REVISITING SKELETON BASED ACTION RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Human skeleton, as a compact representation of human action, has received increasing attention in recent years. Many skeleton-based action recognition methods adopt GCNs to extract features on top of human skeletons. Despite the positive results shown in these attempts, GCN-based methods are subject to limitations in robustness, interoperability, and scalability. In this work, we propose PoseConv3D, a new approach to skeleton-based action recognition. PoseConv3D relies on a 3D heatmap stack instead of a graph sequence as the base representation of human skeletons. Compared to GCN-based methods, PoseConv3D is more effective in learning spatiotemporal features, more robust against pose estimation noises, and generalizes better in cross-dataset settings. Also, PoseConv3D can handle multiple-person scenarios without additional computation costs. The hierarchical features can be easily integrated with other modalities at early fusion stages, providing a great design space to boost the performance. PoseConv3D achieves the state-of-the-art on five of six standard skeleton-based action recognition benchmarks. Once fused with other modalities, it achieves the state-of-the-art on all eight multi-modality action recognition benchmarks.

1 INTRODUCTION

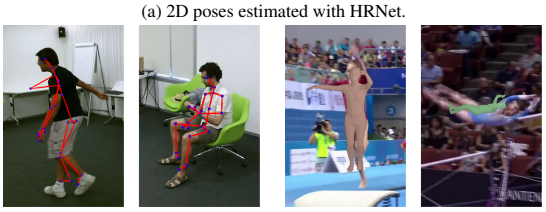
Action recognition is a central task in video understanding. Existing studies have explored various modalities for feature representation, such as RGB frames (Wang et al., 2016; Tran et al., 2015; Carreira & Zisserman, 2017), optical flows (Simonyan & Zisserman, 2014), audio waves (Xiao et al., 2020), and human skeletons (Yan et al., 2018). Among these modalities, skeleton-based action recognition has received increasing attention in recent years due to its action-focusing nature and compactness. In practice, human skeletons in a video are mainly represented as a sequence of joint coordinate lists, where the coordinates are extracted by pose estimators. Since only the pose information is included, skeleton sequences capture only action information while being immune to contextual nuisances, such as background variation and lighting changes.

Among all the methods for skeleton-based action recognition (Du et al., 2015; Wang et al., 2012; Vemulapalli et al., 2014), graph convolutional networks (GCN) (Yan et al., 2018) have been one of the most popular approaches. Specifically, GCNs regard every human joint at every timestep as a node. Neighboring nodes along the spatial and temporal dimensions are connected with edges. Graph convolution layers are then applied to the constructed graph to discover action patterns across space and time. Due to the good performance on standard benchmarks for skeleton-based action recognition, GCNs have been a standard approach when processing skeleton sequences.

While encouraging results have been observed, GCN-based methods are limited in the following aspects: (1) *Robustness:* While GCN directly handles coordinates of human joints, its recognition ability is significantly affected by the distribution shift of coordinates, which can often occur when applying a different pose estimator to acquire the coordinates. A small perturbation in coordinates often leads to completely different predictions (Zhu et al., 2019). (2) *Interoperability:* Previous works have shown that representations from different modalities, such as RGB, optical flows, and skeletons, are complementary. Hence, an effective combination of such modalities can often result in a performance boost in action recognition. However, GCN is operated on an irregular graph of skeletons, making it difficult to fuse with other modalities that are often represented on regular grids, especially in the early stages. (3) *Scalability:* In addition, since GCN regards every human joint as a node, the complexity of GCN scales linearly with the number of persons, limiting its applicability to scenarios that involve multiple persons, such as group activity recognition.

In this paper, we propose a novel framework **PoseConv3D** that serves as a competitive alternative to GCN-based approaches. In particular, PoseConv3D takes as input 2D poses obtained by modern pose estimators shown in Figure 1. The 2D poses are represented by stacks of heatmaps of skeleton joints rather than coordinates operated on a human skeleton graph. The heatmaps at different timesteps will be stacked along the temporal dimension to form a 3D heatmap volume. PoseConv3D then adopts a 3D convolutional neural network on top of the 3D heatmap volume to recognize actions. The main differences between the proposed PoseConv3D and GCN-based approaches are summarized in Table 1.

(a) 2D poses estimated with HRNet.



(b) 3D poses collected with Kinect. (c) 3D poses estimated with VIBE.

Figure 1: **PoseConv3D takes 2D poses as inputs.** In general, 2D poses are of better quality compared to 3D poses. We visualize 2D poses estimated with HRNet for videos in NTU-60 and FineGYM in (a). Apparently, their quality is much better than 3D poses collected by sensors (b) or estimated with state-of-the-art estimators (c).

Table 1: **Differences between PoseConv3D and GCN.**

	Previous Work	PoseConv3D
Input	2D / 3D Skeleton	2D Skeleton
Format	Coordinates	3D Heatmap Volumes
Architecture	GCN	3D-CNN

PoseConv3D can address the limitations of GCN-based approaches stated above. First, using 3D heatmap volumes is more robust to the up-stream pose estimation: we empirically find that PoseConv3D generalizes well across input skeletons obtained by different approaches. Also, PoseConv3D, which relies on heatmaps of the base representation, enjoys the recent advances in convolutional network architectures and is easier to integrate with other modalities into multi-stream convolutional networks. This characteristic opens up great design space to further improve the recognition performance. Finally, PoseConv3D can handle different numbers of persons without increasing computational overhead since the complexity over 3D heatmap volume is independent of the number of persons. To verify the efficiency and effectiveness of PoseConv3D, we conduct comprehensive studies across several datasets, including FineGYM (Shao et al., 2020), NTURGB-D (Liu et al., 2019), UCF101 (Soomro et al., 2012), HMDB51 (Kuehne et al., 2011), Kinetics400 (Carreira & Zisserman, 2017), and Volleyball (Ibrahim et al., 2016), where PoseConv3D achieves state-of-the-art performance compared to GCN-based approaches.

2 Related Work

3D-CNN for RGB-based action recognition. 3D-CNN is a natural extension of 2D-CNN for spatial feature learning to spatiotemporal in videos. It has long been used in action recognition (Ji et al., 2012; Tran et al., 2015). Due to a large number of parameters, 3D-CNN requires huge amounts of videos to learn good representation. 3D-CNN has become the mainstream approach for action recognition since Carreira & Zisserman (2017) proposed I3D and the large-scale dataset Kinetics400. From then on, many advanced 3D-CNN architectures (Tran et al., 2018; Feichtenhofer et al., 2019; Tran et al., 2019; Feichtenhofer, 2020) have been proposed by the action recognition community, which outperform I3D both in precision and efficiency. In this work, we first propose to use 3D-CNN with 3D heatmap volumes as inputs and reach the state-of-the-art in skeleton-based action recognition.

GCN for skeleton-based action recognition. Graph convolutional network is widely adopted in skeleton-based action recognition. It models human skeleton sequences as spatiotemporal graphs. ST-GCN (Yan et al., 2018) is a well-known baseline for GCN-based approaches, which combines spatial graph convolutions and interleaving temporal convolutions for spatiotemporal modeling. Upon the baseline, adjacency powering is used for multiscale modeling (Liu et al., 2020; Li et al., 2019b), while self-attention mechanisms improve the modeling capacity (Shi et al., 2019b; Li et al., 2019a). Despite the great success of GCN in skeleton-based action recognition, it is also limited in robustness (Zhu et al., 2019) and scalability. Besides, for GCN-based approaches, fusing features from skeletons and other modalities may need careful design (Das et al., 2020).

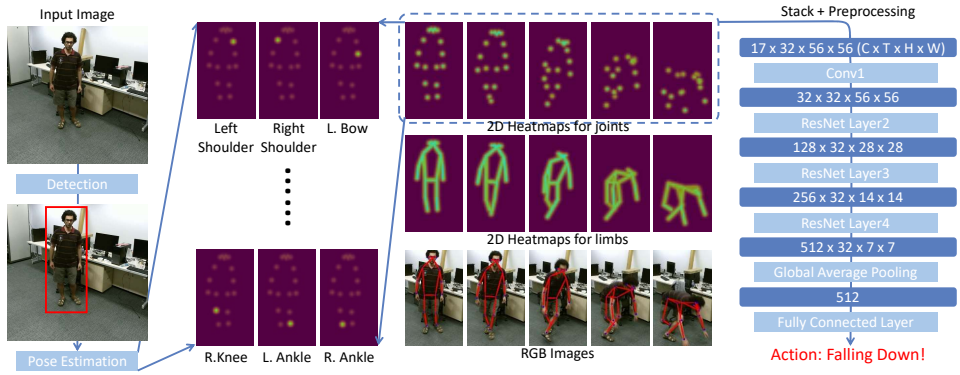


Figure 2: **Our Framework.** For each frame in a video, we first use a two-stage pose estimator (detection + pose estimation) for 2D human pose extraction. Then we stack heatmaps of joints or limbs along the temporal dimension and apply pre-processing to the generated 3D heatmap volumes. Finally, we use a 3D-CNN to classify the 3D heatmap volumes.

CNN for skeleton-based action recognition. Another stream of work adopts convolutional neural networks for skeleton-based action recognition. 2D-CNN-based approaches first model the skeleton sequence as a pseudo image based on manually designed transformations. PoTion (Choutas et al., 2018) aggregates heatmaps along the temporal dimension with color encodings to get the pose motion representation, while PA3D (Yan et al., 2019) does the aggregation with 1×1 convolutions. Although carefully designed, information loss still occurs during the aggregation, which leads to inferior recognition performance. Other works (Ke et al., 2017; Luvizon et al., 2018; Caetano et al., 2019) directly convert the coordinates in a skeleton sequence to a pseudo image with transformations, typically generate a 2D input of shape $K \times T$, where K is the number of joints, T is the temporal length. Such input cannot exploit the locality nature of convolution networks, which makes these methods not as competitive as GCN on popular benchmarks (Caetano et al., 2019). Only a few previous works have adopted 3D-CNNs for skeleton-based action recognition. To construct the 3D input, they either stack the pseudo images of distance matrices (Hernandez Ruiz et al., 2017; Lin et al., 2020) or directly aggregate the 3D skeletons into a cuboid (Liu et al., 2017). These approaches also severely suffer from information loss and obtain much inferior performance to the state-of-the-art. Our work aggregates heatmaps by stacking them along the temporal dimension to form 3D heatmap volumes, preserving all information during this process. Besides, we use 3D-CNN instead of 2D-CNN due to its good capability for spatiotemporal feature learning.

3 Framework

We propose **PoseConv3D**, a **3D-CNN**-based approach for skeleton-based action recognition, which can be a competitive alternative to GCN-based approaches, outperforming GCN under various settings in terms of accuracy with improved robustness, interoperability, and scalability. An overview of PoseConv3D is depicted in Figure 2, and details of PoseConv3D will be covered in the following sections. We begin with a review of skeleton extraction, which is the basis of skeleton-based action recognition but is often overlooked in previous literature. We point out several aspects that should be considered when choosing a skeleton extractor and motivate the use of 2D skeletons in PoseConv3D[1]. Subsequently, we introduce 3D Heatmap Volume that is the representation of a 2D skeleton sequence used in PoseConv3D, followed by the structural designs of PoseConv3D, including a variant that focuses on the modality of human skeletons as well as a variant that combines the modalities of human skeletons and RGB frames to demonstrate the interoperability of PoseConv3D.

3.1 Good Practices for Pose Extraction

Being a critical pre-processing step for skeleton-based action recognition, human skeleton or pose extraction largely affects the final recognition accuracy. However, its importance is often overlooked in previous literature, in which poses estimated by sensors (Shahroudy et al., 2016; Liu et al., 2019)

[1]PoseConv3D can also work with 3D skeletons. An example solution is to divide a 3D skeleton (x, y, z) into three 2D skeletons respectively using (x, y), (y, z) and (x, z).

or existing pose estimators (Cao et al., 2019; Yan et al., 2018) are used without considering the potential effects. Here we conduct a review on key aspects of pose extraction to find a good practice.

In general, 2D poses are of better quality compared to 3D poses, as shown in Figure 1. We adopt 2D Top-Down pose estimators (Newell et al., 2016; Xiao et al., 2018; Sun et al., 2019) for pose extraction. Compared to its 2D Bottom-Up counterparts (Newell et al., 2017; Cao et al., 2017; Cheng et al., 2020a), Top-Down methods obtain superior performance on standard benchmarks such as COCO-keypoints (Lin et al., 2014). In most cases, we feed proposals predicted by a human detector to the Top-Down pose estimators, which is sufficient enough to generate 2D poses of good quality for action recognition. When only a few persons are of interest out of dozens of candidates [2], some priors are essential for skeleton-based action recognition to achieve good performance, *e.g.*, knowing the interested person locations at the first frame of the video. In terms of the storage of estimated heatmaps, they are often stored as coordinate-triplets (x, y, c) in previous literature, where c marks the maximum score of the heatmap and (x, y) is the corresponding coordinate of c. In experiments, we find that coordinate-triplets (x, y, c) help save the majority of storage space at the cost of little performance drop. The detailed ablation study is included in Appendix Sec. A.4.1.

3.2 From 2D Poses to 3D Heatmap Volumes

After 2D poses are extracted from video frames, to feed into PoseConv3D, we reformulate them into a 3D heatmap volume. Formally, we represent a 2D pose as a heatmap of size $K \times H \times W$, where K is the number of joints, H and W are the height and width of the frame. We can directly use the heatmap produced by the Top-Down pose estimator as the target heatmap, which should be zero-padded to match the original frame given the corresponding bounding box. In case we have only coordinate-triplets (x_k, y_k, c_k) of skeleton joints, we can obtain a joint heatmap $\boldsymbol{J}$ by composing K gaussian maps centered at every joint:

$$\boldsymbol{J}_{kij} = \exp(-[(i - x_k)^2 + (j - y_k)^2]/(2 * \sigma^2)) * c_k \tag{1}$$

where σ controls the variance of gaussian maps, and (x_k, y_k) and c_k are respectively the location and confidence score of the k-th joint. We can also create a limb heatmap $\boldsymbol{L}$:

$$\boldsymbol{L}_{kij} = \exp(-\mathcal{D}((i, j), [(x_{a_k}, y_{a_k}), (x_{b_k}, y_{b_k})])^2/(2 * \sigma^2)) * \min(c_{a_k}, c_{b_k}). \tag{2}$$

The k_{th} limb is between two joints a_k and b_k. The function $\mathcal{D}$ calculates the distance from the point (i, j) to the segment $[(x_{a_k}, y_{a_k}), (x_{b_k}, y_{b_k})]$. It is worth noting that although the above process assumes a single person in every frame, we can easily extend it to the multi-person case, where we directly accumulate the k-th gaussian maps of all persons without enlarging the heatmap. Finally, a 3D heatmap volume is obtained by stacking all heatmaps ($\boldsymbol{J}$ or $\boldsymbol{L}$) along the temporal dimension, which thus has the size of $K \times T \times H \times W$.

In practice, we further apply two techniques to reduce the redundancy of 3D heatmap volumes. (1) **Subjects-Centered Cropping.** Making the heatmap as large as the frame is inefficient, especially when the persons of interest only act in a small region. In such cases, we first find the smallest bounding box that envelops *all* the 2D poses across frames. Then we crop all frames according to the found box and resize them to the target size. Consequently, the size of the 3D heatmap volume can be reduced spatially while all 2D poses and their motion are kept. (2) **Uniform Sampling.** The 3D heatmap volume can also be reduced along the temporal dimension by sampling a subset of frames. Unlike previous works on RGB-based action recognition, where researchers usually sample frames in a short temporal window, such as sampling frames in a 64-frame temporal window as in SlowFast (Feichtenhofer et al., 2019), we propose to use a uniform sampling strategy (Wang et al., 2016) for 3D-CNNs instead. In particular, to sample n frames from a video, we divide the video into n segments of equal length and randomly select one frame from each segment. The uniform sampling strategy is better at maintaining the global dynamics of the video. Our empirical studies show that the uniform sampling strategy is significantly beneficial for skeleton-based action recognition. More illustration about generating 3D heatmap volumes is provided in Appendix Sec. A.2.

3.3 3D-CNN for Skeleton-based Action Recognition

For skeleton-based action recognition, GCN has long been the mainstream backbone. In contrast, 3D-CNN, an effective network structure commonly used in RGB-based action recognition (Carreira

[2] In FineGym, there exists dozens of audience, while only the pose of the athlete matters.

& Zisserman, 2017; Hara et al., 2018; Feichtenhofer et al., 2019), is less explored in this direction. To demonstrate the power of 3D-CNN in capturing spatiotemporal dynamics of skeleton sequences, we design two families of 3D-CNNs, namely **PoseConv3D** for the *Pose* modality and **RGBPose-Conv3D** for the *RGB+Pose* dual-modality.

PoseConv3D. PoseConv3D focuses on the modality of human skeletons, which takes 3D heatmap volumes as input and can be instantiated with various 3D-CNN backbones. Two modifications are needed to adapt 3D-CNNs to skeleton-based action recognition: (1) down-sampling operations in early stages are removed from the 3D-CNN since the spatial resolution of 3D heatmap volumes does not need to be as large as RGB clips ($4\times$ smaller in our setting); (2) a shallower (fewer layers) and thinner (fewer channels) network is sufficient to model spatiotemporal dynamics of human skeleton sequences since 3D heatmap volumes are already mid-level features for action recognition. Based on these principles, we adapt three popular 3D-CNNs: C3D (Tran et al., 2015), SlowOnly (Feichtenhofer et al., 2019), and X3D (Feichtenhofer, 2020), to skeleton-based action recognition (Table 11 demonstrates the architectures of the three backbones as well as their variants). The different variants of adapted 3D-CNNs are evaluated on the NTURGB+D-XSub benchmark (Table 2). Adopting a lightweight version of 3D-CNNs can significantly reduce the computational complexity at the cost of a slight recognition performance drop ($\leq 0.3\%$ for all 3D backbones). In experiments, we use SlowOnly as the default backbone, considering its simplicity (directly inflated from ResNet) and good recognition performance. PoseConv3D can outperform representative GCN / 2D-CNN counterparts across various benchmarks, both in accuracy and efficiency. More importantly, the interoperability between PoseConv3D and popular networks for RGB-based action recognition makes it easy to involve human skeletons in multi-modality fusion.

Table 2: **Evalution of PoseConv3D variants.** 's' indicates shallow (fewer layers); 'HR' indicates high-resolution (double height & width); 'wd' indicates wider network with double channel size.

Backbone	Variant	NTU60-XSub	FLOPs	Params
SlowOnly	-	93.7	15.9G	2.0M
SlowOnly	HR	93.6	73.0G	8.0M
SlowOnly	wd	93.7	54.9G	7.9M
C3D	-	93.0	25.2G	6.9M
C3D	s	92.9	16.8G	3.4M
X3D	-	92.6	1.1G	531K
X3D	s	92.3	0.6G	241K

RGBPose-Conv3D. To show the interoperability of PoseConv3D, we propose RGBPose-Conv3D for the early fusion of human skeletons and RGB frames. It is a two-stream 3D-CNN with two pathways that respectively process RGB modality and Pose modality. While a detailed instantiation of RGBPose-Conv3D is included in Appendix Sec. A.3.2, the architecture of RGBPose-Conv3D follows several principles in general: (1) the two pathways are not symmetrical due to the different characteristics of the two modalities: Compared to the RGB pathway, the pose pathway has a smaller channel-width, a smaller depth, as well as a smaller input spatial resolution. (2) Inspired by SlowFast (Feichtenhofer et al., 2019), bidirectional lateral connections between the two pathways are added to promote early-stage feature fusion between two modalities. To avoid overfitting, RGBPose-Conv3D is trained with two individual cross-entropy losses respectively for each pathway. In experiments, we find that early-stage feature fusion caused by the lateral connections can lead to significant and consistent improvement compared to late-fusion only.

4 Experiments

4.1 Dataset Preparation

We use six datasets in our experiments: FineGYM (Shao et al., 2020), NTURGB+D (Shahroudy et al., 2016; Liu et al., 2019), Kinetics400 (Carreira & Zisserman, 2017; Yan et al., 2018), UCF101 (Soomro et al., 2012), HMDB51 (Kuehne et al., 2011) and Volleyball (Ibrahim et al., 2016). Unless otherwise specified, we use the Top-Down approach for pose extraction: the detector is Faster-RCNN (Ren et al., 2015) with the ResNet50 backbone, the pose estimator is HRNet (Sun et al., 2019) pre-trained on COCO-keypoint (Lin et al., 2014). For all datasets except FineGYM, 2D poses are obtained by directly applying Top-Down pose estimators to RGB inputs. We report the **Mean Top-1** accuracy for FineGYM and **Top-1** accuracy for other datasets. Codes and estimated 2D poses will be released.

FineGYM. FineGYM is a fine-grained action recognition dataset with 29K videos of 99 fine-grained gymnastic action classes. During pose extraction, we compare three different kinds of

Table 3: **PoseConv3D is better or comparable to previous state-of-the-arts.** With estimated high-quality 2D skeletons and the great capacity of 3D-CNN to learn spatiotemporal features, PoseConv3D achieves superior performance across **5 out of 6** benchmarks. $\boldsymbol{J}$, $\boldsymbol{L}$ means using joint- and limb-based heatmap respectively. ++ denotes using the same pose estimation result as ours. * means the number is reported by Shao et al. (2020).

Method	NTU60-XSub	NTU60-XView	NTU120-XSub	NTU120-XSet	Kinetics	FineGYM
ST-GCN (Yan et al., 2018)	81.5	88.3	70.7	73.2	30.7	25.2*
AS-GCN (Li et al., 2019b)	86.8	94.2	78.3	79.8	34.8	-
RA-GCN (Song et al., 2020)	87.3	93.6	81.1	82.7	-	-
AGCN (Shi et al., 2019b)	88.5	95.1	-	-	36.1	-
DGNN (Shi et al., 2019a)	89.9	96.1	-	-	36.9	-
FGCN (Yang et al., 2020)	90.2	96.3	85.4	87.4	-	-
Shift-GCN (Cheng et al., 2020b)	90.7	96.5	85.9	87.6	-	-
DSTA-Net (Shi et al., 2020)	91.5	96.4	86.6	89.0	-	-
MS-G3D (Liu et al., 2020)	91.5	96.2	86.9	88.4	38.0	-
MS-G3D ++	92.2	96.6	**87.2**	89.0	45.1	92.6
PoseConv3D ($\boldsymbol{J}$)	**93.7**	**96.6**	86.0	**89.6**	**46.0**	**93.2**
PoseConv3D ($\boldsymbol{J}+\boldsymbol{L}$)	**94.1**	**97.1**	86.9	**90.3**	**47.7**	**94.3**

person bounding boxes: 1. Person bounding boxes predicted by the detector (**Detection**); 2. GT bounding boxes for the athlete in the first frame, tracking boxes for the rest frames (**Tracking**). 3. GT bounding boxes for the athlete in all frames (**GT**). In experiments, we use human poses extracted with the third kind of bounding boxes unless otherwise noted.

NTURGB+D. NTURGB+D is a large-scale human action recognition dataset collected in the lab. It has two versions, namely NTU-60 and NTU-120 (a superset of NTU-60): NTU-60 contains 57K videos of 60 human actions, while NTU-120 contains 114K videos of 120 human actions. The datasets are split in three ways: Cross-subject (**X-Sub**), Cross-view (**X-View**, for NTU-60), Cross-setup (**X-Set**, for NTU-120), for which action subjects, camera views, camera setups are different in training and validation. The 3D skeletons collected by sensors are available for this dataset. Unless otherwise specified, we conduct experiments on the **X-sub** splits for NTU-60 and NTU-120.

Kinetics400, UCF101, and HMDB51. The three datasets are general action recognition datasets collected from the web. Kinetics400 is a large-scale video dataset with 300K videos from 400 action classes. UCF101 and HMDB51 are smaller, contains 13K videos from 101 classes and 6.7K videos from 51 classes, respectively. We conduct experiments using 2D-pose annotations extracted with our Top-Down pipeline.

Volleyball. Volleyball is a group activity recognition dataset with 4830 videos of 8 group activity classes. Each frame contains approximately 12 persons, while only the center frame is annotated with GT person boxes. We use tracking boxes from (Sendo & Ukita, 2019) for pose extraction.

4.2 Comparison with state-of-the-arts

Skeleton-based Action Recognition. PoseConv3D achieves competitive results on multiple datasets. In Table 3, we compare Pose-SlowOnly with state-of-the-arts in skeleton-based action recognition. Since the 2D poses we used are of better quality than 2D/3D poses used previously, we also evaluate the state-of-the-art MS-G3D with our 2D-pose annotations. The extracted 2D poses are saved as coordinate-triplets (x, y, c), directly used by *MS-G3D++*. For *PoseConv3D*, we also use pseudo heatmaps generated from coordinate-triplets as input, thus a fair comparison. We achieve by far the best results on **three of four** NTURGB+D benchmarks, prove that high-quality 2D skeletons with PoseConv3D can yield competitive performance on skeleton-based action recognition. On Kinetics, PoseConv3D surpasses the state-of-the-art MS-G3D by a noticeable margin when using the same input, significantly outperforms previous methods. Except for the baseline obtained by Shao et al. (2020), no work aims at skeleton-based action recognition on FineGYM before, while our work first improves the performance to a decent level.

Multi-modality Fusion. As a powerful representation itself, skeletons are also complementary to other modalities, like RGB appearance. With multi-modality fusion (*RGBPose-Conv3D* or *Late-Fusion*), we achieve state-of-the-art results across **eight** different video recognition benchmarks. We apply the proposed *RGBPose-Conv3D* to FineGYM and four NTURGB+D benchmarks, using ResNet50 as the backbone, 16, 48 as the temporal length for *RGB*-Pathway and *Pose*-Pathway. Table 4a shows that our **early+late** fusion strategy achieves excellent performance across various

Table 4: **Comparison to the state-of-the-art of Multi-Modality Action Recognition.** Perfect recognition performance is achieved on multiple benchmarks with multi-modality fusion. R, F, P indicate RGB, Flow, Pose.

(a) Mulit-modality action recognition with RGBPose-Conv3D.

RGBPose-Conv3D	Previous state-of-the-art	Ours
FineGYM-99	87.7 (R) (Kwon et al., 2021)	**95.6** (R + P)
NTU60 (X-Sub / X-View)	95.7 / 98.9 (R + P) (Davoodikakhki & Yin, 2020)	**97.0 / 99.6** (R + P)
NTU120 (X-Sub / X-Set)	90.7 / 92.5 (R + P) (Das et al., 2021)	**95.3 / 96.4** (R + P)

(b) Mulit-modality action recognition with late fusion.

LateFusion Only	Previous state-of-the-art	Ours
Kinetics400	84.9 (R) (Liu et al., 2021)	**85.5** (R + P)
UCF101	98.6 (R + F) (Duan et al., 2020)	**98.8** (R + F + P)
HMDB51	83.8 (R + F) (Duan et al., 2020)	**85.0** (R + F + P)

benchmarks. We also try to fuse the predictions of PoseConv3D directly with other modalities with *LateFusion*. Table 4b shows that late fusion with the Pose modality can push the recognition precision to a new level. We achieve the new state-of-the-art on three action recognition benchmarks: Kinetics400, UCF101, and HMDB51. On the challenging Kinetics400 benchmark, fusing with PoseConv3D predictions increases the recognition accuracy by 0.6% beyond the state-of-the-art (Liu et al., 2021), which is strong evidence for the complementarity of the Pose modality.

4.3 Preprocessing of 3D Heatmap Volumes

Subjects-Centered Cropping. Since the sizes and locations of persons can vary a lot in a dataset, focusing on the action subjects is the key to reserve as much information as possible with a relatively small $H \times W$ budget. To validate this, we conduct a pair of experiments on FineGYM with input size $32\times56\times56$, with or without subjects-centered cropping. We find that subjects-centered cropping is helpful in data preprocessing, which improves the Mean-Top1 by 1.0%, from 91.7% to 92.7%.

Uniform Sampling. The input sampled from a small temporal window may not capture the entire dynamic of the human action. To validate this, we conduct experiments on FineGYM and NTU-60. For fixed stride sampling, which samples from a fixed temporal window, we try to sample 32 frames with the temporal stride 2, 3, 4; for uniform sampling, we sample 32 frames uniformly from the entire clip. From Figure 3, we see that uniform sampling consistently outperforms sampling with fixed temporal strides. With uniform sampling, 1-clip testing can even achieve better results than fixed stride sampling with 10-clip testing. Note that the video length can vary a lot in NTU-60 and FineGYM. In a more detailed analysis, we find that uniform sampling mainly improves the recognition performance for longer videos in the dataset (Figure 4). Besides, uniform sampling also outperforms fixed stride sampling on RGB-based recognition on the two datasets[3].

Pseudo Heatmaps for Joints and Limbs. GCN-based approaches for skeleton-based action recognition usually ensemble results of multiple streams (joint stream, bone stream, *etc.*) to achieve better recognition performance (Shi et al., 2019b). That practice is also feasible for PoseConv3D. Based on the coordinate-triplets (x, y, c) we saved, we can generate pseudo heatmaps for joints and limbs. In general, we find that both joint heatmaps and limb heatmaps are good inputs for 3D-CNNs. Ensembling the results from joint-PoseConv3D and limb-PoseConv3D (namely PoseConv3D ($\boldsymbol{J}+\boldsymbol{L}$)) can lead to noticeable and consistent performance improvement.

3D Heatmap Volumes *v.s* 2D Heatmap Aggregations. The 3D heatmap volume is a more 'lossless' 2D-pose representation, than 2D heatmap aggregations aggregated with colorization (PoTion) or temporal convolutions (PA3D). PoTion (Choutas et al., 2018) and PA3D (Yan et al., 2019) are not evaluated on popular benchmarks for skeleton-based action recognition, and there are no public implementations. In the preliminary study, we find that the accuracy of PoTion is much inferior ($\leq 85\%$) to GCN or PoseConv3D (all $\geq 90\%$). For an apple-to-apple comparison, we also re-implement PoTion, PA3D (with higher accuracy than reported) and evaluate them on three benchmarks: UCF101, HMDB51, NTURGB+D. PoseConv3D achieves much better recognition results with 3D heatmap volumes, than 2D-CNNs with 2D heatmap aggregations as inputs. With the lightweight X3D backbone, PoseConv3D significantly outperforms 2D-CNNs, with comparable FLOPs and far fewer parameters (Table 5).

Table 5: **An apple-to-apple comparison between 3D heatmap volumes and 2D heatmap aggregations.**

Method	HMDB51	UCF101	NTU60-XSub	FLOPs	Params
PoTion (Choutas et al., 2018)	51.7	67.2	87.8	0.60G	4.75M
PA3D (Yan et al., 2019)	53.5	69.1	88.6	0.65G	4.81M
Pose-SlowOnly (Ours)	**58.6**	**79.1**	**93.7**	15.9G	**2.0M**
Pose-X3D-s (Ours)	**55.6**	**76.7**	**92.3**	**0.60G**	**0.24M**

[3]Please refer to Appendix Sec. A.4.4 for details and discussions.

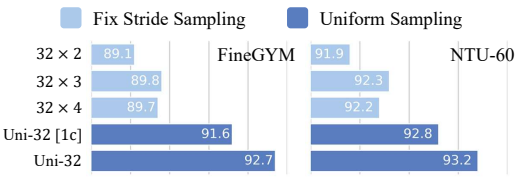


Figure 3: **Uniform Sampling outperforms Fix-Stride Sampling.** All results are for 10-clip testing, except Uni-32[1c], which uses 1-clip testing.

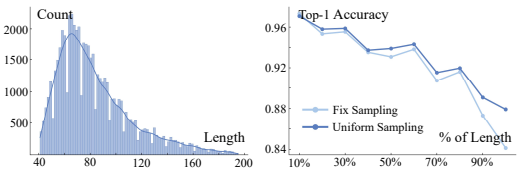


Figure 4: **Uniform Sampling helps in modeling longer videos.** Left: The length distribution of NTU60-XSub val videos. Right: Uniform Sampling improves the recognition accuracy of longer videos.

4.4 Good properties of PoseConv3D

To elaborate on the good properties, we compare Pose-SlowOnly with MS-G3D (Liu et al., 2020), a representative GCN-based approach in multiple dimensions. Two models take exactly the **same** input (coordinate-triplets for GCN, heatmaps generated from coordinate-triplets for PoseConv3D).

4.4.1 Performance & Efficiency

In performance comparison between PoseConv3D and GCN, we adopt the input shape $48\times56\times56$ for PoseConv3D. Table 6 shows that under such configuration, our PoseConv3D is even lighter than the GCN counterpart, both in the number of parameters and FLOPs. Although being lightweight, PoseConv3D achieves competitive performance across different datasets. The 1-clip testing result is better than or comparable with a state-of-the-art GCN while requiring much less computation. When applying 10-clip testing, PoseConv3D consistently outperforms the state-of-the-art GCN. Only PoseConv3D can take advantage of multi-view testing since it subsamples the entire heatmap volumes to form each input. Besides, PoseConv3D uses the same architecture and hyperparameters for different datasets and achieves competitive performance, while GCN tunes architectures and hyperparameters for different datasets (Liu et al., 2020).

4.4.2 Robustness & Generalization

Robustness. To test the robustness of both models, we can drop a proportion of keypoints in the input and see how such perturbation will affect the final accuracy. Since limb keypoints[4] are more critical for gymnastics than the torso or face keypoints, we test both models by randomly dropping one limb keypoint in each frame with probability p. In Table 7, we see that PoseConv3D is highly robust to input perturbations: dropping one limb keypoint per frame leads to a moderate drop (less than 1%) in Mean-Top1, while for GCN, it's 14.3%. Someone would argue that we can train GCN with the noisy input, similar to the dropout operation (Srivastava et al., 2014). However, even under this setting, the Mean-Top1 accuracy of GCN still drops by 1.4% for the case $p = 1$. Besides, with robust training, there will be an additional 1.1% drop for the case $p = 0$. The experiment results show that PoseConv3D significantly outperforms GCN in terms of robustness for pose recognition.

Generalization. To compare the generalization of 3D-CNN and GCN, we design a cross-model check on FineGYM. Specifically, we use two models, *i.e.*, HRNet (Higher-Quality, or HQ for short) and MobileNet (Lower-Quality, LQ) for pose estimation, and train PoseConv3D on top respectively. During testing, we feed LQ input into the model trained with HQ one and vice versa. From Table 8a, we see that the accuracy drops less when using lower-quality poses for both training & testing with PoseConv3D compared to GCN. Similarly, we can also vary the source of person boxes, using either **GT** boxes (HQ) or **tracking** results (LQ), for training and testing. The results are shown in Table 8b. The performance drop of PoseConv3D is also much smaller than GCN.

4.4.3 Scalability

The computation of GCN scales linearly with the increasing number of persons in the video, making it less efficient for group activity recognition. We use an experiment on the Volleyball dataset (Ibrahim et al., 2016) to prove that. Each video in the dataset contains 13 persons and 20 frames. For GCN, the corresponding input shape will be $13\times20\times17\times3$, **13** times larger than the input for one person. Under such configuration, the number of parameters and FLOPs for GCN

[4]There are eight limb keypoints: bow, wrist, knee, ankle (left/right).

Table 6: **3D-CNN *v.s.* GCN.** We compare the performance of 3D-CNN and GCN on several datasets. For 3D-CNN, we report the results of 1/10-clip testing. We exclude parameters and FLOPs of the FC layer, since it depends on the number of classes.

	GCN			Pose-SlowOnly			
Dataset	Acc	Params	FLOPs	1-clip	10-clip	Params	FLOPs
FineGYM	92.0	2.8M	24.7G	**92.4**	**93.2**	**2.0M**	**15.9G**
NTU-60	91.9	2.8M	16.7G	**93.1**	**93.7**		
NTU-120	84.8	2.8M	16.7G	**85.1**	**86.0**		
Kinetics400	**44.9**	2.8M	17.5G	44.8	**46.0**		

Table 7: **Recognition performance w. different dropping KP probabilities.** 3D-CNN is more robust to input perturbations.

Method / p	0	1/8	1/4	1/2	1
GCN	92.0	91.0	90.2	86.5	77.7
GCN w. robust training	90.9	91.0	91.0	91.0	90.6
Pose-SlowOnly	**92.4**	**92.4**	**92.3**	**92.1**	**91.5**

Table 8: **Train/Test w. different pose annotations.** 3D-CNN shows great generalization capability in the cross-PoseAnno setting (LQ for low-quality; HQ for high-quality).

	Train → Test		
	HQ → LQ	LQ → HQ	LQ → LQ
GCN	79.3	87.9	89.0
3D-CNN	**86.5**	**91.6**	**90.7**

(a) Train/Test w. Pose from different estimators.

	Train → Test		
	HQ → LQ	LQ → HQ	LQ → LQ
GCN	78.5	89.1	82.9
3D-CNN	**82.1**	**90.6**	**85.4**

(b) Train/Test w. Pose extracted with different boxes.

Table 9: **The design of RGBPose-Conv3D.** Bi-directional lateral connections outperform uni-directional ones in the early stage feature fusion.

	Late Fusion	RGB→Pose	Pose→RGB	RGB↔Pose
1-clip	92.6	93.0	93.4	**93.6**
10-clip	93.4	93.7	93.8	**94.1**

Table 10: **The universality of RGBPose-Conv3D.** The **early+late** fusion strategy works both on RGB-dominant NTU-60 and Pose-dominant FineGYM.

	RGB	Pose	Late Fusion	Early+Late Fusion
FineGYM	87.2 / 88.5	91.0 / 92.0	92.6 / 93.4	**93.6 / 94.1**
NTU-60	94.1 / 94.9	92.8 / 93.2	95.5 / 96.0	**96.2 / 96.5**

is 2.8M and 7.2G (13×). For PoseConv3D, we can use one **single** heatmap volume (with shape 17×12×56×56) to represent all 13 persons. The base channel-width of Pose-SlowOnly is set to 16. Under such a configuration, Pose-SlowOnly only takes 0.52M parameters and 1.6 GFLOPs. Despite the much smaller parameters and FLOPs, PoseConv3D achieves 91.3% Top-1 accuracy on Volleyball-validation, 2.1% higher than the GCN-based approach.

4.5 RGBPose-Conv3D

The 3D-CNN architecture of PoseConv3D makes it more flexible to fuse pose with other modalities via some early fusion strategies. For example, in *RGBPose*-Conv3D, lateral connections between the *RGB*-pathway and *Pose*-pathway are exploited for cross-modality feature fusion in the early stage. In practice, we first train two models for RGB and Pose modalities separately and use them to initialize the *RGBPose*-Conv3D. We continue to finetune the network for several epochs to train the lateral connections. The final prediction is achieved by late fusing the prediction scores from both pathways. *RGBPose*-Conv3D can achieve better fusing results with **early+late** fusion.

Our experiments are based on *RGBPose*-Conv3D instantiated as Table 12. We first compare uni-directional lateral connections and bi-directional lateral connections in Table 9. The result shows that bi-directional feature fusion is better than uni-directional ones for RGB and Pose. With bi-directional feature fusion in the early stage, the **early+late** fusion with 1-clip testing can outperform the **late** fusion with 10-clip testing. Besides, *RGBPose*-Conv3D also works in situations when the importance of two modalities is different. In FineGYM, Pose modality is more important, while in NTU-60, RGB modality is more important, yet we observe performance improvement by **early+late** fusion on both of them in Table 10.

5 Conclusion

In this work, we propose **PoseConv3D**: a 3D-CNN-based approach for skeleton-based action recognition, which takes 3D heatmap volumes as input. PoseConv3D resolves the limitations of GCN-based approaches in *robustness*, *interoperability*, and *scalability*. With light-weighted 3D-ConvNets and compact 3D heatmap volumes as input, PoseConv3D outperforms GCN-based approaches in both accuracy and efficiency. Based on PoseConv3D, we achieve state-of-the-art on both skeleton-based and multi-modality-based action recognition across multiple benchmarks.

6 Reproducibility Statement

Reproducing PoseConv3D doesn't take much effort. The architectures of PoseConv3D (three backbones as well as their variants) and RGBPose-Conv3D are described in detail in Table 11, 12. We also demonstrate our practice of pose extraction in detail in Sec. A.2, as well as provide a jupyter notebook for illustration. We utilize opensource codebases for pose extraction, including MMDetection (Chen et al., 2019) and MMPose (Contributors, 2020). For reproducibility, we will also release the codes, estimated 2D poses for six datasets, and the trained checkpoints.

References

Carlos Caetano, Jessica Sena, François Brémond, Jefersson A Dos Santos, and William Robson Schwartz. Skelemotion: A new representation of skeleton joint sequences based on motion information for 3d action recognition. In *AVSS*, pp. 1–8. IEEE, 2019. 3

Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *TPAMI*, 2019. 4

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 4

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pp. 6299–6308, 2017. 1, 2, 4, 5, 15, 19

Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv:1906.07155*, 2019. 10

Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *CVPR*, 2020a. 4

Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *CVPR*, pp. 183–192, 2020b. 6

Vasileios Choutas, Philippe Weinzaepfel, Jérôme Revaud, and Cordelia Schmid. Potion: Pose motion representation for action recognition. In *CVPR*, pp. 7024–7033, 2018. 3, 7, 19

MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. https://github.com/open-mmlab/mmpose, 2020. 10

Srijan Das, Saurav Sharma, Rui Dai, Francois Bremond, and Monique Thonnat. Vpn: Learning video-pose embedding for activities of daily living. In *ECCV*, pp. 72–90. Springer, 2020. 2

Srijan Das, Rui Dai, Di Yang, and Francois Bremond. Vpn++: Rethinking video-pose embeddings for understanding activities of daily living. *arXiv:2105.08141*, 2021. 7

Mahdi Davoodikakhki and KangKang Yin. Hierarchical action classification with network pruning. In *ISVC*, pp. 291–305. Springer, 2020. 7

Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *CVPR*, pp. 1110–1118, 2015. 1

Haodong Duan, Yue Zhao, Yuanjun Xiong, Wentao Liu, and Dahua Lin. Omni-sourced webly-supervised learning for video recognition. In *ECCV*, pp. 670–688. Springer, 2020. 7, 19

Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *CVPR*, pp. 203–213, 2020. 2, 5, 16

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pp. 6202–6211, 2019. 2, 4, 5, 17, 20

Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, pp. 6546–6555, 2018. 5

Alejandro Hernandez Ruiz, Lorenzo Porzi, Samuel Rota Bulò, and Francesc Moreno-Noguer. 3d cnns on distance matrices for human action recognition. In *MM*, pp. 1087–1095, 2017. 3

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*, 2017. 18

Mostafa S Ibrahim, Srikanth Muralidharan, Zhiwei Deng, Arash Vahdat, and Greg Mori. A hierarchical deep temporal model for group activity recognition. In *CVPR*, 2016. 2, 5, 8, 15

Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *TPAMI*, 35(1):221–231, 2012. 2

Qiuhong Ke, Mohammed Bennamoun, Senjian An, Ferdous Sohel, and Farid Boussaid. A new representation of skeleton sequences for 3d action recognition. In *CVPR*, 2017. 3

Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, pp. 5253–5263, 2020. 18

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *NeurIPS*, 25:1097–1105, 2012. 15

Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, pp. 2556–2563. IEEE, 2011. 2, 5

Heeseung Kwon, Manjin Kim, Suha Kwak, and Minsu Cho. Learning self-similarity in space and time as generalized motion for action recognition. *arXiv:2102.07092*, 2021. 7

Bin Li, Xi Li, Zhongfei Zhang, and Fei Wu. Spatio-temporal graph routing for skeleton-based action recognition. In *AAAI*, volume 33, pp. 8561–8568, 2019a. 2

Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *CVPR*, pp. 8971–8980, 2018. 18

Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019b. 2, 6

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 *ECCV*, 2014. 4, 5, 14

Zeyi Lin, Wei Zhang, Xiaoming Deng, Cuixia Ma, and Hongan Wang. Image-based pose representation for action recognition and hand gesture recognition. In *FG*, pp. 532–539. IEEE, 2020. 3

Hong Liu, Juanhui Tu, and Mengyuan Liu. Two-stream 3d convolutional neural network for skeleton-based action recognition. *arXiv:1705.08106*, 2017. 3

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. *TPAMI*, 2019. doi: 10.1109/TPAMI.2019.2916873. 2, 3, 5, 13

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv:2106.13230*, 2021. 7, 21

Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *CVPR*, 2020. 2, 6, 8, 18, 20

Diogo C Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *CVPR*, pp. 5137–5146, 2018. 3

Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, pp. 2640–2649, 2017. 18

Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pp. 483–499. Springer, 2016. 4

Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *NeurIPS*, pp. 2277–2287, 2017. 4

Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *CVPR*, 2019. 18

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv:1506.01497*, 2015. 5

Kohei Sendo and Norimichi Ukita. Heatmapping of people involved in group activities. In *ICMVA*, 2019. 6

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *CVPR*, June 2016. 3, 5, 13, 19

Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *CVPR*, pp. 2616–2625, 2020. 2, 5, 6, 14, 20

Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *CVPR*, pp. 7912–7921, 2019a. 6

Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, pp. 12026–12035, 2019b. 2, 6, 7

Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Decoupled spatial-temporal attention network for skeleton-based action recognition. *arXiv:2007.03263*, 2020. 6

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv:1406.2199*, 2014. 1

Yi-Fan Song, Zhang Zhang, Caifeng Shan, and Liang Wang. Richly activated graph convolutional network for robust skeleton-based action recognition. *TSCVT*, 31(5):1915–1925, 2020. 6

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv:1212.0402*, 2012. 2, 5

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014. 8

Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pp. 5693–5703, 2019. 4, 5, 15, 18

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 1, 2, 5, 15

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, pp. 6450–6459, 2018. 2

Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *ICCV*, pp. 5552–5561, 2019. 2

Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *CVPR*, pp. 588–595, 2014. 1

Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, pp. 1290–1297. IEEE, 2012. 1

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 *ECCV*, 2016. 1, 4

Figure 5: **The extracted skeletons of the NTURGB+D dataset.** The actions of the visualized frames are: "cheer up", "touch other person's pocket", "jump up", "put the palms together", "taking a selfie", "shake fist".

Philippe Weinzaepfel, Romain Brégier, Hadrien Combaluzier, Vincent Leroy, and Grégory Rogez. Dope: Distillation of part experts for whole-body 3d pose estimation in the wild. In *ECCV*, 2020. 18

Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018. 4

Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audio-visual slowfast networks for video recognition. *arXiv:2001.08740*, 2020. 1

An Yan, Yali Wang, Zhifeng Li, and Yu Qiao. Pa3d: Pose-action 3d machine for video recognition. In *CVPR*, pp. 7922–7931, 2019. 3, 7, 19

Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, volume 32, 2018. 1, 2, 4, 5, 6, 20, 21

Hao Yang, Dan Yan, Li Zhang, Dong Li, YunDa Sun, ShaoDi You, and Stephen J Maybank. Feedback graph convolutional network for skeleton-based action recognition. *arXiv:2003.07564*, 2020. 6

Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10, 2012. 18, 19

Dingyuan Zhu, Ziwei Zhang, Peng Cui, and Wenwu Zhu. Robust graph convolutional networks against adversarial attacks. In *KDD*, pp. 1399–1407, 2019. 1, 2

A Appendix

A.1 Visualization

We provide more visualization of the extracted pose of the four datasets: FineGYM, NTURGB+D, Kinetics400, Volleyball to demonstrate the performance of the proposed pose extraction approach qualitatively. The videos corresponding to the visualized frames are also provided in supplementary materials.

NTURGB+D (Shahroudy et al., 2016; Liu et al., 2019). Figure 5 displays some examples of extracted skeletons of NTURGB+D. Our pose extractor achieves almost perfect performance on

Figure 6: **The extracted skeletons of the FineGYM dataset.** The extracted skeletons are far from perfect, but discriminative enough for action recognition.

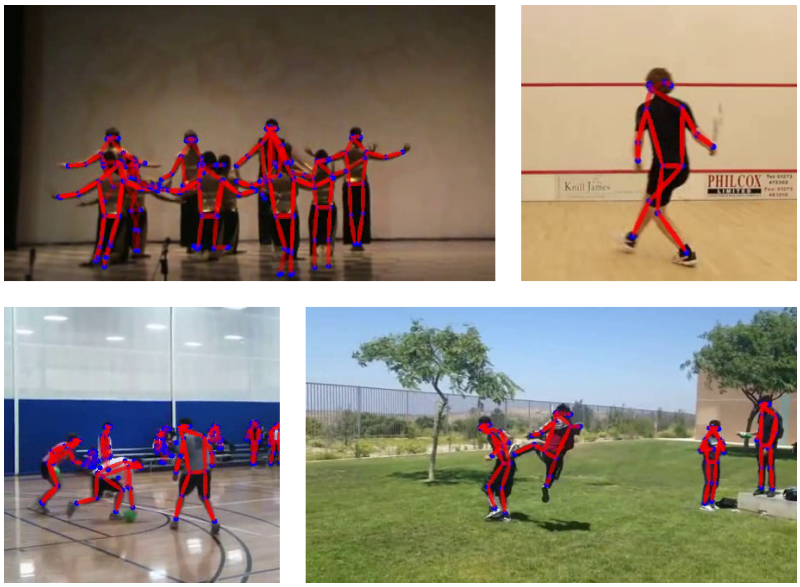


Figure 7: **The extracted skeletons of the Kinetics400 dataset.**

NTURGB+D due to the simple scenarios: the background scene is not complicated, while there are two persons at most in each frame, with little occlusion.

FineGYM (Shao et al., 2020). Figure 6 displays some examples of extracted skeletons of FineGYM. Although we perform pose extraction with ground-truth bounding boxes of the athletes, the extracted 2D poses are far from perfect. The pose extractor is extremely easy to make mistakes for poses the rarely occur in COCO-keypoint (Lin et al., 2014) or when motion blur occurs. Even though the quality of extracted skeletons are not satisfying, they are still discriminative enough for skeleton-based action recognition.

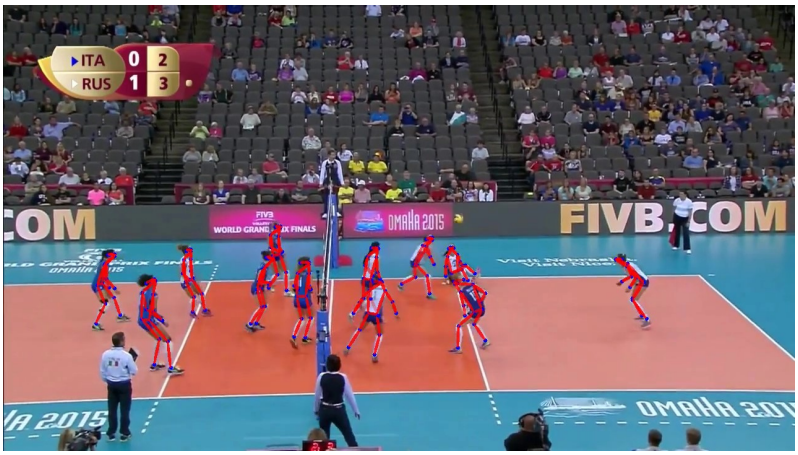


Figure 8: **The extracted skeletons of the Volleyball dataset.**

Kinetics400 (Carreira & Zisserman, 2017). Kinetics400 is not a human-centric dataset for action recognition. In Kinetics videos, the person locations, scales, and the number of persons may vary a lot, which makes extracting human skeletons of Kinetics400 much more difficult than NTURGB+D or FineGYM. In Figure 7, we provide some examples that our pose estimator accurately predicts the human skeletons. We also discuss some failure cases in Sec. A.4.6.

Volleyball (Ibrahim et al., 2016). Volleyball is a group activity recognition dataset. Each frame of a video contains around a dozen people (six for each team). Most of the human poses in a volleyball video are regular ones (unlike FineGYM). In Figure 8, we see that our pose extractor can predict the human pose of each person accurately.

A.2 Illustraction of Generating Pseudo Heatmap Volumes.

In this section, we illustrate how we generate the pseudo heatmap volumes, the input of PoseConv3D. We also provide a jupyter notebook named `GenPseudoHeatmaps.ipynb` in supplementary materials, which can extract skeleton keypoints from RGB videos (optional) and generate pseudo heatmaps based on the skeleton keypoints.

Figure 9 illustrates the pipeline of pose extraction (RGB video $\rightarrow$ coordinate-triplets) and generating pseudo heatmap volumes (coordinate-triplets $\rightarrow$ 3D heatmap volumes). The visualization in Figure 9 is just for one frame, while you can find the visualization for the entire video in the jupyter notebook. Since the heatmaps are of K channels ($K = 17$ for COCO-keypoints), we visualize the heatmap in one 2D image with color encoding. The pose extraction part is straight-forward: we use a Top-Down pose estimator instantiated with HRNet (Sun et al., 2019) to extract the 2D poses for each person in each frame, and save the extracted poses as coordinate-triplets: (x, y, score). For generating pseudo heatmaps, we first perform uniform sampling, which will sample T ($T = 32$ or 48 in experiments) frames uniformly from the video and discard the remaining frames. After that, we will find a global cropping box (The red box in Figure 9, same for all T frames) that envelops all persons in the video, and crop all T frames with that box to reduce the spatial size (as illustrated in Figure 9). In `GenPseudoHeatmaps.ipynb`, you can run the entire pipeline to process a video from the NTURGB-D dataset.

A.3 The Architecture of PoseConv3D and RGBPose-Conv3D

A.3.1 Different variants of PoseConv3D.

In Table 11, we demonstrate the architectures of the three backbones we adapted from RGB-based action recognition as well as their variants:

C3D (Tran et al., 2015). C3D is one of the earliest 3D-CNN developed for RGB-based action recognition (like AlexNet (Krizhevsky et al., 2012) for image recognition), which consists of eight

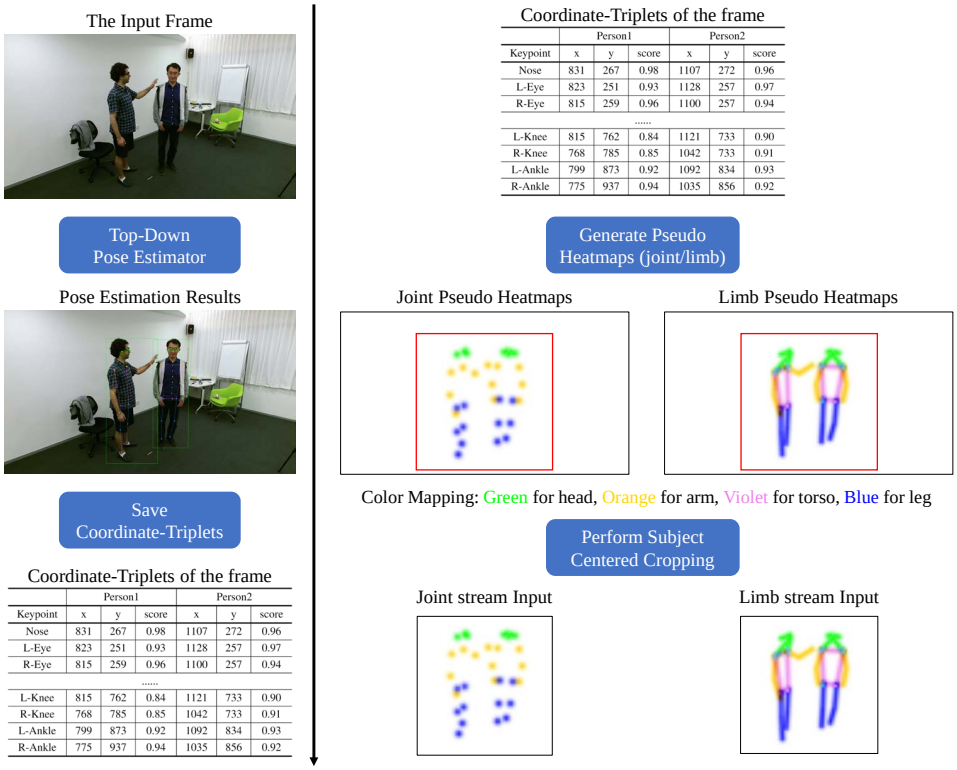


Figure 9: **The pipeline of generating the input of PoseConv3D. Left, Pose Extraction:** We perform Top-Down pose estimation for each single frame. The estimated 2D poses are saved as coordinate-triplets: (x, y, score). **Right, Generating Pseudo Heatmap Volumes:** Based on the coordinate-triplets, we generate pseudo heatmaps for joints and limbs using Eq 1, 2. We perform subjects-centered cropping and uniform sampling to make the heatmap volumes compact.

Table 11: **The architecture of PoseConv3D instantiated with three backbones: C3D, X3D, SlowOnly.** The dimensions of kernels are denoted by $T \times S^2, C$ for temporal, spatial, channel sizes. Strides are denoted with T, S^2 for temporal and spatial strides. GAP denotes global average pooling.

stage	C3D-s	C3D	X3D-s	X3D	SlowOnly	SlowOnly-wd	SlowOnly-HR
data layer	Uniform 48, 56 ×56						Uniform 48, 112 ×112
stem layer	conv 3×3^2, 32		conv 1×3^2, 24 stride 1, 2^2 conv 5×1^2, 24		conv 1×7^2, 32	conv 1×7^2, 64	conv 1×7^2, 32
stage1	maxpool 1×2^2 $[3\times3^2, 64]\times1$		$\begin{bmatrix}1\times1^2, 54\\3\times3^2, 54\\1\times1^2, 24\end{bmatrix}\times2$	$\begin{bmatrix}1\times1^2, 54\\3\times3^2, 54\\1\times1^2, 24\end{bmatrix}\times5$	None		$\begin{bmatrix}1\times1^2, 32\\1\times3^2, 32\\1\times1^2, 128\end{bmatrix}\times3$
stage2	maxpool 1×2^2 $[3\times3^2, 128]\times2$		$\begin{bmatrix}1\times1^2, 108\\3\times3^2, 108\\1\times1^2, 48\end{bmatrix}\times5$	$\begin{bmatrix}1\times1^2, 108\\3\times3^2, 108\\1\times1^2, 48\end{bmatrix}\times11$	$\begin{bmatrix}1\times1^2, 32\\1\times3^2, 32\\1\times1^2, 128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1^2, 64\\1\times3^2, 64\\1\times1^2, 256\end{bmatrix}\times4$	
stage3	maxpool 1×2^2 $[3\times3^2, 256]\times2$		$\begin{bmatrix}1\times1^2, 216\\3\times3^2, 216\\1\times1^2, 96\end{bmatrix}\times3$	$\begin{bmatrix}1\times1^2, 216\\3\times3^2, 216\\1\times1^2, 96\end{bmatrix}\times7$	$\begin{bmatrix}\underline{3\times1^2}, 64\\1\times3^2, 64\\1\times1^2, 256\end{bmatrix}\times6$	$\begin{bmatrix}\underline{3\times1^2}, 128\\1\times3^2, 128\\1\times1^2, 512\end{bmatrix}\times6$	
stage4	None	$[3\times3^2, 256]\times2$	conv 1×1^2, 216		$\begin{bmatrix}\underline{3\times1^2}, 128\\1\times3^2, 128\\1\times1^2, 512\end{bmatrix}\times3$	$\begin{bmatrix}\underline{3\times1^2}, 256\\1\times3^2, 256\\1\times1^2, 1024\end{bmatrix}\times3$	
	GAP, fc						

3D convolution layers. To adapt C3D for skeleton-based action recognition, we reduce its channel-width to half ($64 \rightarrow 32$) for better efficiency. In addition, for Pose-C3D-s, we remove the last two convolution layers.

X3D (Feichtenhofer, 2020). X3D is a recent state-of-the-art 3D-CNN for action recognition. Replacing vanilla convolutions with depth-wise convolutions, X3D achieves competitive recognition

Table 12: **RGBPose-Conv3D instantiated with the SlowOnly backbone.** The dimensions of kernels are denoted by $T \times S^2, C$ for temporal, spatial, channel sizes. Strides are denoted with T, S^2 for temporal and spatial strides. The backbone we use is ResNet50. GAP denotes global average pooling.

stage	*RGB* Pathway	*Pose* Pathway	output sizes T×S^2
data layer	uniform 8,1^2	uniform 32,4^2	*RGB*: 8×224^2 *Pose*: 32×56^2
stem layer	conv 1×7^2, 64 stride 1, 2^2 maxpool 1×3^2 stride 1, 2^2	conv 1×7^2, 32 stride 1, 1^2	*RGB*: 8×56^2 *Pose*: 32×56^2
res$_2$	$\begin{bmatrix} 1\times1^2, 64 \\ 1\times3^2, 64 \\ 1\times1^2, 256 \end{bmatrix}\times3$	N.A.	*RGB*: 8×56^2 *Pose*: 32×56^2
res$_3$	$\begin{bmatrix} 1\times1^2, 128 \\ 1\times3^2, 128 \\ 1\times1^2, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1^2, 32 \\ 1\times3^2, 32 \\ 1\times1^2, 128 \end{bmatrix}\times4$	*RGB*: 8×28^2 *Pose*: 32×28^2
res$_4$	$\begin{bmatrix} 3\times1^2, 256 \\ 1\times3^2, 256 \\ 1\times1^2, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 3\times1^2, 64 \\ 1\times3^2, 64 \\ 1\times1^2, 256 \end{bmatrix}\times6$	*RGB*: 8×14^2 *Pose*: 32×14^2
res$_5$	$\begin{bmatrix} 3\times1^2, 512 \\ 1\times3^2, 512 \\ 1\times1^2, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 3\times1^2, 128 \\ 1\times3^2, 128 \\ 1\times1^2, 512 \end{bmatrix}\times3$	*RGB*: 8×7^2 *Pose*: 32×7^2
	GAP, fc	GAP, fc	# classes

performance with tiny amounts of parameters and FLOPs. The architecture of the adapted Pose-X3D is almost unchanged compared to the original X3D-S, except that we remove the original first stage. For Pose-X3D-s, we remove convolution layers from each stage uniformly by changing the hyper-parameter γ_d from 2.2 to 1.

SlowOnly (Feichtenhofer et al., 2019). SlowOnly is a popular 3D-CNN used for RGB-based action recognition. It is obtained by inflating the ResNet layers in the last two stages from 2D to 3D. To adapt SlowOnly for skeleton-based action recognition, we reduce its channel-width to half ($64 \rightarrow 32$) as well as remove the original first stage in the network. We also have conducted experiments with Pose-SlowOnly-wd (with channel-width 64) and Pose-SlowOnly-HR (with 2x larger input and deeper network). There is no performance improvement despite the much heavier backbone.

A.3.2 RGBPose-Conv3D instantiated with SlowOnly.

RGBPose-Conv3D is a general framework for RGB-Pose dual-modality action recognition, which can be instantiated with various 3D-CNN backbones. In this work, we instantiate both pathways with the SlowOnly network. As shown in Table 12, the RGB pathway has a smaller frame rate and a larger channel width since RGB frames are low-level features. On the contrary, the Pose pathway has a larger frame rate and a smaller channel width. Time stride convolutions are used as bi-directional lateral connections between the two pathways (after res$_3$ and res$_4$) so that semantics of different modalities can sufficiently interact. Besides lateral connections, the predictions of two pathways are also combined in a late fusion manner, which leads to further improvements in our empirical study. RGBPose-Conv3D is trained with two individual losses respectively for each pathway, as a single loss that jointly learns from two modalities leads to severe overfitting.

A.4 Supplementary Experiments

A.4.1 Ablation Study on Pose Extraction

This section discusses different alternatives that can be adopted in pose extraction to validate our choice. The input size for all 3D-CNN experiments is $T \times H \times W = 48 \times 56 \times 56$.

2D v.s. 3D Skeletons. We first compare the recognition performance of using 2D and 3D skeletons for action recognition. The 3D skeletons are either collected by sensors (NTU-60) or estimated

Table 13: **Ablation study on Pose Extraction.**

Input	GYM	NTU-60
Kinect-3D (Zhang, 2012)	N.A.	89.4
DOPE-3D (Weinzaepfel et al., 2020)	76.3	N.A.
VIBE-3D (Kocabas et al., 2020)	87.0	N.A.
HRNet-2D (Sun et al., 2019)	**92.0**	**91.9**
MobileNet-2D (Howard et al., 2017)	**89.0**	**90.2**

(a) **2D skeleton v.s. 3D skeleton.**

Input	GYM
DOPE (Weinzaepfel et al., 2020)	76.3
VIBE (Kocabas et al., 2020)	87.0
FrameLift (Martinez et al., 2017)	90.0
VideoLift (Pavllo et al., 2019)	90.2
HRNet-2D (Sun et al., 2019)	92.0

(b) **3D-pose from a 'lifting' model doesn't help in recognition.**

Model	COCO AP	NTU-60
HRNet (Top-Down)	**0.746**	**93.6**
HRNet (Bottom-Up)	0.654	93.0
Mobile (Top-Down)	0.646	92.0

(c) Top-Down v.s. Bottom-Up approaches for pose estimation.

Proposals	GYM Mean-Top1
Detection	75.8
Tracking	85.3
GT	**92.0**

(d) **Pose extracted with different boxes.**

Input	GYM Mean-Top1
Coordinate-MobileNet	90.7
Coordinate-HRNet	93.2
Heatmap-MobileNet	92.7
Heatmap-HRNet	93.6

(e) **Coordinate v.s. Heatmap.**

with state-of-the-art 3D pose estimators based on RGB inputs (Weinzaepfel et al., 2020; Kocabas et al., 2020) (FineGYM). For a fair comparison, we use MS-G3D (Liu et al., 2020) (the current state-of-the-art GCN for skeleton-based action recognition) with the same configuration and training schedule for 2D and 3D keypoints and list the results in Table 13a. The estimated 2D keypoints (even low-quality ones) consistently outperform 3D keypoints (sensor collected or estimated) in action recognition. Besides RGB-based 3D-pose estimators, we also consider the 'lifting' approaches (Martinez et al., 2017; Pavllo et al., 2019), which directly 'lift' 2D-pose (sequences) to 3D-pose (sequences). We regress the 3D poses based on 2D poses extracted with HRNet, use the lifted 3D poses for action recognition. The results in Table 13b indicate that such lifted 3D poses do not provide any additional information, performs even worse than the original 2D poses in action recognition.

Bottom-Up v.s. Top-Down. To compare the pose estimation quality of Bottom-Up and Top-Down approaches, we instantiate the two approaches with the same backbone (HRNet-w32). Besides, we also instantiate the Top-Down approach with the MobileNet-v2 backbone for comparison, which has a similar performance to HRNet (Bottom-Up) on COCO-validation. We use extracted 2D poses to train a *Pose-SlowOnly* on NTU-60. Table 13c shows that the performance of HRNet (Bottom-Up) on COCO-val is much worse than HRNet (Top-Down) and close to MobileNet (Top-Down). However, the Top-1 accuracy of HRNet (Bottom-Up) is much higher than MobileNet (Top-Down) and close to HRNet (Top-Down). Although the potential of Bottom-Up should not be neglected, considering the better performance and faster inference speed (Top-Down runs faster when there aren't many persons in a frame), we use Top-Down for pose extraction in this work.

Interested Person v.s. All Persons. Many people may exist in a video, but not all of them are related to the interested action. For example, in FineGYM, only the pose of the athlete is helpful, while other persons like the audience or referee are unrelated. We compare using 3 kinds of person bounding boxes for pose extraction: **Detection**, **Tracking**(with Siamese-RPN (Li et al., 2018)) and **GT** (with increasing prior about the athlete). In Table 13d, we see that the prior of the interested person is extremely important: even weak prior knowledge (1 GT box per video) can improve the performance by a large margin.

Coordinates v.s. Heatmaps. Storing 3D heatmap volumes may take vast amounts of disk space. To be more efficient, we can save the 2D poses as coordinate-triplets (x, y, score) and restore them to 3D heatmap volumes following the methods we introduced in Sec. 3.2. We conduct experiments on FineGYM to explore how much information is lost during the heatmap $\rightarrow$ coordinate compression. In Table 13e, we see that for low-quality pose estimators, it leads to a 2% drop in Mean-Top1. For high-quality ones, the degradation is more moderate (only a 0.4% Mean-Top1 drop). Thus we choose to store coordinates instead of 3D heatmap volumes.

Table 14: **Transferring Ability.** Skeleton representations learned on the large-scale Kinetics400 can transfer to downstream datasets well. Backbone parameters are frozen for the 'Linear' setting.

PoseConv3D	HMDB51	UCF101
Scratch	58.6	79.1
Linear	64.9	83.1
Finetune	**69.3**	**87.0**

Table 15: **Comparison with state-of-the-art multi-modality action recognition approaches.**

	HMDB51	UCF101
I3D (Carreira & Zisserman, 2017)	80.7	98.0
PoTion (Choutas et al., 2018)	43.7	65.2
PoTion + I3D	80.9	98.2
PA3D (Yan et al., 2019)	55.3	-
PA3D + I3D	82.1	-
PoseConv3D	**69.3**	**87.0**
PoseConv3D + I3D	**82.7**	**98.4**

A.4.2 Multi-Modality Results Action Recognition on UCF101 and HMDB51

In Table 5, we train different PoseConv3D on UCF101 and HMDB51 from scratch. In this section, we demonstrate that PoseConv3D can also take advantage of pretraining on large-scale datasets. We adopt weights pretrained on Kinetics400 to initialize the PoseConv3D. Pretraining with skeleton data from the large-scale Kinetics400 benefits the downstream recognition tasks on smaller datasets, under both 'Linear' and 'Finetune' paradigms (Table 14).

We further compare PoseConv3D with previous state-of-the-arts of skeleton-based action recognition on UCF101 and HMDB51: PoTion (Choutas et al., 2018) and PA3D (Yan et al., 2019). For a fair comparison, we fuse the skeleton-based predictions with I3D (Carreira & Zisserman, 2017) predictions, instead of predictions from the more advanced OmniSource (Duan et al., 2020). Table 15 shows that PoseConv3D not only outperforms other approaches by a large margin on skeleton-based action recognition, but also leads to better overall performance after fusing with predictions based on other modalities.

A.4.3 Using 3D Skeletons in PoseConv3D

PoseConv3D takes stacked 2D skeleton keypoint heatmaps as input. Assume only 3D skeletons are available for a target dataset, one can also use the 3D skeletons in PoseConv3D by projecting them to a 2D plane. The NTURGB+D dataset (Shahroudy et al., 2016) provides 3D skeleton sequences collected by Microsoft Kinect v2 sensors (Zhang, 2012). Besides, the dataset also includes the projection of 3D joints onto the 2D image coordinate systems. We use the projected 2D skeletons of NTU-60 as the input for PoseConv3D and study the effect.

Table 16 demonstrates the recognition performance of using projected 2D skeletons in PoseConv3D. Using the projected 2D skeletons as inputs instead of the original 3D skeletons, there is a 2% Top-1 accuracy drop for MS-G3D due to the information lost in 3D $\rightarrow$ 2D compression. If both use 2D skeletons as input, PoseConv3D outperforms the GCN-based counterpart by 2.4%, even surpasses the MS-G3D with 3D skeletons as input by 0.4%, which indicates the great spatiotemporal modeling capability of 3D-CNN can compensate for the information lost in 3D $\rightarrow$ 2D projection.

A.4.4 Uniform Sampling for RGB-based recognition

Based on the outstanding improvement by uniform sampling on skeleton-based action recognition, we wonder if this sampling strategy also works for RGB-based action recognition. Thus we apply uniform sampling to RGB-based action recognition on NTU-60 (Shahroudy et al., 2016) and

[5]We rerun the official code of MS-G3D to get this accuracy.

Table 16: **PoseConv3D with projected 2D poses.** We report the recognition performance of the joint model.

Method	Top-1
2D-projection + MS-G3D (Liu et al., 2020)	86.8
3D-skeleton + MS-G3D (Liu et al., 2020)	88.8[5]
2D-projection + PoseConv3D	**89.2**

Table 17: **Uniform sampling also works for RGB-based action recognition.** Alls results are for 10-clip testing, except the 'uniform-16 (1c)', which uses 1-clip testing.

(a) **FineGYM.**

Sampling	Mean-Top1
16x2	87.9
16x4	88.7
uniform-16 (1c)	**91.1**
uniform-16	**91.6**

(b) **NTU-60 (X-Sub)**

Sampling	Top1
16x2	94.9
16x4	95.1
uniform-16 (1c)	**95.7**
uniform-16	**96.1**

GYM (Shao et al., 2020). We use SlowOnly-R50 (Feichtenhofer et al., 2019) as the backbone and set the input length as 16 frames. From Table 17, we see that uniform sampling also outperforms fix-stride sampling by a large margin in RGB-based recognition on these two datasets: the accuracy of uniform sampling with 1-clip testing is better than the accuracy of fix-stride sampling with 10-clip testing. We mainly attribute the advantage of uniform sampling to the highly variable video lengths in these two datasets. On the contrary, we observe a slight accuracy drop on Kinetics400[6] when applying uniform sampling: for SlowOnly-R50 with input length 8, the Top-1 accuracy drops from 75.6% to 75.2%.

A.4.5 NTU-60 Error Analysis

On NTU-60 X-Sub split, we achieve 94.1% Top-1 accuracy with skeleton-based action recognition, which outperforms the current state-of-the-art result by 2.6%. To further study the failure cases, we first define the confusion score s of a pair of the action classes i, j as:

$$s = n_{ij} + n_{ji} \tag{3}$$

n_{ij} indicates the number of videos belong to the class i but recognized as class j. In NTU-60, there are 1770 pairs of action classes in total, while we list the five most confusing pairs in Table 18. Most failure cases are of these top-confusing pairs, *e.g.*, over 27% failure cases are of the top 5 confusion pairs. It is hard to distinguish these pairs of actions with human skeletons only.

Some confusing pairs can be resolved by exploiting other modalities such as RGB appearance. If the model successfully recognizes the keyboard, then it can distinguish typing from writing. Table 18 shows that, with multi-modality fusion in *RGBPose*-Conv3D, the recognition performance on those confusing pairs improves a lot.

A.4.6 Why skeleton-based pose estimation performs poorly on Kinetics400

PoseConv3D with high-quality 2D skeletons improves the Top-1 accuracy of skeleton-based action recognition on Kinetics400 from 38.0% to 47.7%. However, the accuracy on Kinetics400 is still far below the accuracies on other datasets. Besides the difficulties mentioned in Sec. A.1, two more problems will degrade the quality of extracted skeleton sequences (Figure 10): 1. Since Kinetics400 is not human-centric, human skeletons are missing or hard to recognize in many frames. 2. For the same reason, only small parts of humans appear in many frames, while the pose estimators are easy to fail in this scenario.

We also report the mean class accuracy on Kinetics-Motion (Yan et al., 2018) in Table 19, which contains 30 action classes in Kinetics that are strongly related to body motions. The accuracy of skeleton-based action recognition is much higher on this subset, increasing from 47.7% to 81.9%.

[6] In Kinetics400, most video clips are of the same temporal length: 10 seconds.

Table 18: **Top 5 confusion pairs of skeleton-based action recognition on NTU-60 X-Sub.** Multi-modality fusion with *RGBPose*-Conv3D improves the recognition performance on confusion pairs by a lot.

Action1	Action2	score [Pose]	score [RGB + Pose]
Read	Play with phone/tablet	67	13
Write	Type on a keyboard	57	20
Write	Play with phone/tablet	50	5
Take a selfie	Point to sth. with finger	48	10
Read	Write	44	24

Figure 10: **Problems in Kinetics400 Pose Extraction.** Left: Human missing in action 'kayaking'. Middle: Human skeleton is too small to be recognized in action 'diving cliff'. Right: Only human parts appear, the pose estimator fails ('washing feet').

Test Set	Kinetics-Motion
Swin-L (Liu et al., 2021)	92.7
ST-GCN (Yan et al., 2018)	72.0
PoseConv3D	81.9
Swin-L + PoseConv3D	**94.7**

Table 19: **Mean class accuracy on the Kinetics-Motion subset.**

When combined with the state-of-the-art RGB predictions, the improvement is much more significant, increasing from 0.6% to 2.0%. However, the skeleton-based performance is still far behind the state-of-the-art RGB-based action recognition method (Liu et al., 2021), which achieves 92.7% mean class accuracy on Kinetics-Motion. The inferior recognition performance indicates that there still needs more future work for skeleton-based action recognition in the wild.