathcal{M}}, \mathcal{C}) \right] \tag{7}$$

- (2) The pre-optimized \mathcal{C} is utilized to obtain the pseudo-label $\hat{Y}_t^{(i)}$ of $X_t^{(i)}$ based on Equations (4) and (5) as in Equation (8). Note that $\overline{\mathcal{C}}$ and $\overline{\mathcal{M}}$ are fixed at this step.

$$\max_{\hat{y}_t^{(i)}} L_{\text{CE-t}}(X_t^{(i)}, \hat{Y}_t^{(i)}; \overline{\mathcal{M}}, \overline{\mathcal{C}}) \tag{8}$$

- (3) Afterwards, data and pseudo-labels $(X_t^{(i)}, \hat{Y}_t^{(i)})$ are used to update \mathcal{C} .

$$\min_{\mathcal{C}} \left[\lambda \sum_{X_t^{(i)} \in B_t} L_{\text{CE-t}}(X_t^{(i)}, \hat{Y}_t^{(i)}; \overline{\mathcal{M}}, \mathcal{C}) \right] \tag{9}$$

Steps (1), (2), and (3) are implemented in each iteration of the TAM stage.

4. Experiments

Experimentation is conducted to evaluate the effectiveness of the proposed method in cross-domain human action recognition scenarios. Moreover, experimental ablation studies examine best practices related to the proposed framework, facilitating its reproduction.

4.1. Datasets

NTU RGB+D [8] is a popular large-scale skeleton-form human action recognition dataset. It presents 56,880 samples covering 60 human daily actions recorded in indoor scenes with three Microsoft Kinect V2 cameras mounted in different locations to support “cross-view” (C-View) variations. It also involves “cross-subject” (C-Sub) variations by

involving 40 actors in action performance. This dataset encoded skeleton representations over 25 joints.

PKU-MMD [9] is another popular public skeleton dataset presenting fewer data samples. For instance, PKU-MMD Part I has 21,545 class-specific sequences covering 51 human daily activities (41 single-person actions and 10 two-person actions), while PKU-MMD Part II contains 7000 class-specific sequences covering the 41 single-person actions. This dataset was also recorded with three Microsoft Kinect V2 devices, which provided skeleton representations encoded over 25 joints. Compared to NTU RGB+D, it involves significant camera view variations in the samples, which is used to mimic practical environments where data collection configurations can be inconsistent. The dataset also involves C-View and C-Sub variations.

Northwestern-UCLA [18] is a smaller 3D skeleton dataset composed of 1494 short sequences covering 10 human actions. This dataset was recorded with one Microsoft Kinect V1, which provided skeleton representations encoded over 20 joints. The discrepancy on the number of joints with respect to PKU-MMD acquisition configuration provides an effective evaluation scenario for domain adaptation. The dataset also involves C-View and C-Sub variations.

4.2. Implementations

As components of the proposed network structure, we adopt ST-GCN [5] as the feature encoder (\mathcal{M}) due to its wide adoption in recent research. Second, the projector \mathcal{P} is composed of a spatial pooling layer and a temporal pooling layer, followed by two linear layers equipped with a ReLU activation after the first linear layer. The classifier \mathcal{C} is composed of a spatial pooling layer and a temporal pooling layer, followed by a linear layer and a SoftMax activation.

The proposed framework in Figure 1 is deployed using PyTorch on a NVIDIA RTX 3090 GPU. For the **CPL** phase, we adopt the same hyperparameter values as in previous work [14], i.e., C_{prj} is set as 128, τ as 0.07, momentum value as 0.999. The size of \mathcal{N} is set as 32,768 for the source trimmed dataset NTU RGB+D and 16,384 for PKU MMDv1. The model is updated by using SGD with weight decay 0.0001 and learning rate 0.01. In the **TAM** phase, we use SGD to update the network with Nesterov momentum 0.9, weight decay 0.0005, and learning rate 0.05. We also use cosine annealing for the learning rate scheduler.

We use joint Normalization in data preprocessing. First, we establish a body frame system: (i) define the vector from the “spine” to the “left shoulder” as the x -axis; (ii) define the vector from the “spine” to the “spine base” as the y -axis; (iii) define the z -axis as the cross product of the x -axis and the y -axis. Second, we normalize the skeleton sequences by subtracting the trajectory of the “spine” joint.

4.3. Evaluation Protocols

First, multiple UDA tasks are considered to validate the performance of the proposed framework. In each UDA task, a pair of datasets, one labeled and the other unlabeled, are involved to represent the skeleton data distribution shift, i.e., the “C-View” evaluation protocol is used to represent the variation in image acquisition configuration and the “C-Sub” evaluation protocol is used to represent subject variations across domains.

- **NTU RGB+D to PKU-MMD.** We consider a single source domain, NTU RGB+D, and two target domains, PKU-MMD part I and PKU-MMD part II. In the first experimentation “NTU RGB+D to PKU-MMD part I” (Table 1), skeleton sequences under 50 actions common to both datasets are used for training and testing, while in “NTU RGB+D to PKU-MMD part II” (Table 2), 41 single-person actions are used for experiments.

Table 1. Experimental results on unsupervised domain adaptation (test case 1).

NTU RGB+D to PKU-MMD Part I	Source				Target			
	C-Sub		C-View		C-Sub		C-View	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Source only	69.07	41.83	72.86	44.59	57.32	27.40	59.02	29.09
UDA	70.05	45.09	72.45	46.34	69.34	37.35	72.17	36.12
Target only	34.41	18.67	35.30	17.36	85.06	49.21	87.34	52.18

Table 2. Experimental results on unsupervised domain adaptation (test case 2).

NTU RGB+D to PKU-MMD Part II	Source				Target			
	C-Sub		C-View		C-Sub		C-View	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Source only	75.34	38.46	80.98	39.21	23.58	10.06	22.46	9.5
UDA	74.15	36.53	80.02	38.50	30.55	12.42	31.29	14.53
Target only	35.23	12.35	32.73	11.45	45.91	24.68	47.58	25.39

- **NTU RGB+D to NW-UCLA.** This evaluation considers NTU RGB+D as the source domain and Northwestern–UCLA with 20-joint skeletons as the target domain (Table 3). The skeletons from the NTU RGB+D are preprocessed to remap from 25 to 20 joints. The datasets share seven common actions, i.e., “drop”, “pick up”, “throw”, “sitting down”, “standing up”, “wear jacket”, and “take off jacket”, that form a label space.

Table 3. Experimental results on unsupervised domain adaptation (test case 3).

NTU RGB+D to NW-UCLA	Source				Target			
	C-Sub		C-View		C-Sub		C-View	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Source only	88.39	49.92	93.03	65.28	65.36	34.65	69.42	37.61
UDA	87.74	43.80	90.57	64.63	72.69	42.19	76.44	40.53
Target only	44.31	23.11	47.49	30.12	89.24	43.23	92.06	65.28

- **(PKU-MMD) Part I to Part II.** We consider the trimmed PKU-MMD Part I as the source domain and the untrimmed PKU-MMD Part II as the target domain (Table 4). The skeleton sequences under 41 common single-person actions are used for training and testing.

Table 4. Experimental results on unsupervised domain adaptation (test case 4).

(PKU-MMD) Part I to Part II	Source				Target			
	C-Sub		C-View		C-Sub		C-View	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Source only	85.06	39.92	87.34	42.96	27.43	14.28	29.07	14.28
UDA	85.21	41.01	34.41	14.28	33.62	15.92	32.19	15.84
Target only	19.40	7.22	24.39	11.04	45.91	23.94	47.46	21.91

4.4. Results

Experimental results in terms of Top 1 accuracy (Acc) and F1 scores [8] are reported in Tables 1–4. First, we consider two baseline models, a source-only model and a target-only model, each of which is trained exclusively with supervision from a specific (source or target) domain by using the domain’s training set with annotations. Second, we trained our UDA model via the proposed **CPL** and **TAM** strategies.

The experimental results first demonstrate that either baseline model can yield convincing results in its original training domain (e.g., the source-only model in Table 1 trained on NTU RGB+D achieves 69.07% on the “C-Sub” testing set and 72.86% on the “C-View” testing set of NTU RGB+D, while the target-only model trained on PKU-MMD Part I achieves 85.06% and 87.34% on the testing sets of PKU-MMD Part I). However, given the domain shift between the source and the target domains, either baseline model fails when performing cross-domain action recognition (e.g., in Table 1 the source-only model (trained on NTU RGB+D) only achieves 57.32% on PKU-MMD Part I and similar results are observed for the target-only model when tested on NTU RGB+D data).

When the proposed UDA framework is used to learn a joint-domain classifier using the (labeled) training set of the source domain and the (unlabeled) training set of the target domain, the results in the second row of Tables 1–4 demonstrate that the resulting UDA model achieves significant improvement on cross-domain action recognition. It achieves a gain of 12.02% in Acc (from 57.32% to 69.34%) and 9.95% in F1 (from 27.40% to 37.35%) over the source-only model performance in Table 1. The UDA model still achieves comparable performance on the source domain compared to the source-only model in most of testing cases (e.g., 69.07% vs. 70.05% under C-sub in Table 1), demonstrating its effectiveness in learning action knowledge from two domains. However, given that target labels are absent, the UDA model expectedly underperforms the fully supervised (i.e., target-only) model. Similar performance is observed for every one of the four cross-domain test scenarios considered. Furthermore, even though the proposed UDA model obtains effective adaptation outcome under both the “C-View” and “C-Sub” evaluation protocols, it performs better in the “C-View” case, which we attribute to the different degree of skeleton data distribution shift.

4.5. Effects of Contrastive Prototype Learning

4.5.1. Training with Fewer Target Domain Samples

The critical contribution of Contrastive Prototype Learning (CPL) is the optimization of discriminative action prototypes via supervised contrastive learning using the source domain data. It is relevant to validate the transferability of the action prototypes that are learned in the source domain while only involving a subset of the target domain data for refinement. For this experiment, we first use CPL to learn an encoder \mathcal{M} using the labeled data from NTU-RGB+D. Second, we fix the pre-trained \mathcal{M} and use different proportions (varying from 5% to 100%) of labeled data from PKU MMD Part I to fine-tune a classifier \mathcal{C} . Note that this differs from the experiments reported in Section 4.4 that use only unlabeled data for model fine-tuning. Experimental results in Table 5 demonstrate that the model performance tends to increase monotonically with the ratio of target data samples involved but only until a saturation point. Interestingly, even using a very small number (e.g., 5%) of data samples from the target domain, the model still achieves compelling performance compared to the best model (85.06% in Table 1) when using 100% of the PKU-MMD Part I dataset.

Table 5. Performance (Top 1 accuracy) for different proportions of the target data samples in classifier C refinement training in contrastive prototype learning.

Percentage of Data Use	5%	10%	30%	50%	70%	100%
CPL	69.47%	72.65%	80.11%	83.86%	84.34%	85.06%

4.5.2. Comparison to State-of-the-Art Methods

The proposed method offers the advantage of remaining free of data augmentation for contrastive learning, unlike recently introduced approaches relying on a combination of skeleton augmentations. Therefore, we also include a comparison with similar studies. From the experimental results reported in Table 6, CPL demonstrates better Top1 accuracy

than the methods described in [19–21], revealing a promising adaptation performance and effectiveness in learning from skeleton data.

Table 6. Comparison to state-of-the-art works.

NTU RGB+D to PKU-MMD Part II	Top1 Acc
Zheng et al. [19]	44.8%
Lin et al. [20]	45.8%
Thoker et al. [21]	45.9%
CPL (proposed)	47.2%

4.5.3. Comparison to Vanilla Supervised Learning

An intuitive way to reach the same goal as **CPL** is to use the vanilla supervised pretraining paradigm. For instance, one can use a cross-entropy loss to train a regular action recognition model, $F = \mathcal{C} \circ \mathcal{M}$, using the labeled data from the source domain (Equation (1)), and then use the trained model \mathcal{M} for transfer learning on the target domain (Equation (9)). This subsection presents two experiments to evaluate whether utilizing vanilla supervised learning can reach the same effectiveness as the proposed **CPL** strategy. First, we use conventional supervised method [5] and the proposed **CPL** method to train a comparative model F with full supervision from NTU RGB+D. Second, we fine-tuned \mathcal{C} (with \mathcal{M} fixed) using full supervision on 10% data samples from PKU-MMD Part I. Afterwards, we test the refined model on the test set of PKU-MMD Part I. The experimental results in Table 7 demonstrate that the fully supervised learning method offers limited transferability compared to the proposed **CPL** method. The fully supervised transfer learning only achieves 45.97% on the target domain, representing a 26.68% gap in accuracy compared to using **CPL**.

Table 7. Performance of “vanilla” fully supervised learning vs. semi-supervised learning. “NTU” stands for “NTU RGB+D”, and “PKU” stands for “PKU MMD Part I”.

	Full Supervision	CPL
	NTU and 10% PKU (Fine-Tuning)	NTU and 10% PKU (Fine-Tuning)
Accuracy	45.97%	72.65%

4.5.4. Visualization of Action Prototypes

For better understanding of the effectiveness of **CPL** in learning action prototypes, closer examination of the embedding features from the two domains is studied using t-SNE [22]. Figure 3 illustrates the action clusters of the two respective domains (NTU RGB+D and PKU-MMD) encoded on the last convolutional layer of \mathcal{M} as interpreted by the two action recognition models (“vanilla” supervised pretrained on the upper row and “**CPL**” on the bottom row). First, even though the “vanilla” supervised pretrained model presents well-separated action clusters when tested on the source domain (upper left feature map), which reflects the model’s ability to interpret feature representations from the source domain, it presents less separable action clusters when tested on the target domain (upper right map) due to the impacts of domain shift. Such a discrepancy on action cluster interpretation leads to performance degradation across the two domains (from 69.07% to 57.32% in Table 1). Second, when applying the proposed **CPL** method, the model (bottom row) demonstrates more effective adaptation to the target domain whose action clusters are improved in terms of separability. Such improvement is observable on both test cases considering the source domain (bottom left map) and the target domain (bottom right map).

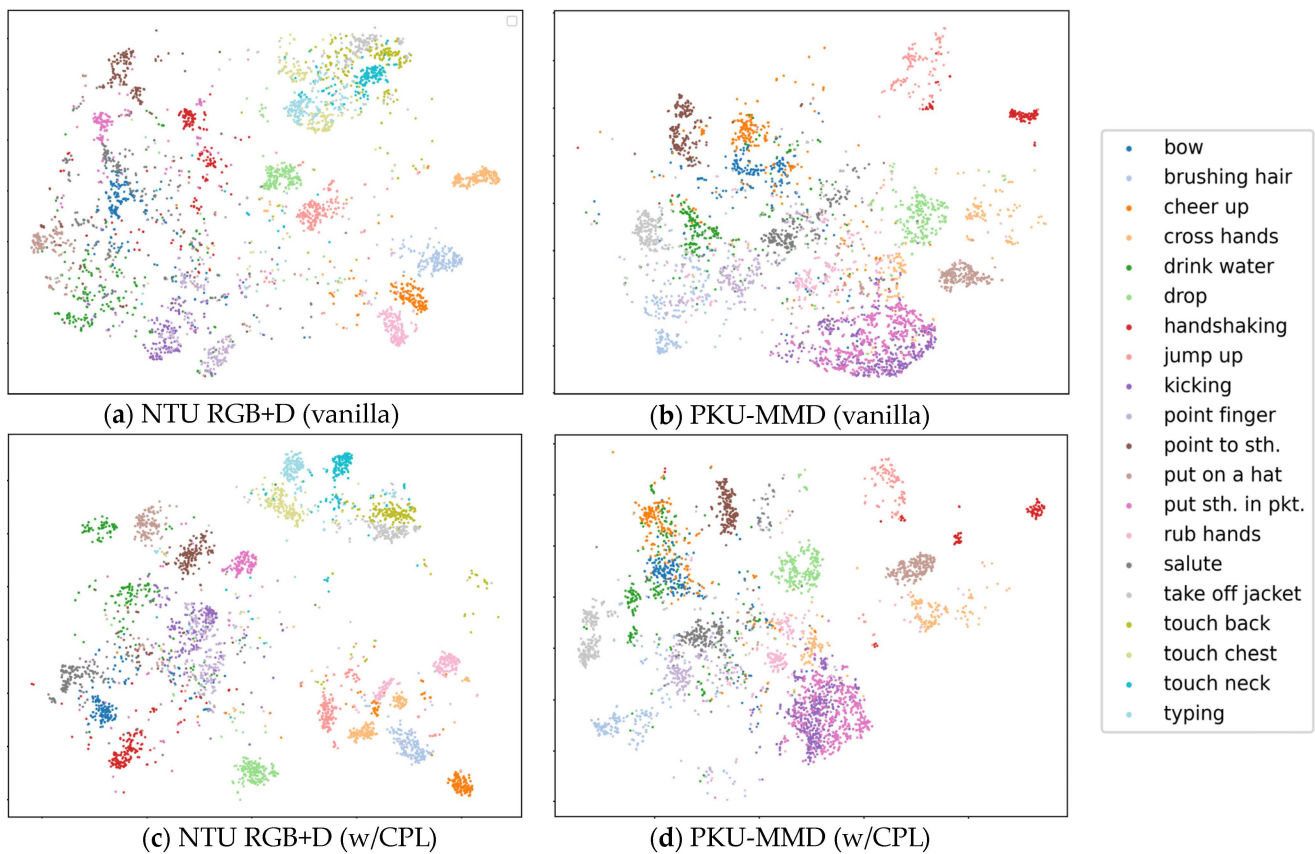


Figure 3. T-SNE visualization on action prototypes on the feature space of ST-GCN (upper row: “vanilla without CPL”; bottom row: “with CPL”). Action prototypes are distinguished by colors where twenty actions are randomly selected among fifty common actions for clarity. The left column represents action clusters of the source domain (NTU RGB+D), and the right column shows clusters of the target domain (PKU-MMD), respectively.

4.6. Ablation Study

We conduct ablation experiments to better understand the effectiveness of each component in the proposed framework.

4.6.1. Pseudo-Labels with TAM

In this work, **Temporal Adaptation Modeling (TAM)** is proposed to assist with integrating unlabeled target domain data into model training, allowing the joint-domain classifier to receive action knowledge from the target domain. To validate its effectiveness, we ablate the term L_{CE-t} proposed in Equation (9) (i.e., $\lambda = 0$ in Equation (9)) and use the exclusive source data in model training using Equation (7). Experimental results presented in Table 8 demonstrate a significant performance discrepancy on the two domains. Since only the source domain data is involved in model training, the resulting model is biased towards the source domain and achieves compromised performance in the target domain. The results demonstrate that **TAM** overcomes the limitation by using pseudo-labels and leads to effective adaptation to the target domain.

Table 8. The effectiveness of using pseudo-labels with TAM.

Models	Source	Target	Top1 Acc	Standard Deviation
without TAM	NTU RGB+D	-	59.35%	2.12%
with TAM	NTU RGB+D	PKU-MMD Part I	69.34%	1.70%

4.6.2. Hyperparameter Analysis for \mathcal{T}

In Equation (4), the parameter \mathcal{T} acts as a threshold to distill reliable pseudo-labels for TAM. Setting a high value of \mathcal{T} will seclude too many target data samples from model training and thus the source domain data will be dominant, giving rise to a biased training process. On the other hand, a low value of \mathcal{T} may cause the inclusion of too many noisy labels from the target domain, thus corrupting the training process. To identify an appropriate value for \mathcal{T} , we alter \mathcal{T} over the range [0.5, 1] in the TAM process while preserving other implementations parameters as defined in Section 4.2. As illustrated in Figure 4, the adaptation performance is affected as \mathcal{T} varies. The best performance is achieved when \mathcal{T} is around 0.85. Ultimately, when $\mathcal{T} = 1$, the model degrades to a source-only model (i.e., the “without TAM” model in Table 8).

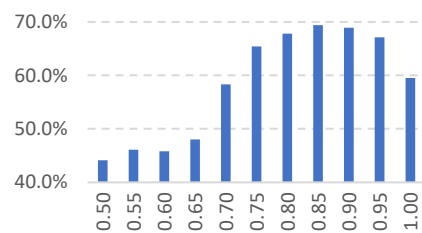


Figure 4. Performance (Top1 Acc) of TAM with varying hyperparameter \mathcal{T} .

4.6.3. Hyperparameter Analysis for λ

The implementation of TAM involves the determination of pseudo-labels on samples from the target domain and uses the parameter λ to weigh the importance of L_{CE-t} in Equation (9). Experimental investigation analyzes the selection of λ by conducting an experiment where the framework learns a classifier \mathcal{C} with the same implementation detailed in Section 4.2 but altering λ over the range [0.1, 1]. As illustrated in Figure 5, the performance is clearly affected as λ varies with an eventual degradation in performance when the hyperparameter places too much weight on L_{CE-t} . Consequently, it is observed that the best performance is achieved when λ remains relatively low, that is, around 0.3.

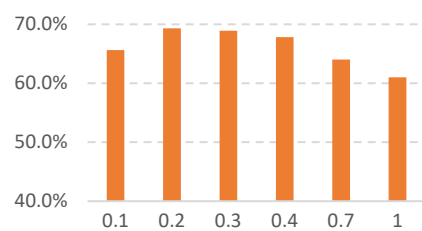


Figure 5. Performance (Top1 Acc) of TAM with varying hyperparameter λ .

5. Discussions

5.1. Future Work

As experimental results demonstrate in Tables 2 and 4, the model achieves limited adaptation results on the dataset PKU-MMD Part II. The latter presents significant domain shift compared to the source domain, NTU-RGB+D or PKU-MMD Part I. We conjecture that if the source and target domains are significantly different, initial pseudo-labels may be noisy, impacting the early training stages of TAM, thus leading to low accuracies in the target domain. In future works, it will be relevant to incorporate more efficient modules to tackle the challenge. We plan to use class-specific thresholds in pseudo-labeling. Specifically, in Equation (4), instead of using a hard threshold, \mathcal{T} , for all classes, we would rather use a class-specific one, \mathcal{T}_k , where we use the class-based statistic, e.g., sample frequency, to weight the confidence threshold for the action k . Such a strategy may help the model pay class-specific attention to optimization.

5.2. Real-World Application

One potential real-world application of the proposed method is to monitor patient movements in hospitals or elderly care institutions to detect abnormal activities. In such an environment, data distribution shifts can originate from the variability in data collection resulting from changing camera views, varying room configurations, and different subjects. The proposed method may help develop a robust action detection or recognition system without leveraging large amounts of data and labels, providing solutions for transfer learning in a variety of real-world environments.

5.3. Limitations

One limitation of the proposed approach is that the accuracy of pseudo-labels directly affects model performance. Low-quality pseudo-labels can propagate errors and degrade the optimization process. Techniques such as confidence-based filtering, adaptive thresholds, and ensemble predictions could be helpful to attenuate the impact of noisy labels.

6. Conclusions

In this paper, an original unsupervised domain adaptation (UDA) method is introduced to effectively adapt skeleton-based human action recognition to variable target environments without requiring labeled data from the latter. The proposed framework first leverages the benefits of contrastive prototype learning (CPL) to extract expressive action prototypes from a labeled source domain. Second, temporal adaptation modeling (TAM) refines a functional action recognition model to the target domain by associating and exploiting pseudo-labels. Experiments demonstrate the effectiveness of the proposed UDA strategy in comparison with fully supervised learning that requires voluminous labeled data from both the source and target domains. It also demonstrates the effectiveness of the proposed approach while bypassing the need for data augmentation in contrastive learning. Ablation studies suggest that the proposed CPL and TAM methods jointly contribute to the effectiveness of the overall model. Future work will study the use of a class-balanced pseudo-labeling scheme to ensure better representativity of training data from the target domain.

Author Contributions: Conceptualization, H.T. and P.P.; methodology, software, validation, formal analysis, H.T.; investigation, H.T. and P.P.; resources, P.P.; writing—original draft preparation, H.T.; writing—review and editing, P.P.; supervision, project administration, funding acquisition, P.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by MITACS Accelerate grant #IT29551 and Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery grant #RGPIN-2020-04307.

Institutional Review Board Statement: This research only involved data from public resources, and no experiments on human subjects were conducted. Ethical review and approval were waived by the Office of Research Ethics and Integrity of the University of Ottawa because the project falls under Article 2.5 of the Tri-Council Policy Statement: Ethical Conduct for Research Involving Humans (TCPS 2) and therefore did not require Research Ethics Board review.

Informed Consent Statement: Not applicable.

Data Availability Statement: Information regarding public data sources is contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kong, Y.; Fu, Y. Human action recognition and prediction survey. *Int. J. Comput. Vis.* **2022**, *130*, 1366–1401. [[CrossRef](#)]
2. Lange, B.; Chang, C.-Y.; Suma, E.; Newman, B.; Rizzo, A.S.; Bolas, M. Development and evaluation of low-cost game-based balance rehabilitation tool using the Microsoft Kinect sensor. In Proceedings of the 2011 annual international conference of the IEEE engineering in medicine and biology society, Boston, MA, USA, 30 August–3 September 2011; pp. 1831–1834.

3. Niu, W.; Long, J.D.; Han, D.; Wang, Y.-F. Human activity detection and recognition or video surveillance. In Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No. 04TH8763), Taipei, Taiwan, 27–30 June 2004; Volume 1, pp. 719–722.
4. Zheng, C.; Wu, W.; Chen, C.; Yang, T.; Zhu, S.; Shen, J.; Kehtarnavaz, N.; Shah, M. Deep learning-based human pose estimation: A survey. *ACM Comput. Surv.* **2023**, *56*, 11. [[CrossRef](#)]
5. Yan, S.; Xiong, Y.; Lin, D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
6. Chen, Y.; Zhang, Z.; Yuan, C.; Li, B.; Deng, Y.; Hu, W. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 13359–13368.
7. Chi, H.G.; Ha, M.H.; Chi, S.; Lee, S.W.; Huang, Q.; Ramani, K. InfoGCN: Representation learning for human skeleton-based action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 20186–20196.
8. Shahroudy, A.; Jun, L.; Tian-Tsong, N.; Gang, W. NTU RGB+D: A large scale dataset for 3d human activity analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019.
9. Liu, C.; Hu, Y.; Li, Y.; Song, S.; Liu, J. PKU-MMD: A large scale benchmark for skeleton-based human action understanding. In Proceedings of the Workshop on Visual Analysis in Smart and Connected Communities, Mountain View, CA, USA, 23 October 2017; pp. 1–2.
10. Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; Vaughan, J.W. A theory of learning from different domains. *Mach. Learn.* **2010**, *79*, 151–175. [[CrossRef](#)]
11. Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. Supervised contrastive learning. In Proceedings of the Advances in Neural Information Processing Systems 33 (NeurIPS 2020), Virtual, 6–12 December 2020; Volume 33, pp. 18661–18673.
12. Li, M.; Chen, S.; Chen, X.; Zhang, Y.; Wang, Y.; Tian, Q. Actional-structural graph convolutional networks for skeleton-based action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3595–3603.
13. Choi, J.; Sharma, G.; Chandraker, M.; Huang, J.-B. Unsupervised and semi-supervised domain adaptation for action recognition from drones. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 1717–1726.
14. Guo, T.; Liu, H.; Chen, Z.; Liu, M.; Wang, T.; Ding, R. Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36, pp. 762–770.
15. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Machine Learning, Virtual, 13–18 July 2020; pp. 1597–1607.
16. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9729–9738.
17. Chen, X.; Fan, H.; Girshick, R.; He, K. Improved baselines with momentum contrastive learning. *arXiv* **2020**, arXiv:2003.04297.
18. Wang, J.; Nie, X.; Xia, Y.; Wu, Y.; Zhu, S.-C. Cross-view action modeling, learning and recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2649–2656.
19. Zheng, N.; Wen, J.; Liu, R.; Long, L.; Dai, J.; Gong, Z. Unsupervised representation learning with long-term dynamics for skeleton based action recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
20. Lin, L.; Song, S.; Yang, W.; Liu, J. Ms2l: Multi-task self-supervised learning for skeleton based action recognition. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 2490–2498.
21. Thoker, F.M.; Doughty, H.; Snoek, C.G. Skeleton-contrastive 3D action representation learning. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual China, 20–24 October 2021; pp. 1655–1663.
22. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.