Directional Residual Frame: Turns the motion information into a static RGB frame

Pengfei Qiu, Yang Zou*, Xiaoqin Zeng, Xiaoxiang Lu, Xiangchen Wu
Institute of Intelligence Science and Technology, School of Computer and Information,
Hohai University, Nanjing, China
{qiupf0718, yzou, xzeng}@hhu.edu.cn

Abstract*—The most commonly adopted methods of video action recognition are optical flow and 3D convolution. Optical flow method requires calculation in advance and a lot of computing resources. 3D convolution method encounters several problems such as many parameters, difficult training, and redundant computation. This paper proposes an approach that can turn the motion information into a static RGB frame by a feasible way of compression: Directional Residual Frame (DRF). This idea comes from a static cartoon that can represent complex events through residual shadows. DRF takes advantage of the scarce nature of residual frames in space and pixel value to achieve similar effects of residual shadows by fusing multiple residual frames. With the DRF, the motion information can be learnt as simply and efficiently as learning the RGB information. In addition, it also proposes a Short-term Residual Shadow Module based on the DRF. Experimental results show that it has better performance than the state-of-the-art model TDN on UCF101 benchmark.

Keywords-DRF; motion information; action recognition; temporal difference module

I. INTRODUCTION

In recent years, Video-based action recognition has drawn a significant amount of attention from the academic community. In action recognition, there are two kinds of key and complementary information: appearances and motion. CNN have achieved great success in classifying images of objects, scenes, and complex events. Thus, it is crucial for action recognition to capture motion information in video, which is usually achieved by two kinds of mechanisms in the current deep learning approaches: two-stream network [1] and 3D convolutions [5,6,7]. Even though the two-stream network can effectively improve the accuracy of action recognition through the optical flow, it requires a lot of computing resources to extract the optical flow. Although the 3D convolution can learn motion features directly from the RGB frames, it also leads to large network models and high computational cost.

Therefore, how to efficiently learn motion information has been a crucial challenge in action recognition.

In everyday life, we can know the motion information of the meteor, the fan and other things through the residual shadow. Obviously, we acquire the motion information from a certain moment of spatial information. Think about it the other way. Can we use a 2D frame to characterize a complex movement process? The optical flow can only reflect the speed, and requires multiple pieces to characterize the non-linear motion. Comics are a very successful case in point. A cartoon can represent a wonderful fight by using a residual shadow. The shadow in static cartoons can be well characterized in the complex motion process. And temporal derivative (difference) is highly relevant to optical flow [2], and has shown effectiveness in action recognition by using RGB difference as an approximate motion representation [3, 4].

In this paper, we propose a motion representation approach based on RGB difference, termed as Directional Residual Frame (DRF). The principle of DRF is similar to the shadow in comics that turns the motion information into a static RGB frame. First, we subtract two adjacent frames in the video with absolute value to obtain residual frames [11]. Then, the residual frames are binarized. During the binarization process, the motion features are retained and the difference caused by the brightness change is removed. Finally, the adjacent residual frames are fused to form a residual shadow-like trajectory map. As shown in Figure 1, our DRF is a good representation of the trees moving to the right (the movement caused by the camera movement) and the people running to the left.

Figure 1. The first 5 frames are consecutive frames in the video, and the sixth frame is the corresponding DRF.

To demonstrate the effectiveness of the DRF, we performed the experimental analysis using Temporal Difference Network (TDN) [12] on the benchmark UCF101 [13], which is the state-of-the-art method without optical flow and 3D convolution. TDN is able to yield a state-of-the-art performance on both motion relevant Something-Something V1 datasets [9] and

*Corresponding author: yzou@hhu.edu.cn(Y. Zou)

DOI reference number: 10.18293/DMSVIVA23-007

scene relevant Kinetics datasets [10], under the setting of using similar backbones[12].

The technical contributions of the paper are summarized as follows:

1) To reduce the serious redundant calculation in video understanding, we propose an effective compression approach DRF that can turn the motion information into a static RGB information by using the scarcity of residual frame, due to the high similarity between adjacent frames. Optical flow requires multiple stacks to react non-linear motion, whereas DRF demands only one.

2) Based on the DRF, we propose a Short-term Residual Shadow Module (S-RSM) to capture the motion information.

3) The experimental results show that compared with the S-TDM in the state-of-the-art model TDN, our approach achieves higher accuracy with fewer model parameters.

The rest of the paper is organized as follows. Section 2 proposes the concept and calculation process of the DRF, and presents the S-RSM module based on the DRF; Section 3 describes the details of the experiments and evaluates the effectiveness of our method on UCF101 benchmark; and Section 4 concludes the paper.

II. DIRECTIONAL RESIDUAL FRAME

In this section, we describe the proposed DRF in detail. First, we give an overview of DRF. Then, we elaborate the calculation process of the DRF. Finally, we provide the implementation details of using DRF in TDN.

A. Overview

Residual shadow is the most successful case that turns RGB information into the motion information. Residual shadow has both motion trajectory information and direction information. So how to form a residual shadow from continuous frames is a challenge. Our thoughts of addressing this include two steps, as follows:

First, motion detection. In this step, the motion information is extracted from the sequential RGB frame. Objects undergoing spatial position changes in the image sequence are presented as foreground (motion region).

Second, motion fusion. In this step, the motion information extracted from the previous step is fused into a static RGB frame where the motion region blurs with time, like residual shadow.

Motion detection. The common methods for motion detection are: background subtraction, temporal difference and optical flow [17]. Both background subtraction and optical flow require a lot of computing resources, which are contrary to efficiency. Therefore, we adopt the temporal difference method to extract the motion object. The temporal difference method may mistakenly detect the area originally covered by the object as moving, called Ghost, which is a problem with motion detection. As shown in Figure 2, the area originally covered by the moving object will be incorrectly detected into motion, which is the Ghost. But Ghost will not be a problem here, because it can be used effectively in motion fusion.

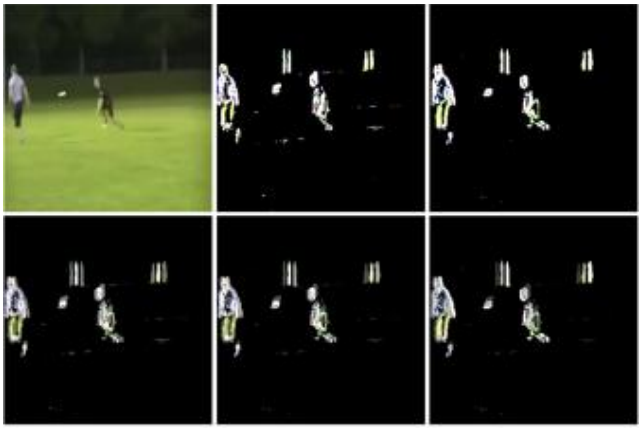


Figure 2. Motion Detection. The second and third frames are the two consecutive temporal differences before the first RGB frame. The fourth frame is (df1 + df2), the fifth is (2 * df2 + df1), and the sixth is the DRF, where df1 is Frame 2, and df2 is Frame 3.

Motion fusion. Motion fusion is the core of the proposed approach.

How to represent the direction of the movement is a crucial issue. In Figure 2, although the fourth and fifth frame preserve more motion traces with the scarcity of the residual frame, there is not any temporal information (direction information). The direction of motion is recognized with the aid of the numerical growth direction. In DRF, objects move from dark to light. From the sixth frame in Figure 2, it can be easily seen through residual shadow that the trees are moving to the right.

$$\begin{bmatrix}0&1&0&0\\1&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}+2*\begin{bmatrix}0&1&0&1\\1&0&0&1\\0&0&0&0\\0&0&0&1\end{bmatrix}+2^2*\begin{bmatrix}1&0&0&1\\1&0&1&0\\0&1&0&0\\0&0&0&0\end{bmatrix}=\begin{bmatrix}4&3&0&6\\7&0&5&2\\0&5&0&0\\0&0&0&3\end{bmatrix}$$

Figure 3. Binary fusion. Every matrix of 5 * 5 represents a residual frame, and the region of the matrix with an element value of 1 indicates the foreground.

Another issue is how to preserve the complete information in the motion fusion process. Although the residual frame is scarce, the foreground of different residual frames may overlap. The overlapping region of the foreground of two adjacent residual frames is the Ghost of the latter residual frame. As for more than two frames, the overlapping region will become difficult to interpret. The overcoverage approach, where the overlapping region takes the same value as the late residual frame, would lose a lot of information. Our approach is inspired by the binary coding to use the value-domain scarce nature of the residual frame. Each number of pixel values indicates an overlapping possibility. In Figure 3, the region with an element value of 7 in the resulting matrix is the overlapping region of 3 matrices; and the region with a value of 5 is the overlapping region of the first and third matrices.

B. The calculation process of the DRF

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and

others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

The process of calculating the RDF is divided into three steps. First, Temporal difference is employed to remove background and acquire motion region. Then, the binarization is adopted to remove the noise and obtain the scarce residual frame. Finally, the DRF is obtained from the fusion of residual frames.

Step 1: Residual frames

Residual frames contain more motion-specific features by removing still objects and background information and leaving mainly the changes between frames [11]. As shown in the third frame in Figure 4, the movement region will be brighter than the static areas.

The movement regions in the residual frame achieve positive or negative values, which are highly correlated with the pixel value of the background. The correlation can cause the movement regions being either positive or negative in the residual frame, thus failing to know the direction of the object. In Figure 4, the Frisbee is white with values above the background color, so it moves from the negative to the positive area in the residual frame. But this direction of movement is unreliable. Therefore, we utilize the absolute residual frame to alleviate the interference of the pixel value of the background. The issue of motion direction in the absolute residual frame will be tackled in step 3.

Figure 4. The first three frames are adjacent frames, the fourth one is the corresponding residual frame, the fifth one is the residual frame after the absolute value, and the sixth one is a binarization of the fourth one.

Here we use $Frame_i$ to represent the *ith* frame data, and $Frame_{i\sim j}$ denotes the stacked frames from the *ith* frame to the *jth* frame. The process of obtaining residual frames can be formulated as follows:

$$ResFrame_{i\sim j} = \left| Frame_{i\sim j} - Frame_{i+1\sim j+1} \right|$$

At this stage, the Residual frames is not a sparse matrix. Influenced by the camera motion and light intensity changes, the gray area is not all *0*.

Step 2: Binarization

Binarization of the residual frames: *0* is used to represent no change area, and *1* is used to represent change area. In the field of image segmentation, there are a few algorithms [14] for image binarization. In this paper, we adopt threshold method in order to reduce the amount of computation as much as possible.

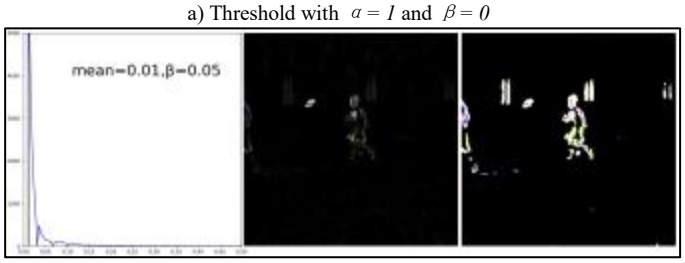
The formula is as follows:

$$threshold = \frac{\alpha}{n^2} * \sum_{x=1}^{n}\sum_{y=1}^{n} ResFrame_i(x,y) + \beta \quad (1)$$

$$ResFrameB_i(x,y) = \begin{cases} 1 & ResFrame_i(x,y) > threshold \\ 0 & otherwise \end{cases} \quad (2)$$

Here $ResFramei(x,y)$ is the image value of the coordinates (x, y) in the *ith* residual frame. α and β are super parameters. And n is the size of the *ith* residual frame.

a) Threshold with $\alpha = 1$ and $\beta = 0$



b) Threshold with $\alpha = 1$ and $\beta = 0.05$

Figure 5. Binarization. The "mean" in the images represents the mean of the residual frames. The first chart of each row is the pixel value statistics chart, the second is the residual frame, and the third is the binarized residual frame.

Since residual frame is a scarcity matrix, the mean value tends to be below the minimum in the movement region. With very small amplitude of motion, the mean may be lower than the difference value caused by changes in light intensity. We denote the minimum of the threshold by β. Figure 5 illustrates the effect of β on removing the background noise.

Step 3: Motion fusion. The higher the value is, the later the event occurs.

The motion fusion of multiple binary residual frames transforms temporal information into numerical information. The higher the value is, the later the event occurs.

$$DRFrame_n^{'} = \sum_{i=1}^{n} 2^{i-1} * ResFrameB_i \quad (3)$$

We accumulate the residual frames according to formula 3. There may be overlaps between the differences of consecutive residual frames. Various overlapping cases of n residual frames will be mapped to the value $0\sim 2^n$. The case with $n=4$ is shown in Figure 6. In Figure 3, the region in the result matrix corresponding to the motion region (value is *1*) in the third

matrix should obtain the maximum value to indicate the movement end point. But the value of overlapping region will be greater than the last motion region. The brightest region appears in the middle region of the motion trajectory, as in the fifth frame of Figure 2.

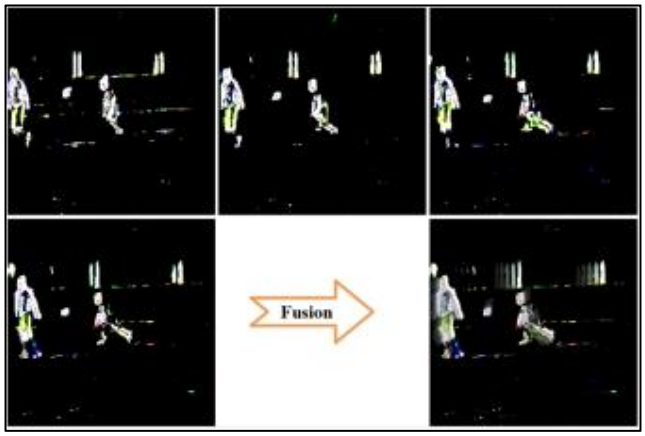


Figure 6. The first four pictures are continuous residual frames after binarization, and the last one is the DRF fused by the first four.

$$Mask_i(x,y)=\begin{cases}1 & DRFrame_n'(x,y)=2^{i-1}\\ 0 & otherwise\end{cases} \quad (4)$$

$$DRFame_n = DRFame_n' + \sum_{i=1}^{n} 2^{i-1}\left(ResFrameB_i * Mask\right) \quad (5)$$

In Formula 4, the operator sets the non-zero element in the matrix to 0 and the zero element to 1. Then, through Formula 5, the value of the non-overlapping difference is doubled.

C. S-RSM with DRF

Based on the DRF, a Short-term Residual Shadow Module (S-RSM) is proposed, as an improvement of the S-TDM in TDN, as illustrated in Figure 8.

Temporal Difference Networks (TDN) is a video-level architecture of capturing both short-term and long-term information for end-to-end action recognition. TDN is composed of a short-term and long-term temporal difference module (TDM), as illustrated in Figure 7 [12]. In Figure 9, the short-term TDM in TDN supply a single RGB frame with a temporal difference to yield an efficient video representation, explicitly encoding both appearance and motion information [12].

In TDN, the stacks of difference frames are processed by 2D convolution, which can only capture limited motion information and of which the main function is to calibrate the moving area on the static image.

The DRF turns the action information into the static RGB information by fusing multiple temporal difference frames. So the model can capture the movement information by learning the RGB information in the DRF. This feature of DRF is beneficial to 2D convolutional networks to learn motion features, so as to perform the task of action recognition even better.

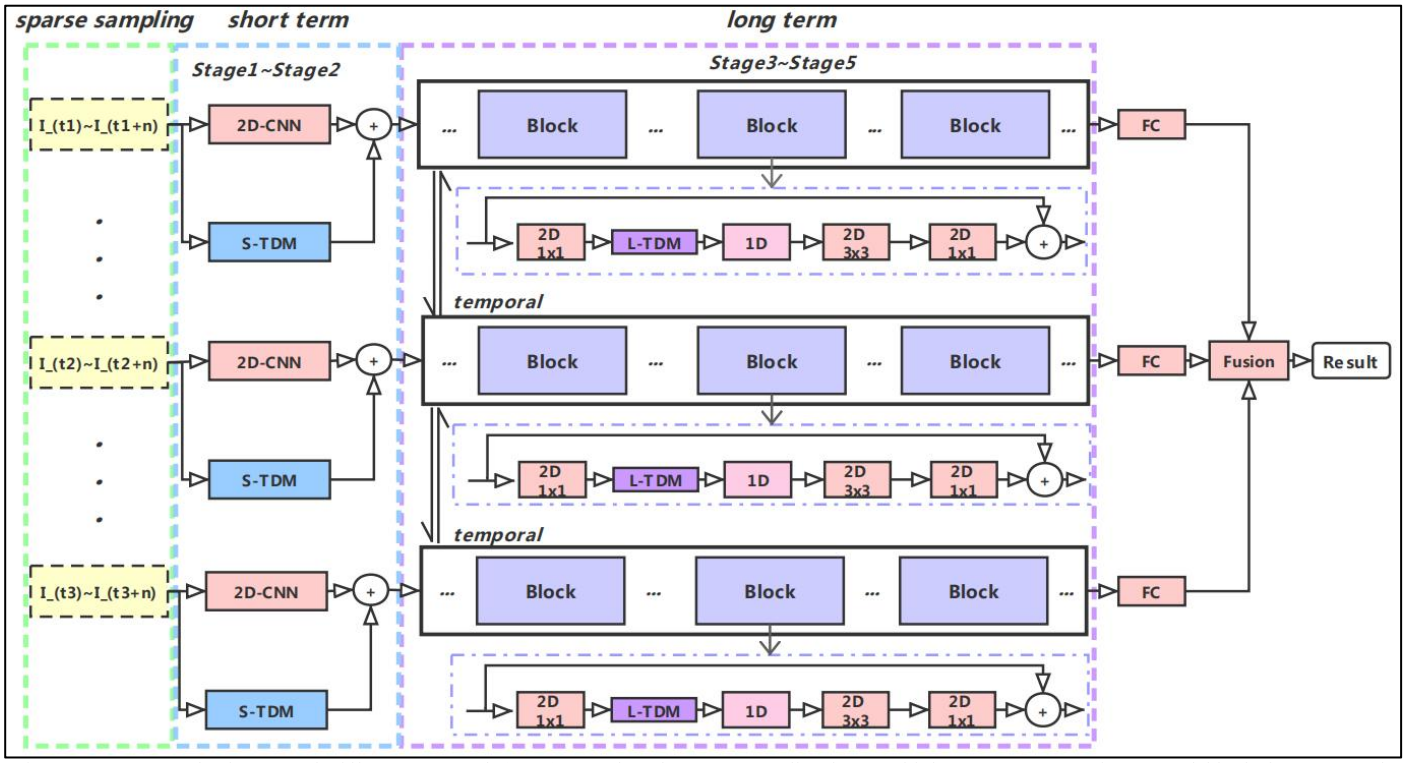


Figure 7. Framework of Temporal Difference Network (TDN). Based on the sparse sampling from multiple segments, TDN aims to model both short-term and long-term motion information.

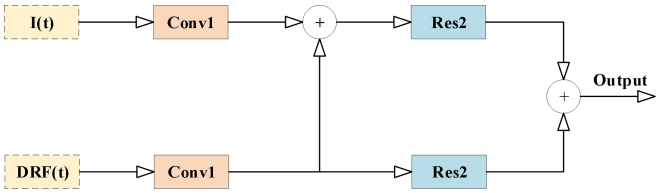


Figure 8. Framework of short-term Residual Shadow Module with DRF

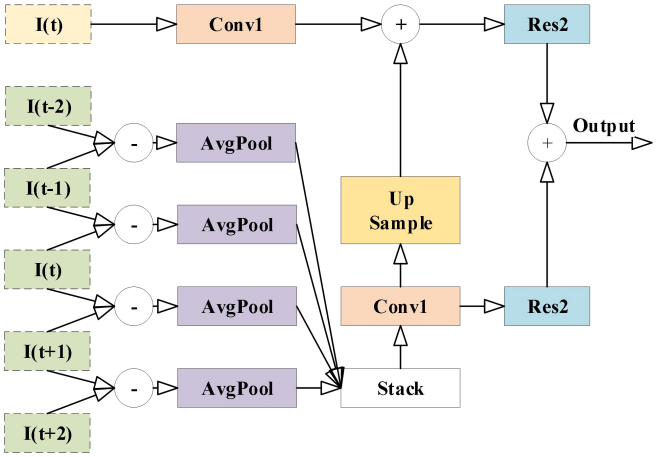


Figure 9. Framework of the short-term TDM.

III. EXPERIMENTS

In this section, we present the experiment results of the proposed DRF. First, we describe the evaluation datasets and implementation details. Then, we evaluated the effectiveness of DRF on the state-of-the-art method TDN.

A. Datasets and implementation details

Video datasets. There are several commonly used datasets for video recognition tasks. We mainly focus on the benchmark: UCF101. UCF101 consist of 13,320 videos in 101 action categories [13].

Training and testing. In experiments, we use ResNet50 to implement TDN framework. Following common practice [15], during training, each video frame is resized to have shorter side in [256, 320] and a crop of 244 × 244 is randomly cropped. We pre-train TDN on the ImageNet dataset [16]. The batch size is 128 and the initial learning rate is 0.02. The total training epoch is set to 60 in the UCF101 benchmark. The learning rate will be divided by a factor of 10 when the performance on validation set saturates. For testing, the shorter side of each video is resized to 256. We implement the kind of testing scheme: 1-clip and center-crop where only 1 center crop of 244 × 244 from a single clip is used for evaluation.

B. Experimental Results

From the experimental results in Table I, it can be found that the *beta* of DRF taking 0.05 is a suitable value. The binarized threshold as the mean has lower accuracy than the other two schemes, which confirms the viewpoint we mentioned in Section 2. With very small amplitude of motion, the mean may be lower than the difference value caused by changes in light intensity.

Since the residual frame is absolute, the *beta* of DRF is the lower limit of the threshold. The accuracy of the *beta* of DRF being 0.05 is higher than that of the *beta* of DRF being 0.1. When the threshold is set too high, some important motion information will be filtered out.

TABLE I. ACC OF DIFFERENT BINARIZATION PARAMETERS ON UCF101 BENCHMARK

Method	Backbone	Input (S-RSM)	Frames	Length (DRF)	Alpha (DRF)	Beta (DRF)	Top-1 (UCF101)
TDN	ResNet50	DRF	3	5	1.00	0.00	84.51%
TDN	ResNet50	DRF	3	5	1.00	0.05	85.33%
TDN	ResNet50	DRF	3	5	1.00	0.10	84.92%

From the experimental results in Table Ⅱ, it can be shown that the frames of the DRF motion fusion is of length 5 in UCF101 benchmark. When the DRF length is set to 3, the reason for the decrease of accuracy is that TDN learns too little action information, whereas it is set to 7 and 9, the reason for the decrease of accuracy is that it is difficult for TDN to learn.

TABLE II. ACC OF DIFFERENT MOTION FUSION LENGTH ON UCF101 BENCHMARK

Method	Backbone	Input (S-RSM)	Frames	Length (DRF)	Alpha (DRF)	Beta (DRF)	Top-1 (UCF101)
TDN	ResNet50	DRF	3	3	1.00	0.05	84.29%
TDN	ResNet50	DRF	3	5	1.00	0.05	85.33%
TDN	ResNet50	DRF	3	7	1.00	0.05	84.48%
TDN	ResNet50	DRF	3	9	1.00	0.05	84.48%

The results in Table Ⅲ show that the proposed TDN outperforms the original model at sampling frames of 4 and 8. With the sample frame of 4, our approach improves by nearly 1% over the original method; and with the sample frame of 8, our approach improves by more than 1.2%.

TABLE III. ACC OF DIFFERENT MODULE ON UCF101 BENCHMARK

Method	Backbone	Input	module	Frames	Pretrain	Top-1 (UCF101)
TDN (original)	ResNet50	RGB + difference	S-TDM	4	ImageNet	84.97%
TDN (original)	ResNet50	RGB + difference	S-TDM	8	ImageNet	87.15%
TDN (ours)	ResNet50	RGB + DRF	S-RSM	4	ImageNet	85.95%
TDN (ours)	ResNet50	RGB + DRF	S-RSM	8	ImageNet	88.39%

From this set of comparative experiments, it can be concluded that DRF contains better motion information than the stacked residual frames. In [12], it has been shown that TDN can reach the state-of-the-art level without the use of optical flow and 3D convolution.

IV. CONCLUSION

To address the problem of serious redundant calculation in video motion recognition, we propose the approach to

squeezing the motion information into a RGB frame. The principle of DRF is similar to the shadow in comics. The shadow in static cartoons can be well characterized in the complex motion process. DRF exploits the scarcity of residual maps to fuse the motion information of multiple residual maps into one spatial frame. In this way, it can learn motion information as it learns about RGB information. Based on the DRF, we propose S-RSM based on 2D convolution to capture motion information. Through comparative experiments, we verified that our approach has better performance than the state-of-the-art model TDN in UCF101 benchmark.

REFERENCES

[1] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in NIPS, 2014, pp. 568-576.

[2] Berthold K.P. Horn and Brian G. Schunck, "Determining optical flow," Artificial Intelligence, vol.17, pp. 185-203, 1981.

[3] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. lin, X. Tang and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," European conference on computer vision, vol. 9912, pp. 20-36, 2016.

[4] Z. Yue, Y. Xiong, and D. Lin, "Recognize Actions by Disentangling Components of Dynamics," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 6566-6575.

[5] S. Ji, W. Xu, M. Yang and K. Yu, "3D convolutional neural networks for human action recognition." IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 35, pp. 221-231, 2013.

[6] D. Tran, L. Bourdev, R. Fergus, L. Torresani and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," Proceedings of the IEEE international conference on computer vision, 2015, pp. 4489-4497.

[7] K. Hara, H. Kataoka, and Y. Satoh, “Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018, pp. 6546–6555.

[8] J. Lin, C. Gan, and S. Han, "TSM: Temporal Shift Module for Efficient Video Understanding," IEEE/CVF International Conference on Computer Vision (ICCV) IEEE, 2019, pp. 7082-7092.

[9] J. Carreira, and A. Zisserman, "Quo vadis, action recognition? a new model and the Kinetics Dataset," Computer Vision and Pattern Recognition IEEE, 2017, pp. 4724-4733.

[10] R. Goyal, SE. Kahou, V. Michalski, J. Materzynska and R. Memisevic, "The" something something" video database for learning and evaluating visual common sense," Proceedings of the IEEE international conference on computer vision, 2017, pp. 5843-5851.

[11] L. Tao, X. Wang, and T. Yamasaki, "Rethinking motion representation: Residual frames with 3D ConvNets," IEEE Transactions on Image Processing, vol. 30, pp. 9231-9244, 2021.

[12] L. Wang, Z. Tong, B. Ji and G. Wu, "TDN: Temporal Difference Networks for Efficient Action Recognition," Computer Vision and Pattern Recognition IEEE, 2021, pp. 1895-1904.

[13] M. S. Hutchinson and V. N. Gadepally, "Video action understanding," IEEE Access, vol. 9, pp. 134611-134637, 2021.

[14] Otsu, N. "A Threshold Selection Method from Gray-Level Histograms," IEEE Transactions on Systems Man & Cybernetics, vol. 9, pp. 62-66, 2007.

[15] C. Feichtenhofer, H. Fan, J. Malik and K. He, "Slowfast networks for video recognition," Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 6201-6210.

[16] L. J. Li, R. Socher and F. F. Li, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework," IEEE Conference on Computer Vision & Pattern Recognition IEEE, 2009, pp. 2036-2043.

[17] A. A. Shafie, F. Hafiz and M. H. Ali, "Motion detection techniques using optical flow," World Academy of Science Engineering & Technology, 2009