EURASIP Journal on Image
and Video Processing

RESEARCH Open Access

Consistent constraint-based video-level learning for action recognition

Qinghongya Shi[1,2,3], Hong-Bo Zhang[1,2,3*], Hao-Tian Ren[1,2,3], Ji-Xiang Du[1,2,3] and Qing Lei[1,2,3]

*Correspondence:
zhanghongbo@hqu.edu.cn
[1]Department of Computer Science and Technology, Huaqiao University, Xiamen, Fujian, China
[2]Fujian Key Laboratory of Big Data Intelligence and Security, Huaqiao University, Xiamen, Fujian, China
Full list of author information is available at the end of the article

Abstract

This paper proposes a new neural network learning method to improve the performance for action recognition in video. Most human action recognition methods use a clip-level training strategy, which divides the video into multiple clips and trains the feature learning network by minimizing the loss function of clip classification. The video category is predicted by the voting of clips from the same video. In order to obtain more effective action feature, a new video-level feature learning method is proposed to train 3D CNN to boost the action recognition performance. Different with clip-level training which uses clips as input, video-level learning network uses the entire video as the input. Consistent constraint loss is defined to minimize the distance between clips of the same video in voting space. Further, a video-level loss function is defined to compute the video classification error. The experimental results show that the proposed video-level training is a more effective action feature learning approach compared with the clip-level training. And this paper has achieved the state-of-the-art performance on UCF101 and HMDB51 datasets without using pre-trained models of other large-scale datasets. Our code and final model are available at https://github.com/hqu-cst-mmc/VLL.

Keywords: Consistent constraint, Video-level learning, 3D CNN, Action recognition, Loss function

1 Introduction

Action recognition has gradually become a research hotspot in computer vision and pattern recognition, which is widely applied in intelligent video surveillance, virtual reality, motion analysis, and video retrieval. How to improve the accuracy of human action recognition has been studied by many researchers.

Many methods have been proposed to recognize action in video in recent years. The key to these methods is to learn effective action feature from input data. Several different neural networks are employed in these methods, such as 3D convolutional neural network (ConvNets) [1, 2], multi-stream 2D ConvNets [3–5], and recurrent neural network [6, 7]. The difference between video feature and image feature is whether it contains temporal information. To deal with the different temporal length of video and reduce computational complexity, the input video is divided into a clip set. Each clip has the same number of frames, and the video label is assigned to each clip. In the training stage, the parameters of network are learned from annotated action clips. In the testing stage, each clip in video is classified by the network and the video category is predicted by the voting strategy. This training approach in these methods is named as clip-level training in this work.

Although these works have obtained some significant results, there are some limitations in clip-level learning. First, to feed into the convolutional network, each clip is segmented from video with fixed length by dense sampling or sparse sampling. However, short clip cannot obtain the complete temporal information of human action in video. Therefore, the vision features extracted from the model, which is trained by the clip set, cannot accurately represent the action. Second, during the training stage, the calculation process of each clip is independent in these clip-level methods. It ignores the correlation between clips in the same video. To solve these problems, this paper proposes a new action feature learning method, called video-level learning.

The object of video-level learning is to train the network which can provide more complete and effective video representation rather than clip representation. In video-level learning, the input of network is the initial video. In the pre-processing stage, the video is also divided into a clip set. The video label is assigned as the label of the clip set, rather than each clip. The difference between clip-level and video-level is shown in Fig. 1. The video-level learning can be regraded as the problem of set learning. The clip set covers all the content of the video; therefore, the features learned from the clip set can contain richer action information than those features learned from a single clip.

To build the video-level learning model, the core is to train the parameters of network through each clip set. In this work, we use 3D ConvNets as the basic network model. According to the theory of convolutional network [8], in the training stage, we have to tell 3D ConvNets what we wish it to minimize. Therefore, to implement video-level learning, the video-level loss function is defined in our method. Using the clip set as input, which sampled from the same video, the video-level loss function of network not only needs to consider the error rate of clip classification, but also needs to consider the relationship between the clips in the same video. To solve this problem, in the proposed video-level learning method, a consistent constraint loss (CCL) function is defined for training the 3D ConvNets. The basic assumption of CCL is that the distance between the clips of the same video in the voting space should be small. Therefore, in the proposed method, the video-level loss (VLL) function includes two items: average classification error of clip set and distance loss of clip set.

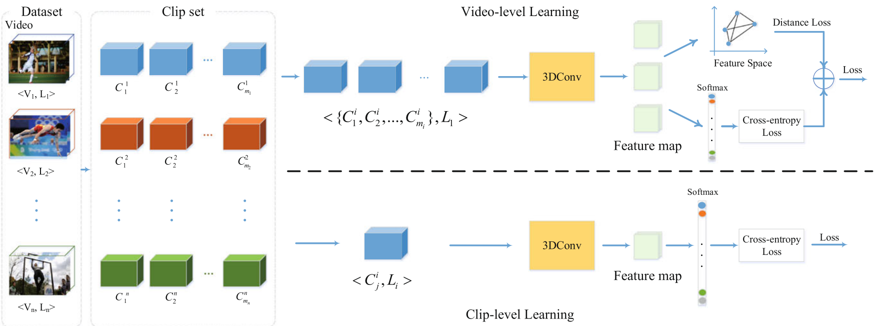


Fig. 1 Comparison of the video-level learning framework and the clip-level learning framework. V_i is the *i*th video, and L_i means action label. C_j^i is the *j*th clip in video *i*, and m_i is the number of clip in video *i*

In summary, although most of clip-level-based approaches also take several clips as a batch and feed into the network in the actual training process, there are still two main differences between clip-level learning and video-level learning. First, the input data of video-level learning is the clips which are sampled from the same video instead of selecting clips randomly in the clip-level learning. At the same time, the number of clips, which corresponds to the batch size in the training stage, will change dynamically in video-level learning. In other words, the batch size is dynamic in the proposed method. It depends on the video length and sampling method. This means that during the training phase, the network can better adapt video data with different temporal scale and learn more complete information, while in clip-level learning, the number of clips is fixed, which is determined by the pre-defined batch size.

Second, clip-level learning methods use clip classification error as the loss function, such as the average of cross entropy loss of input clips. In video-level learning, a new classification loss of video needs to be defined, such as VLL defined in this paper. The framework of video-level learning is shown in Fig. 1.

Finally, in the testing stage, this paper uses the same voting strategy in clip-level learning to achieve video action classification. The contributions of this paper are threefold:

(1) We propose a new end-to-end training method for video-level feature learning to improve the accuracy of action recognition.

(2) For video-level training, consistent constraint loss is defined to minimize the distance between clips of the same video in voting space. A new loss function of video classification is designed to unify all clips that belonged to the same video.

(3) The experimental results demonstrate that the proposed method is effective. The network trained by the video-level method has better performance than the clip-level method. And without using the pre-trained model of other large-scale data, the proposed method provides higher recognition rates than those of state-of-the-art action recognition methods.

The remainder of this paper is organized as follows. Section 2 introduces the related works, Section 3 describes the algorithms used to implement the proposed method, Section 4 presents and discusses the experimental results, and finally, Section 5 concludes this paper.

2 Related work

Many human action recognition methods have been proposed in recent years. Zhang et al. [9] summarized the work in recent years from different data perspectives including RGB video-based methods [1, 2, 10, 11], depth data-based methods [12, 13], and skeleton data-based methods [14–16]. Although research based on depth and skeleton data has attracted some attention, human action recognition methods in RGB video have always been the mainstream research direction. The paper also focuses on the human action recognition methods in video.

In recent studies, deep learning methods have shown good performance in feature exaction and action recognition. They have become the mainstream method of computer vision research. In [17, 18], for real-time face landmark extraction, the authors proposed a new model called EMTCNN that extended from multi-task cascaded convolutional neural network. To learn action feature, there are two main network structures in these methods: 3D ConvNets and multi-stream structure.

2.1 3D ConvNets for action recognition

3D ConvNets is extended from 2D ConvNets, and its convolution kernel contains three dimensions: the two dimensions represent the space information and the other dimension represents the temporal information. 3D convolution kernel can calculate both temporal and spatial features, but it also has more parameters, making the computation of 3D convolutional network larger. Tran et al. [2] first proposed 3D convolutional network for learning spatio-temporal features. Carreira et al. [1] proposed inflated 3D ConvNets (I3D) and used pre-trained models from large dataset to obtain the highest accuracy of human action recognition, such as Kinetics [19] and ImageNet dataset. In [10], a pseudo-3D residual networks (P3D) is proposed to learn spatial-temporal representation. Hara et al. [20] extended the 2D ResNet to 3D structure and proposed the ResNeXt method to recognize action. And Tran et al. [11] tried to decrease the number of parameters, making the 3D convolution kernel decomposing to 2D spatial convolution kernel and 1D temporal convolution kernel.

2.2 Multi-stream structure for action recognition.

To model the temporal information, several methods added the correspondence motion sequence of the input video to the feature learning network, such as optical flow image sequence and motion boundary sequence. Simonyan et al. [21] used a two-stream structure to calculate the appearance and motion from image and optical flow sequence respectively. The appearance feature and motion feature were fused as the action feature. Wang et al. [22] proposed temporal segment networks (TSN) which used sparse temporal sampling to obtain long-range temporal information. In addition, some works applied the self-supervised approaches to learn video feature based on multi-stream structure. Wang et al. [23] proposed a two-stream-based self-supervised approach to learn visual feature by regressing both motion and appearance statistical information without action label. In this work, both RGB data and optical data were used to compute appearance and motion respectively. Crasto et al. [24] introduced a learning network instead of the effect achieved by optical flow, but it also needs optical flow to train. Wang et al. [25] proposed two-stream and 3D ConvNets fusion mode to recognize human action with arbitrary size and length.

In addition, there are also some works using an attention module to improve the accuracy of action detection and recognition [26–28]. Li et al. [26] proposed an attention-based GCN for action detection in video to capture the relations among proposals and reduce the redundant proposals. And in [28], the authors proposed a new spatio-temporal deformable ConvNet model with an attention mechanism, which takes into consideration the mutual correlations in both temporal and spatial domains, to effectively capture the long-range and long-distance dependencies in the video actions.

However, all the above methods use clip-level training strategy. And due to that long-range clip needs higher computational costs, the length of the clip is short, generally 16 frames in these works. In this paper, 3D ConvNets is also used as the basic network. Different with these methods, this paper uses the video-level method instead of the clip-level method to train more accuracy feature representation network. In addition, some works used the large dataset, such as sports1M [29] and Kinetics [19], to achieve high performance. However, large dataset implies greater computational costs. How to obtain more higher recognition performance without pre-trained model is still an issue that is worth to study. It is also discussed in this paper.

3 The proposed method

3.1 Problem definition

To describe the proposed method of action recognition in this paper, the problem of video-level learning can be defined as follows. The proposed method uses a set of pairwise components $D = \{< V_1, L_1 >, < V_2, L_2 >, ..., < V_n, L_n >\}$ to present the input data, where V_i denotes the ith video in the dataset, L_i is the correspondence action label, and n is the number of the videos. In this paper, the input video is segmented into a clip set $V_i = \left\{C_1^i, C_2^i, ..., C_{m_i}^i\right\}$, where C_j^i is the jth clip in ith video and m_i is the size of the clip set sampled from video. The input of network $< V_i, L_i >$ is transferred to a new pairwise of clip set and label $< \left\{C_1^i, C_2^i, ..., C_{m_i}^i\right\}, L_i >$, as illustrated in Fig. 1. This paper samples a fixed-length continuous frames from the video as a clip. The process of clip generation is shown in Fig. 2. Referring to the clip-level learning methods [1, 2, 23], the length of the clip is set to 16 frames in this paper.

Suppose that the proposed model can be defined as a function $L_i = f(V_i)$. The task of the training stage is to learn the parameters in this function. After parameter training, given a testing video V_t, the category of the video L_t is calculated by this function. The detail of the proposed model is described in the following section.

$$L_t = f\left(V_t\right) = f\left(\left\{C_1^t, C_2^t, ..., C_{m_t}^t\right\}\right) \tag{1}$$

3.2 Network structure and video-level learning

In this work, the proposed network uses 3D ConvNets as the feature extractor. The network structure of the proposed network is shown in Fig. 3. In order to make the parameters of network as few as possible, we only use 5 convolution layers and 5 pooling layers (each convolution layer is immediately followed by a pooling layer), 2 fully connected layers and a softmax layer to predict action labels. Inspired by the previous works of 3D ConvNets [1, 2, 23], all of the convolutional kernels are set to $3 * 3 * 3$ in the proposed approach.

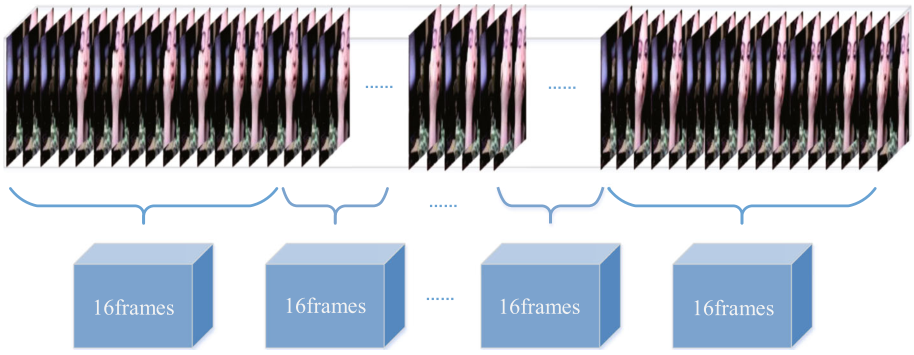


Fig. 2 The process of clip generation

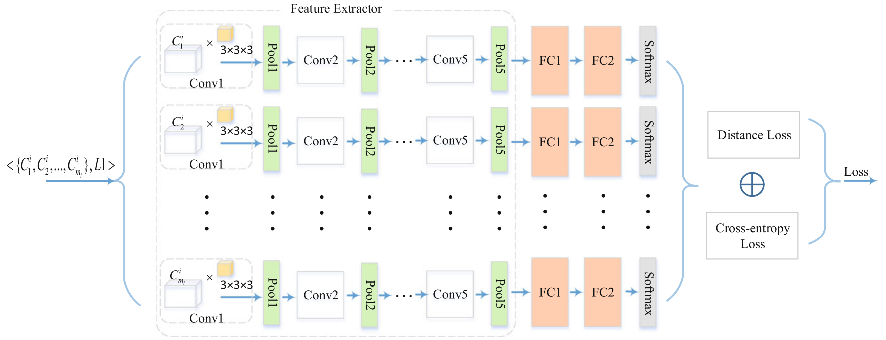


Fig. 3 The structure of the proposed network

To improve the action recognition performance of 3D ConvNets, the video-level learning strategy is proposed in this paper. In our method, the network used the whole video as the input. The network needs to process clip set with different size. Therefore, the network extracts the feature for each clip independently. To achieve video-level training, the video-level loss is defined by minimizing the classification error of each clip and the distance of each clip pair, which is named as consistent constraint loss in this paper.

3.3 Video-level loss function

The most common loss function of 3D ConvNets is cross entropy function Loss_{ce}, aiming to measure the similarity between the distribution of ground truth and the distribution of predicted label, as shown in Eq. 2. In clip-level learning, the 3D ConvNets is trained by minimizing the classification error of clips.

$$\mathrm{Loss}_{ce}\left(y,\hat{y}\right) = -\sum_{j=1}^{N} y_j \log\left(\hat{y}_j\right) \tag{2}$$

where y is the one-hot vector of ground truth of the input video, N is the number of category, and $\hat{y}$ is the predict score of the clip.

However, this strategy ignores the relationship between clips. To address this problem, video-level loss is proposed in this work. To calculate video-level loss, the video classification loss is computed by the average of cross entropy function of all clips from same video. And more importantly, the distance of these clips in voting space is defined as the consistent constraint loss Loss_{ccl}. Finally, the video-level loss Loss_{vll} is a combination of the average Loss_{ce} and Loss_{ccl}, as shown in Fig. 4. The calculation process of these functions is defined as follows.

$$\begin{aligned}\mathrm{Loss}_{vll} =&(1-\alpha)\frac{1}{m_i}\sum_{j=1}^{m_i}\mathrm{Loss}_{ce}\left(y_j,\hat{y}_j\right)\\ &+\alpha\mathrm{Loss}_{ccl}\end{aligned} \tag{3}$$

where m_i is the size of the input clip set corresponding to the ith video and α is the balanced weight of cross entropy loss function and consistent constraint loss function.

To achieve the assumption of the consistent constraint, which means the output of the network for each clip from the same video should be consistent, this paper uses the output vector of the network as the input of consistent constraint loss function. And the consistent constraint is computed by the distance of each clip pair in the same video. The

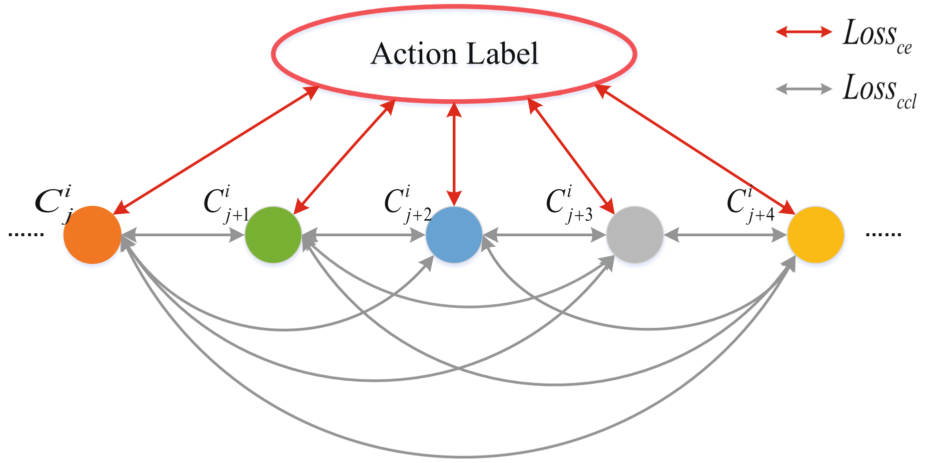


Fig. 4 Video-level loss function

purpose of adding this constraint to the video-level loss function is to make the network provide more closer the classification score for clips from the same video. In this work, there are several consistent constraint loss functions that are discussed.

First, the average of Euclidean distance of each clip pair is applied to compute the consistent constraint loss. It is defined as Euclidean distance loss as shown in Eq. 4.

$$\text{Loss}^{euc}_{ccl} = \frac{2}{m_i * (m_i - 1)} \sum_{k=1}^{m_i} \sum_{j>k}^{m_i} \|\hat{y}_k - \hat{y}_j\|_2 \tag{4}$$

where $\frac{m_i*(m_i-1)}{2}$ is the number of clip pair in the input set. Because Euclidean distance of clip pair is symmetric, that is $\|\hat{y}_k - \hat{y}_j\|_2 = \|\hat{y}_j - \hat{y}_k\|_2$, we only calculate one of them in Eq. 4. The index of k is from 1 to m_i, and the index of j is from $k+1$ to m_i, total $\frac{m_i*(m_i-1)}{2}$ items.

Further, during the training process, the distribution of prediction scores of the samples with incorrect prediction should be more and more consistent with the samples with correct prediction. Therefore, the consistent constraint loss function is further defined in Eq. 5, which is named as error direction loss.

$$\text{Loss}^{err}_{ccl} = \frac{1}{N_e} \sum_{i \in E} \|\hat{y}_i - R_{\text{mean}}\|_2 \tag{5}$$

where E is the set of samples that are predicted incorrectly in each round of training. N_e is the size of set E, and $N_e < m_i$. R_{mean} is the mean vector of the prediction score of the samples that are predicted correctly.

Finally, the consistent constraint loss proposed in this paper is composed of Eqs. 4 and 5, as shown as follows. During the training, we not only make the output of the network more consistent for the clips in the same video, but more importantly, we require the network to adjust the output of the clips, which are incorrectly predicted, to be more consistent with the output of the clips that are predicted correctly.

$$\text{Loss}_{ccl} = \begin{cases} \text{Loss}^{euc}_{ccl} & N_e = m_i || N_e = 0 \\ \text{Loss}^{err}_{ccl} & \text{others} \end{cases} \tag{6}$$

4 Experimental results and discussion

In this section, some experiments are performed on UCF101 [30] and HMDB51 [31] datasets to verify the effectiveness of the proposed video-level learning method.

4.1 Dataset and experiment setting

UCF101 dataset. UCF101 contains 101 action categories, total 13320 videos which are collected from YouTube. It is one of the most commonly used datasets in the research of action recognition. Each video has been divided into several clips, total 148166 clips in this experiment. And UCF101 provides the diversity in terms of actions, and with the presence of large variations in camera motion, object appearance, and pose, it is also a challenging dataset.

HMDB51 dataset. HMDB51 collects the videos from various sources, mostly from movies, and a small proportion from public databases such as the Prelinger archive, YouTube, and Google videos. The dataset contains 6766 videos with 51 action categories, each containing a minimum of 101 clips.

In the experiment, this paper uses standard training/testing splits from the official website of these datasets. To reduce the parameters of 3D ConvNets, this paper uses relu function as activate function and set dropout value to 0.05. All input images are resized to 112*112 with random crop. The entire network is fine-tuned with SGD on 0.001 learning rate. And every 2000 iterations, the learning rate decreased by 0.1. The balanced weight of video-level loss function is set to 0.3. The proposed method is implemented on two NVIDIA GeForce RTX 2080Ti GPUs which take about 8 and 2.5 h to train the model on UCF101 and HMDB51 datasets respectively. For fair comparison, in the testing, all methods use the accuracy of video category, which is predicted by the voting of clip category in this video, as the measure metric.

4.2 Performance evaluation on UCF101

4.2.1 Comparison of video-level training and clip-level training

To compare the performance of video-level and clip-level training, some experimental results are shown in Table 1. In Table 1, using clip-level training with cross entropy loss function, the accuracy of action classification is 51.52%. The action classification accuracy of the network, which is trained by video-level strategy with the same loss function, is 53.44%. From this comparison, it can be seen that video-level training is more effective than clip-level training, and the accuracy is improved by 1.92%.

4.2.2 Comparison of different loss functions in video-level training

In video-level training, the proposed loss functions also are discussed in the experiment. In Table 1, $\text{Loss}_{vll}\left(\text{Loss}_{ccl}^{euc}\right)$ indicates that Euclidean distance loss function Loss_{ccl}^{euc}

Table 1 Accuracy of clip-level training and video-level training with different loss functions

Training method	Loss function	UCF101 (%)
Clip level	Loss_{ce}	51.52
Video level	Loss_{ce}	53.44
	$\text{Loss}_{vll}\left(\text{Loss}_{ccl}^{err}\right)$	55.38
	$\text{Loss}_{vll}\left(\text{Loss}_{ccl}^{euc}\right)$	57.11
	$\text{Loss}_{vll}\left(\text{Loss}_{ccl}\right)$	**58.76**

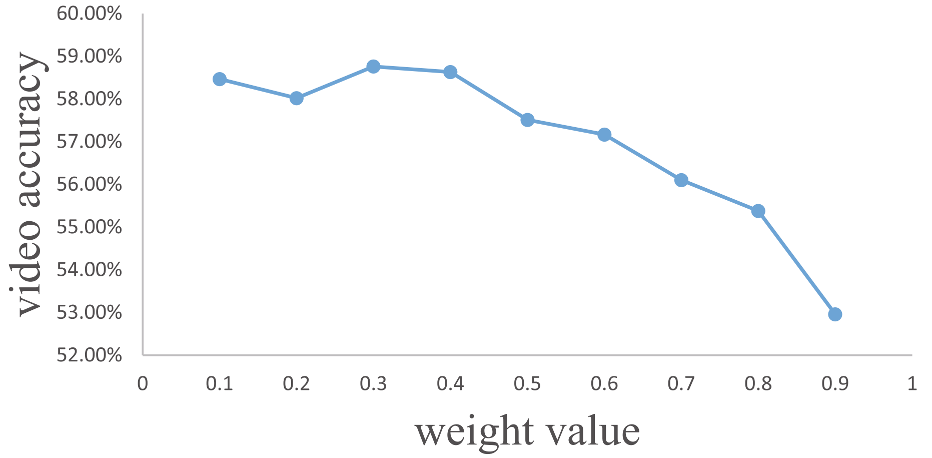


Fig. 5 Recognition accuracy with different weight α

is used as the consistent constraint loss function in video-level loss function Loss_{vll}. Loss_{vll} $\left(\text{Loss}_{ccl}^{err}\right)$ indicates that error direction loss function Loss_{ccl}^{err} is applied as the consistent constraint loss function in Loss_{vll}. And Loss_{vll} (Loss_{ccl}) means using Loss_{ccl} function which is defined in Eq. 6 as the consistent constraint loss function in Loss_{vll}. From this comparison, the action recognition accuracy of the network which is trained by the video-level loss function with Euclidean distance loss Loss_{ccl}^{euc} is 57.11%. It is 3.67% higher than the results of the network trained by cross entropy loss function.

Compared to the loss function Loss_{vll} $\left(Loss_{ccl}^{euc}\right)$, the network trained by the loss function Loss_{vll} $\left(\text{Loss}_{ccl}^{err}\right)$ has better recognition performance. Finally, the proposed method, which uses Loss_{vll} (Loss_{ccl}) function to train the 3D ConvNets, obtains the highest accuracy 58.76%, which is 7.24% higher than the network trained by clip-level training.

4.2.3 Balanced weight discussion

The performance of different weight value α in video-level loss function Loss_{vll} is shown in Fig. 5. In this experiment, the consistent constraint loss function in Loss_{vll} use Loss_{ccl} function defined in Eq. 6, and α is set from 0.1 to 0.9. In Fig. 5, $\alpha = 0.3$ achieves the best recognition accuracy.

4.2.4 Comparison with the state-of-the-art

To evaluate the effectiveness of the proposed method, the accuracy of the proposed method is compared with the state-of-the-art methods. The comparison results are

Table 2 Comparison results of the proposed with other state-of-the-art action recognition methods

Training method	Method	Accuracy (%)
Clip level	C3D [2]	51.52
	Geometry [32]	55.2
	CD-UAR [33]	42.5
	3D-ShuffleNetV2 [34]	56.52
	MASN [23]	53.44
Video level	**Ours**	**58.76**

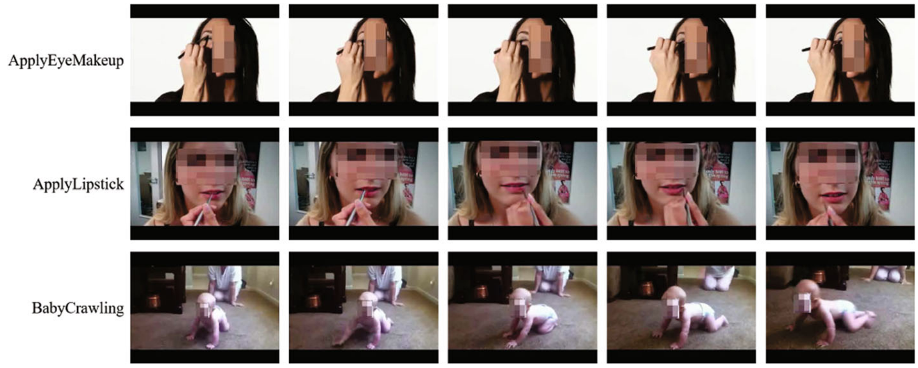


Fig. 6 Examples of the videos selected for qualitative evaluation

shown in Table 2. None of the methods in Table 2 uses the pre-trained of other large datasets. Table 2 indicates that the proposed method has the better recognition accuracy.

4.2.5 Qualitative evaluation

To further verify the effectiveness of the proposed method, we show some action recognition results of video-level training and clip-level training. We use red to represent the corresponding ground truth, and other colors for the rest of the categories. We select three videos for qualitative evaluation. Some examples of these videos are shown in Fig. 6. These samples are selected from three different action categories: "ApplyEyeMakeup," "ApplyLipstick," and "BabyCrawling." Each video contains 16 clips. Each clip has 16 frames. Figures 7 and 8 show the recognition results of these three samples.

In Fig. 7a, it is the vote distribution of the clips in testing video. Each clip category is predicted by the clip-level training model. From Fig. 7a, we can find that based on the clip-level training model, only one clip is correctly classified, and the other 15 clips are classified into the wrong category in sample 1. In sample 2, two clips are correctly classified, and the other 14 clips are classified into the wrong categories. In sample 3, there are 7 clips correctly classified. Therefore, based on clip-level training, the recognition results of sample 1 and sample 2 are wrong, and sample 3 is correctly predicted. Similarly, Fig. 7b is the clip vote distribution predicted by the video-level training model. Based on the video-level training model, in these samples, the number of correctly classified clips

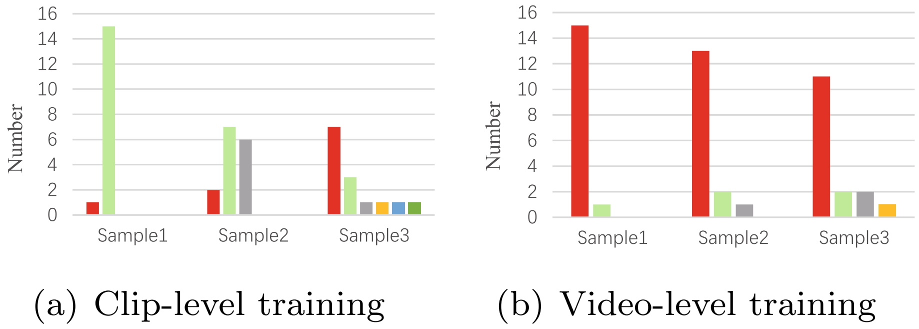
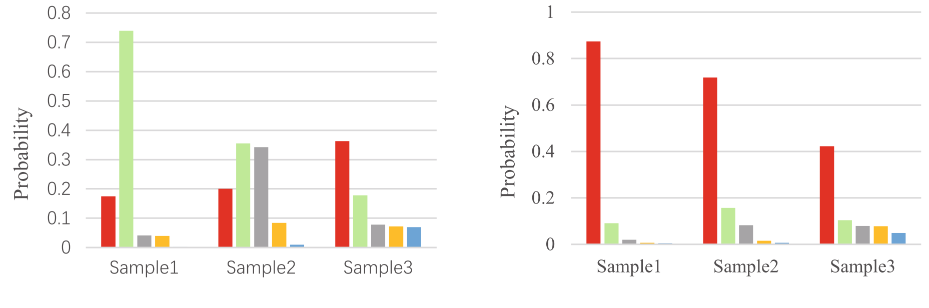


Fig. 7 Visualization of action recognition result by voting



(a) Clip-level training (b) Video-level training

Fig. 8 Visualization of action recognition result on top 5 probabilities

has increased significantly. For example, in sample 1, there are 15 clips correctly predicted and only one clip is classified incorrectly. Finally, both sample 1 and sample 2 can be correctly classified by using the video-level training model.

In addition, we can obtain the same conclusion from the probability of video classification. Figure 8a shows the top 5 probabilities of video classification by the clip-level training, while Fig. 8b shows the top 5 probabilities of video-level training. We choose the category with the highest predict probability as the predict result. From the comparison, it can be found that based on the video-level training model, sample 1 and sample 2 can be adjusted from the classification error to a high probability on the category corresponding to ground truth, and then correctly classified. For sample 3, the probability corresponding to ground truth is increased by using the video-level training model.

4.3 Performance evaluation on HMDB51

4.3.1 Comparison of video-level training and clip-level training

In this section, we evaluate the performance of the proposed method on the HMDB51 dataset. HMDB51 is less than UCF101, which has only 51 classes. Table 3 shows the experimental results. From Table 3, we can find that compared with clip-level training, using the same cross entropy loss function, the model trained by video-level strategy can improve the recognition accuracy from 32.6 to 33.15%.

As the above introduction, a new loss function Loss$_{vll}$ is used in video-level learning. Comparing with only cross entropy loss function, the accuracy of the model which is trained by the proposed consistent constraint loss function in Loss$_{vll}$ is 35.38%, which is improved by 2.23%.

4.3.2 Comparison with the state-of-the-art

We also compare the proposed method with other state-of-the-art methods on HMDB51 dataset. Table 4 shows the comparison results. Compared with the methods without using

Table 3 Accuracy of clip-level training and video-level training with different loss functions on HMDB51

Training method	Loss function	HMDB51
Clip level	Loss$_{ce}$	32.6%
Video level	Loss$_{ce}$	33.15%
	Loss$_{vll}$ (Loss$_{ccl}$)	**35.38%**

Table 4 Comparison results of the proposed with other state-of-art action recognition methods on HMDB51

Training method	Method	Accuracy (%)
Clip level	Geometry [32]	23.3
	MASN [23]	32.6
	ST-puzzle (Kinetics) [35]	28.3
	MASN (Kinetics) [23]	33.4
Video level	**Ours**	**35.38**

other large-scale dataset to pre-train the model, our method achieves the higher accuracy. Compared with the methods using Kinetics dataset to pre-train the model, the proposed method also has better performance. All the above experiments further verify that the proposed video-level training method is also effective on small dataset.

5 Conclusion

In this paper, we proposed a new neural network training method named video-level learning to improve the performance of 3D ConvNets. Different with the traditional training method which used clips as input, the proposed method used the entire video as input. This method defined a video-level loss function which contained cross entropy loss function and consistent constraint loss function to train the 3D ConvNets. And in this paper, we discussed three different consistent constraint loss functions. The experimental results show that in comparison with the clip-level learning method, the proposed method has better action recognition performance. And the effectiveness of the proposed method is verified by comparison with the state-of-art methods.

Although the proposed method can effectively improve the accuracy of the network, this work still has some limitations. In this paper, we only report the results without using pre-trained models of other large-scale datasets. There are mainly the following reasons. First, to verify the effectiveness of the proposed method, our motivation is to use the simplest 3D ConvNets as the basic network to highlight the impact of video-level learning. The backbone network of 3D ConvNets only contains 5 convolution layers, 5 pooling layers, 2 fully connected layers, and a softmax layer. Second, we also pay attention to some complex convolution networks which have been proposed with better performance in recent years, such as P3D [10] and 3D ResNet [20]. To use these pre-trained models, it needs to modify the structure of the backbone network to be consistent with the structure of these well-trained models. However, what kind of network structure is the best for action recognition still is an open and complex issue in action recognition research.

In the future work, we will try to find more effective 3D convolutional model instead of the simple 3D ConvNets which is used in this work, discuss the performance of these methods based on the well-trained models, and apply these methods on other large-scale action datasets.

Abbreviations

3D ConvNets: Three-dimension convolution network; CNN: Convolution neural network; CCL: Consistent constraint loss; VLL: Video-level learning

Acknowledgements
The authors would like to thank the anonymous reviewers for their valuable and insightful comments on an earlier version of this manuscript.

Authors' contributions

All authors took part in the discussion of the work described in this paper. All authors read and approved the final manuscript.

Authors' information

Qinghongya Shi received the B.S. degree from Huaqiao University, China, in 2017, where she is currently pursuing the M.S. degree. Her research interests include image processing and computer vision.
Hong-Bo Zhang received a Ph.D. in Computer Science from Xiamen University in 2013. Currently, he is an associate professor with the School of Computer Science and Technology of Huaqiao University. He is the member of Fujian key laboratory of big data intelligence and security. His research interests include computer vision and pattern recognition.
Hao-Tian Ren received the B.S. degree from Huaqiao University, China, in 2018, where she is currently pursuing the M.S. degree. Her research interests include computer vision and machine learning.
Ji-Xiang Du received a Ph.D. in Pattern Recognition and Intelligent System from the University of Science and Technology of China (USTC), Hefei, China, in 2005. He is currently a professor at the School of Computer Science and Technology at Huaqiao University. He is the director of Fujian key laboratory of big data intelligence and security. His current research interests mainly include pattern recognition and machine learning.
Qing Lei received a Ph.D. from the Cognitive Science Department of Xiamen University, China. She joined the faculty of Huaqiao University in 2005. Her research interests include human motion analysis and object detection/recognition.

Funding

This work is supported by the Natural Science Foundation of China (No. 61871196 and 61673186), the National Key Research and Development Program of China (No. 2019YFC1604700), the Natural Science Foundation of Fujian Province of China (No. 2019J01082) and Promotion Program for Young and Middle-aged Teacher in Science and Technology Research of Huaqiao University (ZQN-YX601), Subsidized Project for Postgraduates'Innovative Fund in Scientific Research of Huaqiao University(No.18014083014).

Availability of data and materials

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Department of Computer Science and Technology, Huaqiao University, Xiamen, Fujian, China. [2]Fujian Key Laboratory of Big Data Intelligence and Security, Huaqiao University, Xiamen, Fujian, China. [3]Xiamen Key Laboratory of Computer Vision and Pattern Recognition, Huaqiao University, Xiamen, Fujian, China.

Received: 6 April 2020 Accepted: 14 July 2020
Published online: 31 August 2020

References

1. J. Carreira, A. Zisserman, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Quo vadis, action recognition? A new model and the kinetics dataset (IEEE, Honolulu, 2017), pp. 4724–4733. https://doi.org/10.1109/CVPR.2017.502
2. D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, in *Proceedings of the IEEE International Conference on Computer Vision*, Learning spatiotemporal features with 3d convolutional networks (IEEE, Santiago, 2015), pp. 4489–4497
3. W. Dai, Y. Chen, C. Huang, M. Gao, X. Zhang, in *2019 International Joint Conference on Neural Networks (IJCNN)*, Two-stream convolution neural network with video-stream for action recognition (IEEE, Budapest, 2019), pp. 1–8. https://doi.org/10.1109/IJCNN.2019.8851702
4. J. Xu, K. Tasaka, H. Yanagihara, in *2018 24th International Conference on Pattern Recognition (ICPR)*, Beyond two-stream: skeleton-based three-stream networks for action recognition in videos (IEEE, Beijing, 2018), pp. 1567–1573. https://doi.org/10.1109/ICPR.2018.8546165
5. V. A. Chenarlogh, F. Razzazi, Multi-stream 3D CNN structure for human action recognition trained by limited data. IET Comp. Vision. **13**(3), 338–344 (2019). https://doi.org/10.1049/iet-cvi.2018.5088
6. L. Song, L. Weng, L. Wang, X. Min, C. Pan, in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Two-stream designed 2d/3d residual networks with lstms for action recognition in videos (IEEE, Athens, 2018), pp. 808–812. https://doi.org/10.1109/ICIP.2018.8451662
7. T. Lin, X. Zhao, Z. Fan, in *2017 IEEE International Conference on Image Processing (ICIP)*, Temporal action localization with two-stream segment-based RNN (IEEE, Beijing, 2017), pp. 3400–3404. https://doi.org/10.1109/ICIP.2017.8296913
8. Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013). https://doi.org/10.1109/TPAMI.2013.50
9. H.-B. Zhang, Y.-X. Zhang, B. Zhong, Q. Lei, L. Yang, J.-X. Du, D.-S. Chen, A comprehensive survey of vision-based human action recognition methods. Sensors. **19**(5), 1005 (2019). https://doi.org/10.3390/s19051005
10. Z. Qiu, T. Yao, T. Mei, in *Proceedings of the IEEE International Conference on Computer Vision*, Learning spatio-temporal representation with pseudo-3d residual networks (IEEE, Venice, 2017), pp. 5533–5541
11. D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, M. Paluri, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, A closer look at spatiotemporal convolutions for action recognition (IEEE, Salt Lake City, 2018), pp. 6450–6459
12. C. Zhang, Y. Tian, X. Guo, J. Liu, DAAL: deep activation-based attribute learning for action recognition in depth videos. Comp. Vision Image Underst. **167**, 37–49 (2018). https://doi.org/10.1016/j.cviu.2017.11.008

13. Z. Shi, T.-K. Kim, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Learning and refining of privileged information-based RNNS for action recognition from depth sequences (IEEE, Honolulu, 2017), pp. 3461–3470
14. C. Si, W. Chen, W. Wang, L. Wang, T. Tan, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, An attention enhanced graph convolutional lstm network for skeleton-based action recognition (IEEE, Long Beach, 2019), pp. 1227–1236. https://doi.org/10.1109/CVPR.2019.00132
15. S. Yan, Y. Xiong, D. Lin, in *Thirty-second AAAI Conference on Artificial Intelligence*, Spatial temporal graph convolutional networks for skeleton-based action recognition (AAAI, New Orleans, 2018)
16. M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, Q. Tian, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Actional-structural graph convolutional networks for skeleton-based action recognition (IEEE, Long Beach, 2019), pp. 3590–3598. https://doi.org/10.1109/CVPR.2019.00371
17. H. Kim, H. Kim, E. Hwang, in *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Real-time facial feature extraction scheme using cascaded networks (IEEE, Kyoto, 2019), pp. 1–7
18. H.-W. Kim, H.-J. Kim, S. Rho, E. Hwang, Augmented EMTCNN: a fast and accurate facial landmark detection network. Appl. Sci. **10**(7), 2253 (2020). https://doi.org/10.3390/app10072253
19. W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al., The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
20. K. Hara, H. Kataoka, Y. Satoh, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet? (IEEE, Salt Lake City, 2018), pp. 6546–6555
21. K. Simonyan, A. Zisserman, in *Advances in Neural Information Processing Systems*, Two-stream convolutional networks for action recognition in videos (Neural information processing systems foundation, Montreal, 2014), pp. 568–576
22. L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, L. Van Gool, in *European Conference on Computer Vision*, Temporal segment networks: towards good practices for deep action recognition (Springer, Amsterdam, 2016), pp. 20–36
23. J. Wang, J. Jiao, L. Bao, S. He, Y. Liu, W. Liu, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics (IEEE, Long Beach, 2019), pp. 4006–4015
24. N. Crasto, P. Weinzaepfel, K. Alahari, C. Schmid, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Mars: Motion-augmented RGB stream for action recognition (IEEE, Long Beach, 2019), pp. 7882–7891
25. X. Wang, L. Gao, P. Wang, X. Sun, X. Liu, Two-stream 3-D convNet fusion for action recognition in videos with arbitrary size and length. IEEE Trans. Multimed. **20**(3), 634–644 (2018)
26. J. Li, X. Liu, Z. Zong, W. Zhao, M. Zhang, J. Song, in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, February 7-12, 2020*, Graph attention based proposal 3D convNets for action detection (AAAI Press, New York, NY, USA, 2020), pp. 4626–4633. https://aaai.org/ojs/index.php/AAAI/article/view/5893
27. J. Li, X. Liu, W. Zhang, M. Zhang, J. Song, N. Sebe, Spatio-Temporal Attention Networks for Action Recognition and Detection. IEEE Trans. Multimed., 1–1 (2020). https://doi.org/10.1109/TMM.2020.2965434
28. J. Li, X. Liu, M. Zhang, D. Wang, Spatio-temporal deformable 3d convnets with attention for action recognition. Pattern Recog. **98**, 107037 (2020). https://doi.org/10.1016/j.patcog.2019.107037
29. A. Karpathy, G. Toderici, S. Shetty, T. Leung, F. F. Li, in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Large-scale video classification with convolutional neural networks (IEEE, Columbus, 2014)
30. K. Soomro, A. R. Zamir, M. Shah, Ucf101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
31. H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, T. Serre, in *2011 International Conference on Computer Vision*, HMDB: a large video database for human motion recognition (IEEE, Barcelona, 2011), pp. 2556–2563
32. C. Gan, B. Gong, K. Liu, H. Su, L. J. Guibas, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Geometry guided convolutional neural networks for self-supervised video representation learning (IEEE, Salt Lake City, 2018), pp. 5589–5597
33. Y. Zhu, Y. Long, Y. Guan, S. Newsam, L. Shao, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Towards universal representation for unseen action recognition (IEEE, Salt Lake City, 2018), pp. 9436–9445
34. O. Köpüklü, N. Kose, A. Gunduz, G. Rigoll, Resource efficient 3D convolutional neural networks. arXiv preprint arXiv:1904.02422 (2019)
35. D. Kim, D. Cho, I. S. Kweon, Self-supervised video representation learning with space-time cubic puzzles. Proc. AAAI Conf. Artif. Intell. **33**, 8545–8552 (2019)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.