Technical Report of Challenge on Kinetics-TPS Track on Part-level Action Parsing and Action Recognition

Jiawei Dong[1], Yuliang Chen[2], Shuo Wang[1]

[1] Shanghai Paidao Intelligent Technology Co., Ltd.
{Jiawei.Dong, Shuo.Wang}@ai-prime.ai
[2] Chongqing Jiaotong University
yuliang.chen@mails.cqjtu.edu.cn

Abstract. This is a technical report on Kinetics-TPS track. The report focuses on the dataset usage, method framework, model selection, training process, and inference process of our methods. In order to achieve part-level detection and action recognition, we propose the following four methods: video-category-level method, video-level method, segment-level method and instance-level method, the method based on person-context-person modeling showed very promising results, thus demonstrating the existence of high-order relation in videos. The proposed ensemble method based on multiple heterogeneous models is also proved to expand the capacity of the our methods.

Keywords: Action recognition, spatio-temporal localization, instance-level, person-context-person relation

1 Introduction

Action recognition has been treated as a high-level video classification problem. However, such manner ignores detailed and middle-level understanding about human actions.

The Kinetics-TPS benchmark is a large-scale dataset encoding human actions as spatio-temporal composition of body parts. Different from existing video action datasets, Kinetics-TPS provides 7.9M annotations of 10 body parts, 7.9M part state (i.e., how a body part moves) and 0.5M interactive objects in the video frames of 24 human action classes, which bring new opportunity to understand human action by compositional learning of body parts.

In this challenge, we propose several action recognition method for part-level detection and spatio-temporal localization, these methods are: video-category-level method, video-level method, segment-level method and instance-level method, all the methods are sharing the same object detector and video-level action recognition network, each method has it own design of the part state recognition block. Among these methods, our score of instance-level method can reach up to 0.6624 on Leaderboard, which is our best single method.

In order to ensemble multiple results of heterogeneous methods, we propose the ensemble method based on IoU and voting. In order to improve the overall capacity of our methods, we propose an ensemble method based on video-category. For all categories of videos, we can always find a method that is most suitable for the video category. With above ensemble methods, our score can reach up to 0.7389 on Leaderboard, outperforms all participants by considerable margins.

2 Data Preprocessing

According to the terms and condition of the challenge, we only use the provided Kinetics-TPS dataset for training without any extra dataset. Our data preparation mainly includes frame extraction, data augmentation and sampling methods.

2.1 Frame Extracting

We extract 574851 labeled frames from 3809 training set videos, extract 48655 frames from 932 testing set videos with 5 frames interval. The extracted frame images retain the original resolution.

2.2 Data Augmentation

On object detection, the data augmentation methods we used mainly include mixup, mosaic, rotation, perspective, scale, shear. On action recognition network, the data augmentation methods we used mainly include rotation and scale.

In particular, we considered that the task required to distinguish the left and right parts of the human body, so when horizontal flipping is used, we needed to swap the label with "left" and "right". For example, if the bounding box label is "right_arm", after horizontal flipping, bounding box with label is converted to "left_arm", and "left_arm" is converted to "right_arm" in the same way, as shown in Figure 1. We define this operation as "label swap".

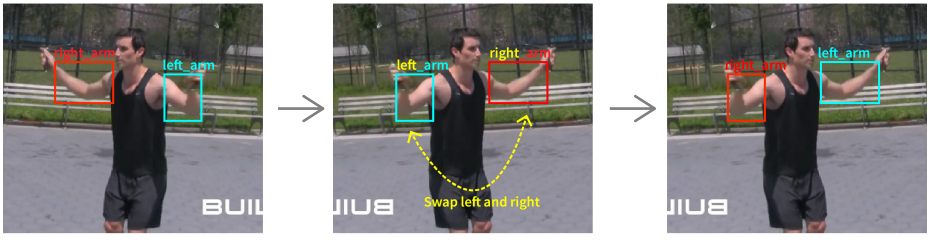

Fig. 1. Horizontal flipping and label swap

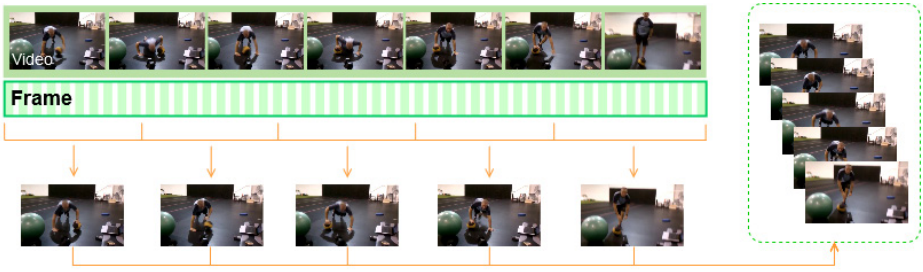

Fig. 2. Uniform sample

Fig. 3. Dense sample

2.3 Uniform Sample

The definition of uniform sampling is, when n frames is required for sampling from a video, the video is divided into n segments of equal length, for each segment, there is only one frame is sampled in random position, which is shown in Figure 2.

The advantage of this sampling method is, no matter how long the video duration is, uniform sampling can avoid missing key information. The disadvantage is, the sampled frames may lack continuous information for videos with long video duration or short duration of key actions.

2.4 Dense Sample

The definition of dense sampling is, for one video, we sample a segment with fixed length, and the length of this segment is determined by the number of sampling frames and frame interval. For each segment, the label of start frame or middle frame will be used as the label of the segment, and we used padding for the beginning and end of the video, which is shown in Figure 3.

This sampling method can strengthen the recognition of action with short duration. All frames in the segment have strong temporal information due to their small frame interval. The disadvantage is, the number of sampled frames directly affects the performance of action recognition network, which requires manual adjustment.

3 Method

The methods we used are mainly composed of three parts: human and body parts detection, video action recognition and part state recognition. All the methods share the same detection and video action recognition block, the only difference between methods is part state recognition block.

3.1 Human and Body Parts Detection

First, we train a object detector with total of 11 classes of human and human body parts, but we found that in post-processing, this method inevitably requires a process of assigning parts to people by IoU. When the number of people in the video is large, with the horizontal angle of view, this method will assign body parts to wrong person, which had a huge impact on the score.

We propose a two-stage detection structure. First we train a detector that only detects the human body, when human body is detected, we crop the RGB image of the person according to the person's bounding box, and pass it to the second detector, which only detects human body part of 10 classes. At last, the results of two stages are merged. This method bypasses the process of assigning parts, and has a high accuracy even in the case of a large number of people in the video.

The object detector we used is YOLOv7-X [1], we train for 50 epochs with batch size 128, image resolution 640, base learning rate 0.01, one-cycle scheduler and SGD optimizer. The pre-trained model we used is the official released pre-trianed model on COCO dataset [2].

3.2 Video Action Recognition

For the recognition of 24 categories of videos, we mainly use common action recognition networks. The main process is, uniform sampling the video, and pass the sampled segment into the action recognition network to get the predicted category of the video.

The action recognition network we used is Video Swin Transformer [3]. We train for 80 epochs with batch size 2, segment length 32, video resolution 360, base learning rate 0.0003, one-cycle scheduler and AdamW optimizer. We used the ImageNet [4] pre-trained Swin Transformer as its backbone. On local validation, the top1 accuracy of action recognition network can reach up to 99%.

3.3 Part State Recognition

Action recognition of human body parts is undoubtedly the most critical step of this challenge. According to the fine-grained level, from low to high, we propose video-category-level, video-level, segment-level and instance-level method.

Video Category Level. Considering high correlation between the category of video and the state of body part, taking advantage of this characteristic, we propose the method based on the video category.

As shown in Figure 4 is the overall structure of the video-category-level method. First, we count the part state in each category, and obtain the most frequently occurring part state in each video category. Such as, In the category "belly_dancing", the most frequently used part states are: Left_arm_bend, right_arm_bend, hip_turn, right_foot_step_on, left_foot_step_on, right_hand_none, left_hand_none, right_leg_step, left_leg_step and head_none. Second, For a given video, according to the predicted video category , the most frequently occurring part states of the video category are assigned to the part states of each person in each frame of the video. Using this method, our score on leaderboard can reach up to **0.4834**.

The advantage of this method is that no extra action recognition model is required, and the part state can be obtained directly based on statistics. The disadvantage is, it is a statistics-based method and depends on the long-tailed distribution of the data, and the granularity of prediction is very low.

Video Level. To increase the granularity of predictions, we proposed the video-level method which can predict the part state by video.

As shown in Figure 5 is the overall structure of the video-level method. First, we count the most frequently occurring part state of each part of each video in training data set. Second, for each video, we use the most frequently occurring part state of each part as its labels for training. For example, There are 34 left_arm_bend, 5 left_arm_unbend, 30 right_arm_bend, 10 left_arm_unbend and 50 hip_turn in video "0001.mp4". Following above operation, this video is labeled as left_arm_bend, right_arm_bend and hip_turn. We transformed this task into a multi-label video classification task. Third, after training, for single testing video, we assign the predicted label to each human of each frame in this video.

The multi-label action recognition network we used is Video Swin Transformer. We train for 80 epochs with batch size 2, labels num 108, segment length 32, video resolution 320, base learning rate 0.0003, one-cycle scheduler and

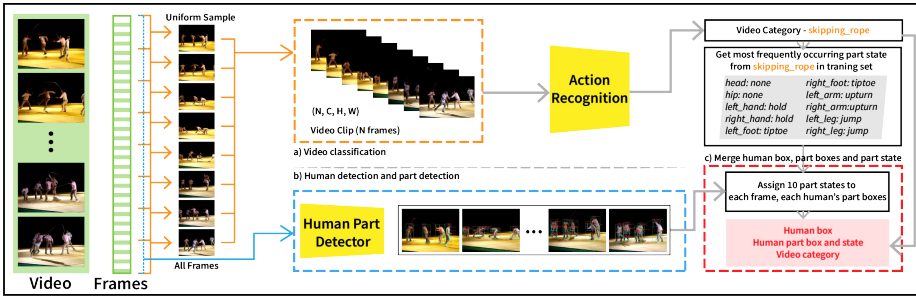

Fig. 4. Category-level method

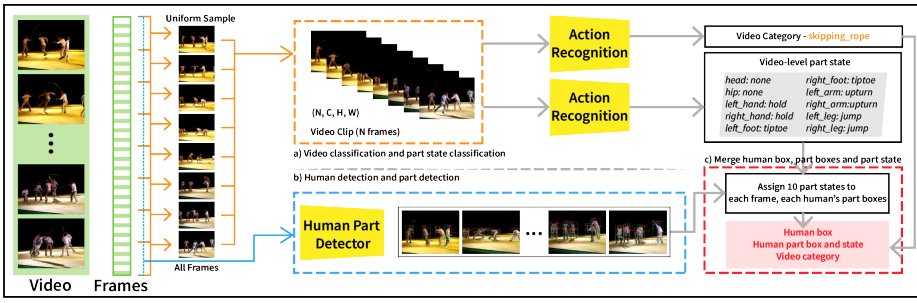

Fig. 5. Video-level method

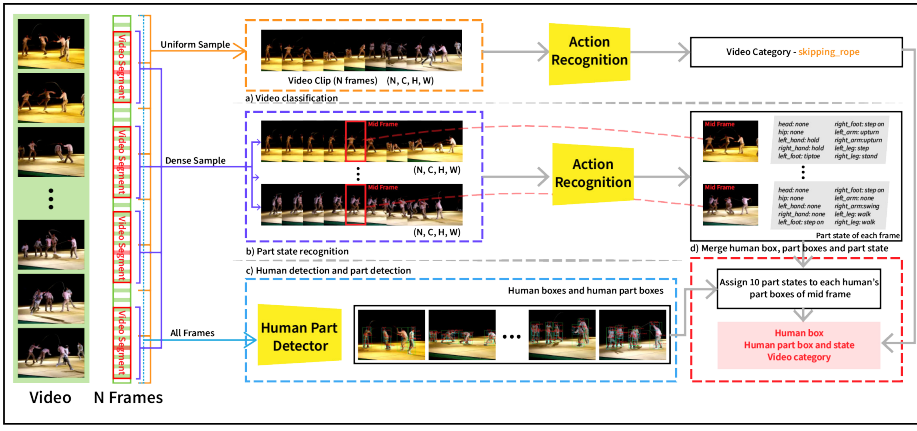

Fig. 6. Segment-level method

AdamW optimizer. We used something-something-v2 [5] pre-trained model for training. Using this method, our score on leaderboard can reach up tp **0.5911**.

Segment Level. As shown in Figure 6 is the overall structure of the segment-level method. For each frame in video, taking the frame as key frame for dense sampling, converting video into a series of continuous segment, the label of the frame is also used as the label of the segment for the training of multi-label action recognition network. For single video, we can get frame-level predictions from continuous dense sampled segment, and we assign the frame-level prediction to each human in this frame.

The multi-label action recognition network we used is ir-CSN [6]. We train for 80 epochs with batch size 2, labels num 108, segment length 32, video resolution 320, base learning rate 0.000256, one-cycle scheduler and AdamW optimizer. We used IG-65M [7] pre-trained model for training. Using this method, our score on leaderboard can reach up tp **0.5600**.

Table 1. Segment-level method experiment results.

Model	Backbone	Segment Length	Lr	Epoch	Leaderboard Score
ir-CSN	ResNet3dCSN	16	5.12E-04	58	0.549715
ir-CSN	ResNet3dCSN	32	5.12E-04	58	-
ir-CSN	ResNet3dCSN	32	2.56E-04	58	**0.560093**

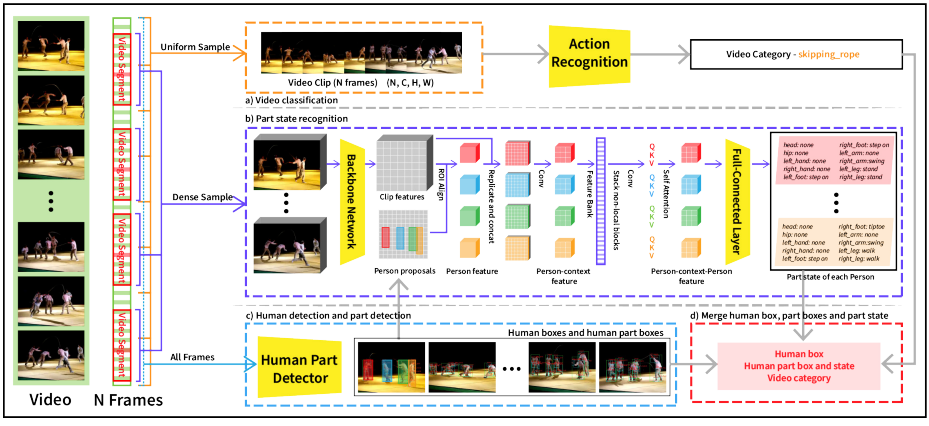

Fig. 7. Instance-level method

Instance Level. In order to further improve the fine granularity of the method, we propose a general method capable of instance-level prediction.

As shown in Figure 7 is the overall structure of the instance-level method. After getting person bounding box from detector, we extract person features from the context features by RoIAlign, then we replicate and concatenate each person feature to all spatial locations of concatenated features, and person-context feature is encoded by applying convolutions to concatenated features.

The next step is to learn high-order relations between pairs of person-context. Inspired by ACAR-Net [8], the operator is modeled as stacking several modified

Table 2. Instance-level method experiment results.

Relation Model	Backbone	Segment Length	Epoch	Threshold	Leaderboard score
Person-Person	Slowfast-Resnet101	16	3	0.1	0.395823
	Slowfast-Resnet101	16	4	0.1	0.554853
	Slowfast-Resnet101	16	5	0.1	**0.558903**
	Slowfast-Resnet101	16	6	0.1	0.554733
Person-Context-Person	Slowfast-Resnet101	16	6	0.1	0.620262
	Slowfast-Resnet101	32	6	0.1	0.623791
	Slowfast-Resnet101	32	6	0.1	0.626608
	Slowfast-Resnet101	32	6	0.01	**0.662429**

non-local blocks. For each non-local block, convolutions are used to convert the input person-context feature into query Q, key K and value V embeddings of the same spatial size as person-context feature. The attention vectors are computed separately at every spatial location, and the person-context-person relation feature is given by the linear combination of all value features according to their corresponding attention weights. After the person-context-person features are obtained, a fully-connected layer is used as simple action classifier to output the confidence scores of each person with different part state.

The backbone for feature extracting is Slowfast-Resnet101 pre-trained on AVA 2.2 [9] dataset. We train for 6 epochs with batch size 1, labels num 164, segment length 32, video resolution 320, base learning rate 0.008, StepLR scheduler and SGD optimizer. Using this method, by lowering the threshold of input bounding box, our score on leaderboard can reach up tp **0.6624**.

We have also experimented another method which is focusing on modeling person-person relation, inspired by AIA[10]. Similar to the above method, we only need to replace the person-context-person module in the part state recognition part with person-person module. The testing results based on person-person modeling are relatively general, which may be due to the following reasons: first, for this challenge, some highly interactive actions, such as skipping rope, ignoring the high-order person-context-person relation may not be appropriate. Second, original AIA has person-object relation modeling, which relies on the extra object detector, and there are relatively few bounding boxes of labeled object in the dataset. Therefore, in this challenge, we can only train and predict part state through person-person features. Third, this method relies on the results of person tracking. During training, it is necessary to provide the track ids of all people in the video to perform accurate person-person modeling. However, when the number of people in the video is large and the video resolution is low, it is difficult to guarantee the accuracy of tracking, so training and inferring based on track results are inherently unreliable.

Instance Level - One-stage. In our experiments, we want to check the importance of temporal information in this challenge, therefore, we propose a method

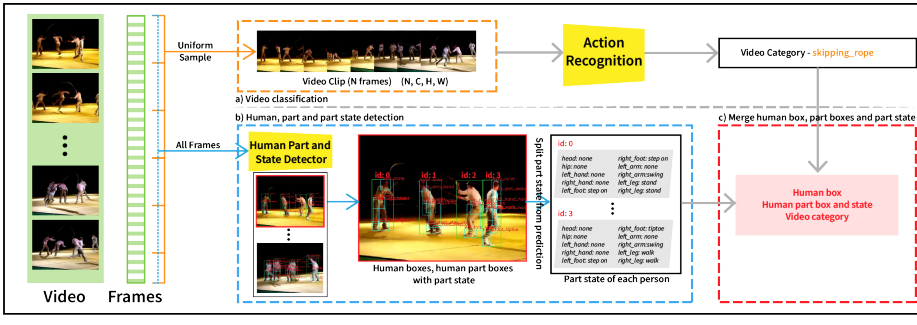

Fig. 8. Instance-level one-stage method

which can output instance-level prediction based on single frame with out any temporal information.

As shown in Figure 8 is the overall structure of the instance-level method. In training, we concatenate the part name and part state into a new label, for example, if the part state of "left_arm" in for a human is "Bend", then the label of this box is converted to "left_arm_bend". In inferring, after getting predictions, we can easily split part name and part state form the predicted labels of bounding box.

The object detector we used is YOLOv7-X, we train for 50 epochs with batch size 128, image resolution 640, base learning rate 0.01, one-cycle scheduler and SGD optimizer. The pre-trained model we used is the official released pre-trianed model on COCO dataset. Using this method, our score on leaderboard can reach up to **0.6597**. We infer that, benefited from the long-tailed distribution of the dataset, even if the temporal information is discarded, simple object detection network can still learn the part state that best matches the current moment based on the scene and person feature of the video.

3.4 Ensemble and Postprocessing

At last, we need to ensemble the results of all methods. The ensemble process mainly includes the following two steps.

Ensemble by Voting. We use the bounding box predicted by the method with the highest score on the Leaderboard as the output bounding box, and traverse the bounding boxes of all people in all frames under all methods. If the IoU between the bounding boxes of all methods is larger than 0.8, it is assumed that, these boxes from multiple methods are referring to the same person. Then we enter the voting stage. For this person, count the state of each part predicted by different methods, and take the part state with the largest count number as the part state of our ensemble result. Using this ensemble method, our score on Leaderboard can reach up to **0.6824**.

Ensemble by Video Category. We calculate the Part State Correctness(PSC) scores of all videos on the local training set with all methods (including the ensemble method above), and then we take the category of the video as a group to count the average PSC score of each method in each category, so as to get the most suitable method for each video category. At last, on the test set, according to the predicted video categories, we assign the prediction of the method which is most suitable for this video category to video. Using this ensemble method, our score on Leaderboard can reach up to **0.7389**, which is our score of final submission.

4 Conclusion and Future Work

In this report, we propose several general methods for part state recognition, here is our conclusions.

To improve detection performance, we use two detectors to detect people and parts separately, thus bypassing the process of assigning parts to people. And in the training, we also proposed a data augmentation method - label swap, which can flip horizontally without affecting the network's discrimination of the left and right directions of the parts.

For the fine-grained improvement of part state prediction, we propose methods at the video-category-level, video-level, segment-level and instance-level. During our research on instance-level method, we also verified that the modeling of person-context-person relation can effectively improve the network's ability to recognize complex actions, which is more efficient than traditional person-context and person-person modeling. It is robust and does not need to rely on external object detectors and person trackers. We also found that although temporal information is considered to be critical in part state recognition, but even if the temporal information is discarded, high PSC score can be obtained with only two detectors, which may due to the long-tailed distribution of the dataset and high correlation between part state and video scene.

In model capacity improvement, methods designed with different structures are good at different categories of videos in the prediction of part state, so ensemble multiple results of heterogeneous methods can greatly improve our score.

In our future work, we will construct a more refined method that replaces the input of the RGB image of the person with vector input such as pose keypoints, so as to learn more accurate and interpretable actions of each part, and combine the person-context-person module to build an online-available and end-to-end network without any additional object detectors.

References

1. Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022.
2. T. Y. Lin, M. Maire, S. Belongie, J. Hays, and C. L. Zitnick. Microsoft coco: Common objects in context, 2014.
3. Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer, 2021.
4. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2014.
5. Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzyńska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense, 2017.
6. D. Tran, H. Wang, L. Torresani, and M. Feiszli. Video classification with channel-separated convolutional networks, 2019.
7. Deepti Ghadiyaram, Matt Feiszli, Du Tran, Xueting Yan, Heng Wang, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition, 2019.
8. J. Pan, S. Chen, M. Z. Shou, Y. Liu, J. Shao, and H. Li. Actor-context-actor relation network for spatio-temporal action localization, 2021.
9. C. Gu, S. Chen, D. A. Ross, C. Vondrick, and J. Malik. Ava: A video dataset of spatio-temporally localized atomic visual actions, 2018.
10. Jiajun Tang, Jin Xia, Xinzhi Mu, Bo Pang, and Cewu Lu. Asynchronous interaction aggregation for action detection, 2020.