IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 6, 2021, accepted June 6, 2021, date of publication June 11, 2021, date of current version June 29, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3088155

Human Action Recognition Based on Motion Feature and Manifold Learning

JUN WANG[1], LIMIN XIA[2], AND WENTAO MA[2]
[1]Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China
[2]School of Automation, Central South University, Changsha 410083, China

Corresponding author: Wentao Ma (194611057@csu.edu.cn)

This work was supported by the Hunan Science and Technology Project under Grant 2017GK2271.

ABSTRACT Human action recognition is an important task in the fields of video content analysis and computer vision. Since the performance of most existing action recognition frameworks depends on the representation of features, many researches aim to construct more discriminative features. In this paper, we propose a manifold learning framework based on optical flow for action recognition. First, we calculate the dense optical flow field of the original video sequence, and the attention pooling layer (AP) is adopted to separate target area and background area to eliminate background interference. On this basis, motion features (MF) based on the physical characteristics of dense optical flow are developed to characterize human motion information. After that, manifold learning is introduced to calculate the motion variance features (MVF), which reflect the change rate of motion features and measure the spatial correlation between features in non-Euclidean space. Finally, fusing the MVF obtained by manifold learning and MF, feeding fusion features into two fully connected layers (FC) in series for action classification and recognition. Experiments on several classic datasets show that the proposed method achieves 0.98%, 1.86% and 0.99% performance improvement on UCF 101, HMDB51 and JHMDB.

INDEX TERMS Action recognition, attention pooling, manifold learning, motion features, motion variance features.

I. INTRODUCTION

The purpose of human action recognition (HAR) is to realize understanding of human behavior by analysing and processing the video containing human behavior. Although the research of HAR has made significant progress in image segmentation [1]–[4], target detection [5]–[8] and etc., it is still confronted with a great challenge because of the diversity and high non-linearity of human behavior, which is caused by the non-rigid structure of human body and the confusion of background and motion feature, etc.

At present, the mainstream action recognition framework is mainly limited by the following three aspects: (1) deep learning framework often needs to be trained with a large number of parameters, which is easy to fall into the disaster of dimensionality; (2) Due to the one-sidedness of manual features, its recognition ability is not enough to characterize motion states; (3) the intense interference caused by complex background confuses the recognition model.

To deal with these issues, we propose a novel framework for action recognition. In our framework, we first calculate the dense optical flow field for subsequent feature processing and extraction. And then, the attention pooling layer (AP) is inserted into the traditional 3-layer CNN structure to capture the region of interest (ROI) in continuous video frames. The purpose of this step is to reduce the interference caused by the background, and reduce the computational burden, effectively. On this basis, the divergence and curl information of the optical flow field is calculated to measure the change of the original dense flow field, which are accumulated as MF. After that, the Riemannian manifold learning method is applied to calculate the degree of spatial motion variation between the feature vectors of different frames to obtain the motion variation features (MVF). We think this helps to make up for the shortcomings of MF we calculated previously that cannot take into account the spatial position changes of the motion parts. Finally, concatenating MF with the motion

The associate editor coordinating the review of this manuscript and approving it for publication was Huiyu Zhou.

variation features, and then fed them into two fully connected layers to complete the action recognition tasks.

To sum up, our contribution can be summarized as the following four aspects:

1) We propose an effective action recognition framework under complex background. Compared with most mainstream action recognition methods, our proposed framework combined with manifold learning can accurately identify the types of behaviors with low computational cost.
2) The attention pooling layer (AP) is introduced into the framework to capture the region of interest (ROI) of continuous video frames, which can eliminate the interference caused by complex background and reduce the computational burden.
3) In the field of action recognition, we innovatively propose a motion feature (MF) accumulated and calculated based on physical characteristics of the flow field that can represent the target motion state, including the divergence and curl characteristics of optical flow. The effectiveness of MF has been verified in Ablation Study.
4) Manifold learning method is developed to calculate the motion variance features, which measure the motion change rate between feature vectors. Concretely, the projection algorithm of unit n-sphere is applied to map MF into a Non-European Spatial manifold to count the changes in Non-European spatial position of body parts participating in motion.
5) The method proposed in this paper has achieved competitive performance with the other state-of-the-art methods on 5 benchmark datasets. Concretely, mean Average Precision (mAP) is applied as a measurement standard, and the proposed framework improves by 0.09% on the UCF 101 dataset, 1.12% on HMDB51 and 0.66% on the JHMDB dataset.

The rest of this paper is organized as follows. Section II states the current research situation and existing problems. Section III gives an introduction to the principles and details of our proposed methods. Some the experimental details and results are given in Section IV. Finally, Section V presents a brief conclusion to this paper.

II. RELATED WORKS

With the development of human action recognition technology, HAR has been widely used in all walks of life. This also puts forward higher and higher requirements for the accuracy and anti-interference ability of HAR. On this basis, a large number of scholars have carried out extensive research on the problems.

A. HUMAN ACTION RECOGNITION BASED ON MANUAL FEATURE

In the past few years, although deep learning related methods have made breakthrough progress, but due to its data-driven characteristics resulting in a high degree of data dependence [9], this series of methods still have defects. This also means that the traditional method of designing manual features is still reasonable. In this series of methods, the recognition performance of HAR mainly depends on the discriminative of the designed features.

1) METHODS BASED ON TRAJECTORIES

Wang *et al.* [10] proposed to sample feature points on the dense grid of each frame and use optical flow algorithm to track them. Through different scale sampling, the points of each frame are connected to form dense trajectories (DTs). This method is proved to be an effective method to combine dense sampling with feature tracking to reduce the information loss caused by sparse interest points. Then, considering the influence of camera motion on the recognition results, Wang and Schmid [11] also proposed an improved dense trajectory (iDTs) and introduced the moving boundary histogram (MBH) to correct the optical flow. This method can effectively reduce the interference trajectories caused by camera motion, but cannot effectively eliminate the trajectories caused by background clutter. Therefore, the recognition results and computing speed of traditional iDT algorithm are bound to be affected by the complex background.

2) METHODS BASED ON OPTICAL FLOW

Jiang *et al.* [12] proposed a method that using neural network and single stream long-term optical flow convolution learning video representation to complete the modeling of the whole frame range of action. Wang *et al.* [13] calculates the motion intensity by accumulating the adjacent optical flow in a time interval, so as to reduce the motion feature displacement caused by image noise. Li *et al.* [14] used RGB video frames to construct optical flow images in order to eliminate the interference background. Xu *et al.* [15] proposed a fast human action recognition network, which improves the efficiency of optical flow feature extraction by exploring the method of spatiotemporal feature fusion. Yi *et al.* [16] proposed a new method based on optical flow to compute saliency map to highlight foreground motion region. Besides, based on traditional local descriptors (including directional gradient histogram, optical flow histogram and motion boundary histogram), the correlation between trajectory and target motion is considered. Tanberk *et al.* [17] using 3D-CNN and LSTM to classify and analyze the optical flow of video sequence.

Although the above method overcomes the disadvantage of deep learning method relying on a large number of data samples training to a certain extent, it is difficult to get rid of the trouble caused by the high one sidedness of manual features. Therefore, we need to further explore the information contained in video sequences and manual features.

B. HUMAN ACTION RECOGNITION BASED ON MANIFOLD LEARNING

In the manifold learning method, human action modeling based on video is a hot topic. In addition to the

classical methods in Euclidean space, various methods based on manifold analysis have been proposed in recent years. Abdelkader *et al.* [18] expressed each pose contour as a point in the closed curve shape space, and each pose as a trajectory in the space. Gong and Medioni [19] proposed a spatio-temporal manifold (STM) model to analyze nonlinear multivariable time series with potential spatial structure, and applied it to motion recognition in joint trajectory space. Based on STM, they proposed a dynamic manifold distortion (DMW) and motion similarity measure to compare human motion sequences extracted from images using 2D tracker in two-dimensional space and human motion sequences extracted from images using motion capture data in three-dimensional space. Gall *et al.* [20] coupled action recognition and 3D pose estimation on 2D images using a multi view system, where action specific manifolds act as links between them. Slama *et al.* [21] used a dynamic system whose observability matrix is a Grassmann manifold to model and analyze human motion accurately. Carrillo *et al.* [22] uses the covariance matrix to gather a set of local trajectories based on optical flow in space to characterize the action, and then builds a Riemannian manifold describing the motion through the set of frame-level covariance matrices.

The manifold learning method can measure the position relationship of feature vector in higher dimension because it gets rid of the limitation of Euclidean space [20]. This enables us to more effectively use manual features for accurate behavior modelling.

III. PROPOSED METHOD

The proposed framework is illustrated in Fig. 1. In our method, we first divide the whole video sequence into several blocks, each block contains 15 consecutive video frames, to facilitate subsequent processing (The last part of the video sequence with less than 15 frames will be filled with a number of white frames to make a total of 15 frames and counted as a block). And then, the AP is applied to capture the Region of Interest (ROI) of each block, which helps to weaken the influence of background interference (as shown in the inner part of the red dotted box in the figure). After that, calculate the dense optical flow field of the ROI in each block and the physical characteristics of this block (as shown in the inner part of the yellow dotted box), the physical characteristics of each block are accumulated to obtain the MF we need later. On this basis, unit n-sphere projection algorithm is applied to map the feature vector normalized by $L2$-norm of physical characteristics into Non-European Spatial manifold (as show in the inner part of the blue dotted box) and calculate the motion change rate on Non-Euclidean space of different feature vector as MVF. Finally, concatenating the MF and MVF, the final fusion features are fed into two fully connected layers for classification.

A. ATTENTION POOLING

In this subsection, we describe the principle and implementation details of the AP. What we focus in this subsection is to capture the region of interest of consecutive frames of video, rather than conduct accurate action recognition. To achieve this goal, we improve the method proposed in [23] to capture ROI.

Scholars have proved that the second-order statistical information is helpful to fine-grained classification in [23], [24]. At present, the most commonly used classification method is to extract feature vectors f, and then get the final classification score through the learning and training of f. Different from the above method, [23] applies the second-order information to calculate the classification score. Specifically, the feature vector f^2 is obtained by vectorization, and then the classification score of the feature matrix f^2 is learned and calculated. In order to facilitate the subsequent processing, we adopt the inner product of vector instead of f^2, that is, $\mathrm{Tr}\left(AB^T\right) = \mathrm{dot}(A(\cdot), B(\cdot))$. Where $A(\cdot)$ and $B(\cdot)$ denote the elements of matrix. Assume that the layer to be pooled is $X \in R^{n\times f}$, where f is the number of channel number, n is the number of spatial locations. On this basis, we assume that the attention matrix is $W \in R^{f\times f}$, So far, the score based on attention pooling can be calculated as follows:

$$score_{attention}(X) = \mathrm{Tr}(X^T X W^T) \tag{1}$$

To reduce the dimension of matrix W, thus reduce the computational burden, the second-order low-rank approximation method is applied to approximate W, that is, $W = ab^T$. We replace W in Eq.(1) with ab^T, then Eq.(1) can be rewritten as:

$$score_{attention}(X) = \mathrm{Tr}(X^T X b a^T) \tag{2}$$

where $X \in R^{n\times f}$, $a, b \in R^{f\times 1}$. On this basis, according to $\mathrm{Tr}(ABC) = \mathrm{Tr}(CAB)$, Eq.(2) can be rewritten as:

$$score_{attention}(X) = \mathrm{Tr}(a^T X^T X b) \tag{3}$$

It is not difficult to observe that the result of the operation in $\mathrm{Tr}(\cdot)$ in Eq.(3) is a scalar. Therefore, Eq.(3) can be rewritten as:

$$score_{attention}(X) = \mathrm{Tr}(Xa)^T (Xb) \tag{4}$$

In fact, the form of Eq.(4) can be regarded as symmetrical, that is, the final score can be regarded as the inner product between two attention heat maps of all feature sampling points. But it is worth noting that Eq.(4) can only be applied for binary classification tasks, and a certain degree of improvement needs to be made when facing multi-classification tasks. That is to replace the weight matrix W in Eq.(1) with the class-specific weight matrix $W_k = a_k b_k^T$, So that Eq.(4) can be rewritten as:

$$score_{attention}(X) = \mathrm{Tr}(a_k^T X^T X b) = (Xa_k)^T (Xb) \tag{5}$$

On this basis, we build the AP layer based on Eq.(5). $W_k = a_k b$ is the parameter of the AP layer, which is the goal we want to optimize. Embedding the AP in a three-layer CNN structure (3 convolutional layers + 3 average pooling layers + 1 fully connected layer), that is, insert the AP

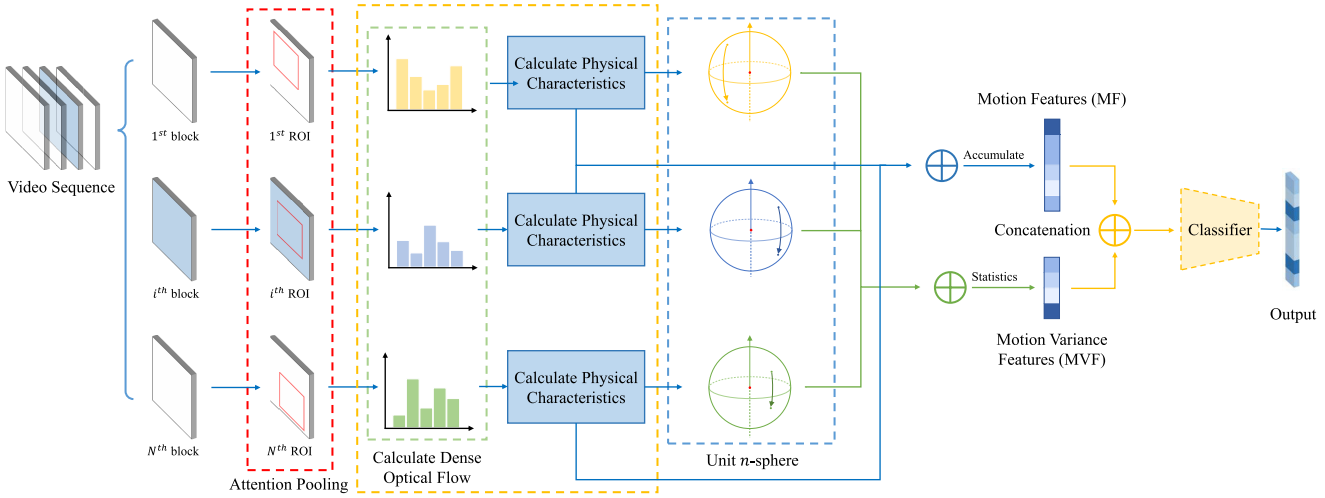

FIGURE 1. The overview of proposed framework. In this framework, the video sequence is divided firstly, and the ROI of each divided block is extracted by the AP. On this basis, modeling the moving target and constructing MF. In addition, measuring the Non-Euclidean spatial change rate of MF to calculate MVF. Finally, fusing the MF and MVF and feeding them into classifier for recognition task.

between the fully connected layer in the CNN and the last set of convolutional pooling layers, and adopting the action label and sampling frame to train CNN with AP. After the model converges, perform deconvolution on the feature map calculated by W_k to obtain the final mask-level attention matrix W_a, which is adopted to extract the ROI of original video sequence directly.

B. CONSTRUCTION OF MOTION FEATURES

In this subsection, we describe how to construct MF based on optical flow in ROI. After the above processing, we get a region of interest that only contains the main motion region of the target, and on this basis, we continue to extract physical features.

Inspired by [25], we noticed that when the target is in motion, the topological relationship of the optical flow field in the moving area changes to a certain extent, and these changes directly lead to changes in the divergence and curl characteristics of the flow field. Therefore, our innovative attempt to use the physical characteristics of the flow field to model moving targets.

Generally, in mathematics, the calculation of the divergence and curl of the flow field is as follows:

$$\mathrm{div}(\boldsymbol{S}) = \lim_{d\sigma \to 0} \frac{1}{d\sigma} \oint S \cdot dn \tag{6}$$

$$\mathrm{curl}(\boldsymbol{S}) = \lim_{d\sigma \to 0} \frac{1}{d\sigma} \oint S \cdot dr \tag{7}$$

where dn and dr denote the normal vector and tangent vector of the point $\boldsymbol{S}$, respectively. However, since we need to calculate the divergence and curl features in a two-dimensional vector field, we cannot directly calculate them according to Eq.(6) and Eq.(7). Therefore, we need to use the accumulation method to approximate the divergence and curl of the flow field.

Under the guidance of [23], we use the point-state accumulation method to calculate the physical characteristics. Concretely, the flow field map is equidistant sampled, that is to say, there are 8 sampling points adjacent to each central sampling point. The specific sampling process is shown in Fig. 2. For the characteristic points of the central sampling point, the divergence calculation method is as follows:

$$\mathrm{div}(\boldsymbol{S}) \propto \sum_{i=0}^{N-1} s_k \cdot n_k \tag{8}$$

where N is the number of sampling points adjacent to the point $\boldsymbol{U}$, s_k is the normalized motion vector from the point $\boldsymbol{U}$, and n_k is the unit vector pointing to the center point $\boldsymbol{U}$ of the adjacent sampling point, that is, the outer normal direction. Similarly, the curl of point U can also be approximated by accumulating surrounding sampling points:

$$\mathrm{curl}(\boldsymbol{S}) \propto \sum_{i=0}^{N-1} s_k \cdot r_k \tag{9}$$

where r_k is the unit tangent vector on the outer normal.

Obviously, directly counting and calculating all the sampling points in the video frame will produce a very large scale feature map, which will cause too much burden on the subsequent calculation. Similarly, the experiment also shows that the effect of directly sending the feature map to the classifier is very poor when there are fewer neurons in the full connection layer (FC). Therefore, the block is needed to divided into $W \times H$ sub-blocks to get rid of this problem, we first accumulate the divergence and curl value of the sampling points in a sub-block, so that the dimension of original feature map can be reduced to $H \times W$. As for the values of H and W, we will discuss in detail in Section IV-C. In this way, physical characteristics is represented as:

$$f_m{}^j = \{div^j, curl^j\} \in \mathbb{R}^{32\times 64} \tag{10}$$

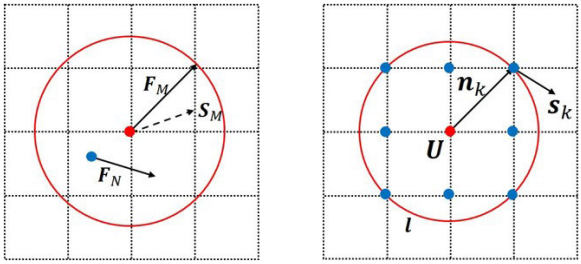

FIGURE 2. The sampling of divergence(left), curl(right) of the particle S in field, the eight blue points are obtained by equidistant sampling.

where j denotes the number of blocks, div^j, $curl^j \in \mathbb{R}^{W\times H\times 2}$ denotes the divergence and curl characteristics of j-th block. Obviously, such a block only contains 15 frames of motion information. In order to get long-term information, we directly accumulate ${f_m}^j$ to obtain the MF we need of the whole video, which is defined as $F_m = \{v_{Div}, v_{Curl}\} \in \mathbb{R}^{32\times 64}$, where v_{Div} and v_{Curl} denotes the accumulated and reshaped results of div^j and $curl^j$, respectively. In this way, F_m can be used to represent the MF of the entire video sequence, that is, the aforementioned MF. In the subsequent processing process, we will fuse the auxiliary features with MF to overcome the shortcomings of a single MF characterization ability is not good enough.

C. MANIFOLD LEARNING FOR MEASURING FEATURE VECTORS

In this subsection, we describe the calculation of the final video representation in detail. In above-mentioned step, we obtain the local region change measure of the moving object, that is, the MF. However, it is not ideal to directly send MF (including divergence and curl characteristics) into the classifier, because this method only considers the change rate of motion size and direction, and does not take into account the change rate of motion position in space. In addition, the MF obtained by this method are limited to represent only short-term video information. In order to solve the above problems, we use the learning method of Riemannian manifold in mathematics to measure the spatial dynamic differences between features over time. As shown in Fig. 3.

In previous work [24], [26], people calculated the position and relative position distance of feature points in different frames to reflect the spatial change rate of feature points. However, with the in-depth study of manifold learning, many scholars gradually realize that feature points are not necessarily distributed in Euclidean space [27], which also means that Euclidean distance $D(x_i, x_j)$ is not suitable for characterizing the spatial change rate of feature points. Under this consideration, some scholars propose to measure the change rate of features by using Riemannian manifolds [24], [26], [27] which conform to the feature distribution. The Riemannian manifold is smooth and satisfies the requirement of measuring the rate of change of behavior, because its inner product in tangent space changes more smoothly to the point. Besides, Riemannian manifold can model human behavior better than other methods due to human action is coherent and smooth.

First of all, what we focus on is to consider construct such a Riemannian manifold that can represent the feature distribution. Because most Riemannian manifolds are complex and spatial-agnostic, we need to map the feature distribution into a spatial-knowable manifold. Specifically, we use $L2$-norm normalization and reshape the MF, so that they are located on a unit hypersphere. Therefore, it is a good choice to analyze the change rate of divergence and curl characteristics on the unit hypersphere as Eq.(11). In this way, such a hypersphere can be applied to simulate the distribution of features in the Riemannian manifold space. Therefore, the change of different features are measured by analyzing the position change rate of different features on the hypersphere.

$$S^n = \{x \in \mathbb{R}^{n+1} \mid \|x\| = 1\} \tag{11}$$

where n denotes the dimension of hypersphere, the unit n-sphere is a kind of hypersphere whose center is at the origin of Euclidean space and radius is 1. In order to transform the divergence and curl characteristics of a moving target into manifold-points (points distributed on the manifold), it is necessary to normalize the $L2$-norm of the feature vector, so as to map the features of the moving target in consecutive frames to the hypersphere, that is, the unit n-sphere. The distance between feature points on manifold can accurately reflect how much changes have taken place in the spatial position of moving target. The distance between two manifold-points on manifold $p_1, p_2 \in S^n$ is calculated by the geodesic distance under Riemannian metric, where the geodesic distance is the great-circle distance $d(p_1, p_2)$ defined as Eq.(12):

$$d(p_1, p_2) = \arccos({p_1}^T p_2) \tag{12}$$

Assume S^{nd} is the manifold of divergence variation analysis and S^{nc} is the manifold of curl variation analysis. Then the reshaped divergence and curl vectors are mapped into manifold-points on their manifolds S^{nd} and S^{nc} respectively. Here we only describe in detail the processing steps of the divergence feature descriptor on the moving manifold. As for the curl feature, it shares the same steps as the divergence feature, so it does not need to be described more. The normalized divergence characteristics vector of div^j is denoted as the point p_t on S^n, the manifold-point of the feature vector generated in the next sampling block is denoted as $p_{(t+1)}$. the moving velocity of block t and $(t+1)$ in manifold is defined as Eq.(13):

$$v_t = \frac{d(p_t, p_{(t+1)})}{\Delta t} \tag{13}$$

For convenience, since the intervals between sampling blocks are equal, we can set Δt to 1. Obviously, the greater the change in motion, the faster the corresponding point moves, and vice versa. In the HAR field, when the human body is in complex motion, its body parts can be divided into the following three situations according to its motion performance: (1) Severely changing parts (including strong

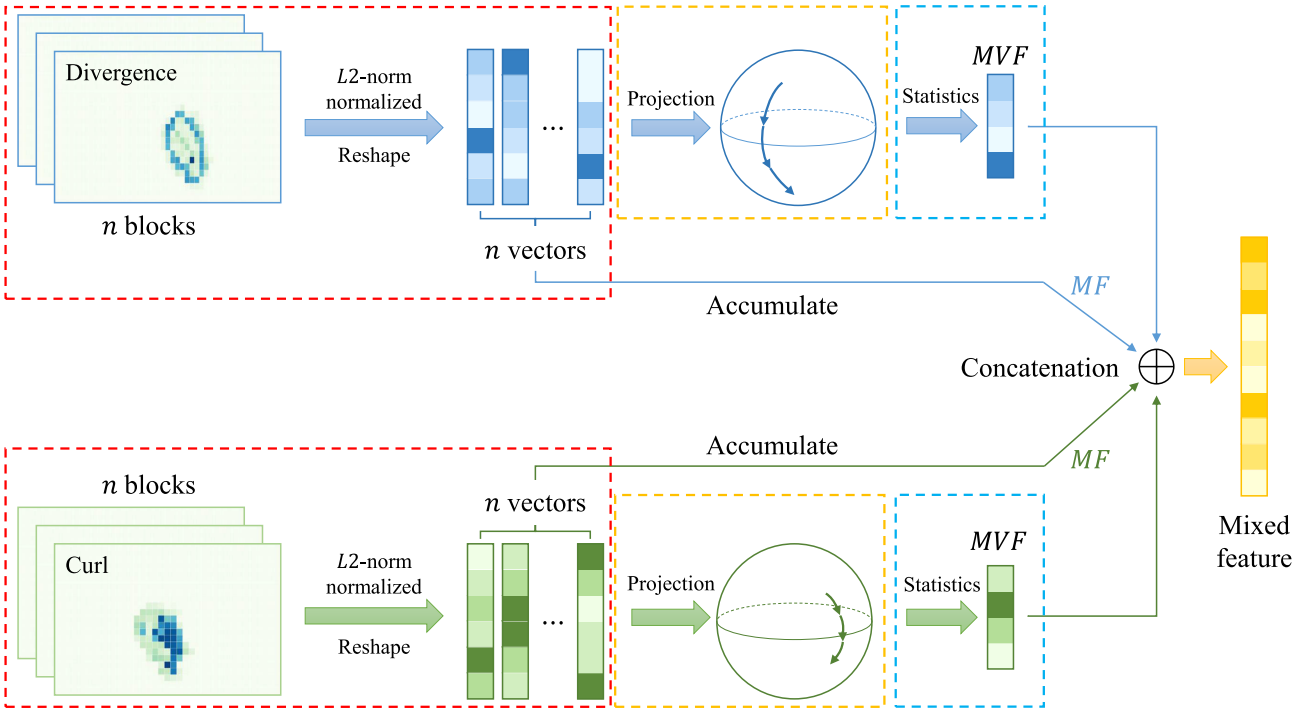

FIGURE 3. The overview of unit *n*-sphere projection algorithm. Firstly, $L2-norm$ is adopted to normalize the physical characteristics and reshape them into *n* feature vectors (as shown in the red dotted box). Then, the normalized physical characteristics of every two adjacent blocks are mapped to *n*-unit spheres, and their geodesic distances are calculated. In this way, we can obtain $(n-1)$ distance measures in non-Euclidean space for the divergence and curl characteristics respectively (as shown in the yellow dotted box). On this basis, in order to further extract the rate of change between features, we use mathematical statistics for the above distance measurement, using the maximum, minimum, mean and variance of the distance measurement to measure the rate of change between features to obtain the MVF (as shown in the blue dotted box).

rotational and translational movements and a large number of irregular movements); (2) Slowly changing parts (including a single translational and rotating part); (3) Slightly shaking parts (almost no movement in consecutive frames, only a small amount of shaking parts). In short, body parts involved in human movement often have a faster rate of change, while areas with a smaller rate of change mainly include body parts that are not involved in sports. In order to further amplify the variance difference between the low-speed group and the high-speed group, all the velocities obtained in Eq.(14) can be projected as following:

$$Z_t = a \cdot (1 - \frac{1}{1 + e^{-\lambda \cdot v_t}}) \tag{14}$$

where λ is a hyperparameter. In order to keep the predicted value in the range of [0,1], we set a to 2. Here we set the value of λ to 5, and its curve according to the projection function Eq.(14) is shown in Fig. 4. It can be seen from the Fig. 4 that the value range of the velocity obtained after the projection of Eq.(13) can be roughly divided into three time periods. Among them, the slope of the curve in the red interval is steeper, and the corresponding variance of the projection value is larger; the slope of the curve in the blue interval is slightly slower, and the corresponding variance is relatively small; the curve in the green interval has almost no slope, it is approximately a straight line, and the corresponding variance is close to zero. In this way, we can correspond Severely changing parts (red interval), Slowly changing parts (blue interval), or Slightly shaking parts (green interval) of the human body during movement to the different parts in the above-mentioned curve one-to-one. Obviously, the projection method of Eq.(14) expands the difference between these three behaviors, thereby increasing the effective perception of spatial motion changes.

Based on this idea, we calculated the changes in motion characteristics between each block, and obtained a set of spatial position change rates defined as Eq.(15):

$$z = \{z_1, z_2, \ldots, z_T\} \tag{15}$$

where T denotes the number of blocks mentioned before. This is a time series composed of Riemannian manifold gradients for an individual, and then the average, variance, maximum and minimum values of the set are used to describe the spatial change state of the motion during this time period. The feature vector is defined as:

$$f_{mv} = [E[z], \sigma[z], max[z], min[z]] \in \mathbb{R}^4 \tag{16}$$

where $\sigma[z]$ denotes the variance of z. In fact, f_{mv} characterize the spatial change of the moving target body part of the entire video sequence, which is called MVF. Finally, the MF in each block are accumulated separately, and concatenating with MVF to construct the final feature representation.

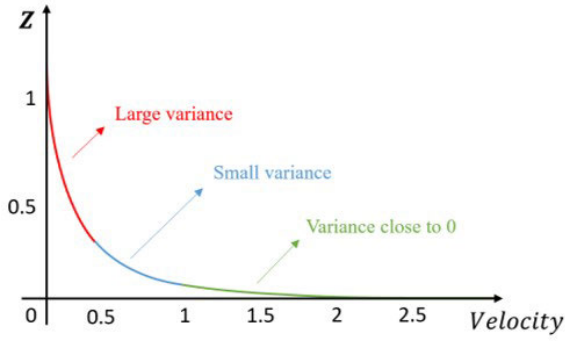

FIGURE 4. The projection curve according to Eq.(14).

D. ACTION RECOGNITION STRATEGY

In this subsection, we describe the recognition strategy in detail. In the previous work, the MF f_m that reflect the trend of motion and the MVF f_{mv} that reflect the spatial position change of moving parts have been calculated. Since both features only consider more one-sided information, it is obviously not appropriate to apply them alone as the input for classification.

In this consideration, three classical fusion strategies is applied to fuse MF and MVF. Section IV-C shows the performance achieved by different fusion strategies. Based on the experiment, we chose the "concatenate" for feature fusion. First, we accumulate the physical characteristics $f_m{}^j = \{div^j, curl^j\}$ of each block to obtain the MF $F_m = \{Div, Curl\} \in \mathbb{R}^{W\times H\times 2}$ of entire video sequence. After that, reshaping MF Div and $Curl$ to feature vectors $v_{Div}, v_{Curl} \in \mathbb{R}^{(W\times H)}$ (One-dimensional). And then, concatenating v_{Div}, v_{Curl} and MVF $f_{mv} \in \mathbb{R}^4$ to obtain the final fusion feature $f \in \mathbb{R}^{(W\times H)\times 2+4}$ (One-dimensional). Finally, two fully connected layers (FC) in series are used as a classifier to classify the fusion features to complete the final recognition target, where the first FC apply **LeakyReLU** as the activation function to prevent neuron death, and the second FC apply **Sigmoid** as the activation function to calculate the classification score.

IV. EXPERIMENTS AND ANALYSIS

In this section, five popular common datasets (KTH, UCF sports, UCF101, HMDB51, JHMDB) are used to evaluate the effectiveness of the proposed framework. Besides, the mean Average Precision (mAP) is applied to verify the performance.

A. DATASETS

KTH [28]. The dataset contains a total of 599 sets of data, including 6 actions. Each action is completed by 25 characters in 4 different scenes. And each video can be divided into 4 subsequence. The movement of KTH dataset is relatively standard, and the number of fixed shots is also relatively abundant for the current model training.

TABLE 1. Public human action recognition datasets information.

Datasets	Categories	resolution	Train	Test	Val.
KTH [28]	6	320 × 240	359	120	120
UCF Sports [29]	10	720 × 480	90	30	30
UCF 101 [30]	101	320 × 240	7992	2664	2664
JHMDB [32]	21	320 × 240	558	185	185
HMDB51 [31]	51	320 × 240	4109	1370	1370

UCF sports [29]. The data set consists of 150 sequences with a resolution of 720 × 480. The series represents a natural pool of action features in a wide range of scenarios and perspectives. Since it was proposed, the dataset has been widely used in the fields of action recognition, action location and saliency detection.

UCF101 [30]. The dataset is collected from YouTube and other video libraries, including 101 action categories, with more than 100 video samples in each category. The data set contains a total of 13320 videos, each action category is divided into 25 groups, each group contains 4-7 action videos from the same perspective.

HMDB51 [31]. HMDB51 contains 51 types of actions, a total of 6849 videos, each action contains at least 51 videos, with a resolution 320 × 240. From YouTube, Google Video, etc., with more complex background and shot switching.

JHMDB [32]. JHMDB is a secondary annotation of the HMDB data set, namely joint-annotated HMDB. The HMDB data set has 51 categories and more than 5100 videos. JHMDB only marked a part of HMDB, that is, only included 21 categories, and deleted some samples that are not obvious to people in these 21 categories. In these 21 categories, each category has $36 - 55$ samples, each sample includes the start and end time of the behavior, and each sample includes 14-40 frames.

The detailed information of the dataset and the division of the training set are shown in Table 1.

B. EXPERIMENTAL SETUP

In experiments, to extract the dense optical flow, we use identical settings as [33]. For AP, the size of the approximate vector b is set as 1024×1 and the size of a_k is set as $1024 \times k$, k is the number of action categories, which change as the datasets changes. The dimension of MF vector is defined as $32 \times 64 \times 2$. The number of epochs is set as 20. The number of neurons in the two fully connected layers is set to $(32 \times 64 \times 2) + 4$ and k respectively, k is the number of action categories. For the unit n-sphere projection algorithm, the hypersphere dimension n is set to be the same as MF vector, that is, $32\times64\times2$. The hyperparameter is set as 5 and a is 2. The above parameters are fixed and will not change with the data set. For the liner SVM which applied in Section IV-C, we set the regularization loss trade-off parameter C of the linear SVM to 100. All the algorithms were implemented in Python 3.7 and performed on a computer with Intel(R) Core(TM) 2 Duo 3.0 GHz i7 CPU 16G RAM, RTX 2080Ti GPU and windows 64bit operation system.

C. ABLATION STUDY

To verify the performance of our proposed framework, we conducted ablation studies with five state-of-the-art methods:

HDL model [34]. This paper proposed a human action recognition method based on hybrid deep learning model. Gaussian mixture model (GMM) and Kalman filter (KF) are combined to detect and extract moving objects. In addition, according to the gating recurrent neural network, the features of each frame are collected to predict human action.

FF-BFS [35]. This paper implements an action recognition technique based on features fusion and best feature selection. In this paper, a new parallel method is used to extract and fuse shape and texture features, and a new weighted entropy variance method is applied to combine vectors for behaviour classification and recognition.

STCM [36]. This paper proposed spatio-temporal context model for action recognition. STCM counts the context information of video sequence, including temporal context information and spatial context information around the target. Then, the dynamic programming method is used to collect evidence on a small candidate set to effectively detect the temporal and spatial location of action.

SOD-SSD [37]. This paper proposed a strong object detector based on single shot multi-box detector framework for action recognition. SOD-SSD introduces an anchor thinning branch at the end of the backbone to refine the input anchor, and adds a batch normalization layer before connecting the intermediate feature mapping at the frame level to obtain more accurate feature representation.

iGDA [38]. This paper proposed a framework for classifying motion sequences based on Grassmann discriminant analysis (GDA). iGDA projects the subspace like subspace onto the generalized difference molecular space before mapping the subspace like subspace to the Grassmann manifold, so as to remove the overlap of Subspaces in the vector space.

Girdhar and Ramanan [23]. This method first proposed the concept of introducing the attention mechanism into the pooling layer. Since the method we proposed is mainly a large-scale improvement on this method, we will choose this method as the baseline method for experiments.

We reproduced the above methods as much as possible for ablation study, and verified the effectiveness of our proposed framework based on the performance of these methods. It is worth noting that in order to demonstrate the generalization ability of our model on different datasets, we mainly conduct experiments on three types of datasets in this subsection, including **KTH** (simple behavior under simple background), **UCF Sports** (complex behavior under simple background) and **UCF 101** (complex behavior under complex background). As for the HMDB51 and JHMDB datasets, because they only have more shot switching and jitters than UCF 101, we only give examples in the above three datasets.

TABLE 2. The performance comparison of MF and other methods.

Method	KTH	UCF Sports	UCF 101
HDL	96.30%	89.01%	89.30%
FF-BFS	100%	99.18%	-
STCM	-	59.57%	49.39%
SOD-SSD	-	96.6%	56.6%
iGDA	-	-	84.67%
Girdhar et al. [23]	91.23%	87.95%	79.82%
MF(ours)	92.57%	92.74%	81.27%

1) EVALUATION OF MOTION FEATURE EXTRACTION

In our proposed framework, the extraction and construction of MF plays an important role. To analyse its contribution, we visualized the results of optical flow extraction based on ROI, and built the characteristic chart of divergence and curl of optical flow field, as shown in Fig. 5.

From the result shown in Fig. 5, it is not difficult to see that the dense optical flow field shown in Fig. 5(a) extracted directly from the video sequence sample soften contains a certain degree of background interference. After attention pooling processing, We can obtain the target motion area, so that, the Fig. 5(c), Fig. 5(d), Fig. 5(e) and Fig. 5(f) respectively show the thermal map of divergence and curl accumulation for the flow field after attention pooling processing, where Fig. 5(c) is the 3D representation of the divergence feature in Fig. 5(d), Fig. 5(e) is the 3D representation of Fig. 5(e). We divide the image into multiple cells, and build MF by accumulating the physical characteristics of the flow field of 15 consecutive frames in the cell. As can be seen from Fig. 5(d), the MF that we are cheap enough to effectively reflect the changes in the movement, including the direction of movement, the magnitude of change and other information. It's not hard to see that the divergence feature can better describe the boundary information of the image, because the boundary of the moving target has a larger flow change and can be recorded by the divergence feature; and the curl feature can better describe the rotation information of the image, as shown in the Fig. 5(d), the wrist of moving target has a larger amount of rotation, which also means that there will be a larger rotation value here.

In addition, we also performed a quantitative analysis of the constructed MF. We sent the MF to linear SVM classification after being encoded by Fisher vector, and compared the results with other methods, as shown in Table 2.

From the result shown in Table 2, we notice that our MF have high discrimination, but still cannot reach the recognition rate of some mainstream methods. We believe that this is because single MF is not comprehensive enough and it cannot consider the correlation between features, let alone quantitatively calculate the spatial position relationship of the moving parts. This also lead to the problem of low recognition rate.

2) EVALUATION ON SIZE OF MF

Regarding the number of sub-blocks obtained by the division, that is, the scale of the MF, it is not difficult for us to draw the

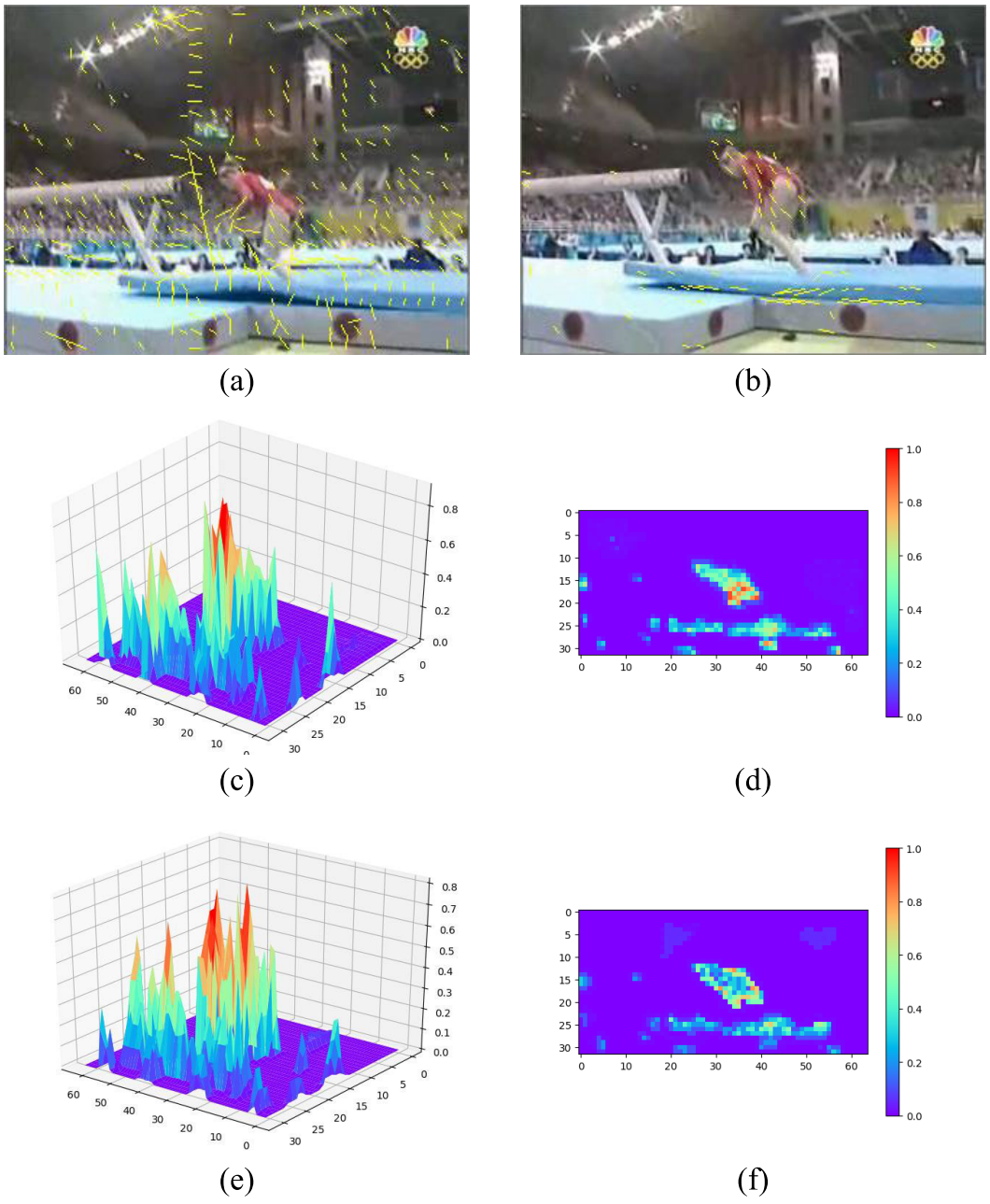

FIGURE 5. Visualization of pre-processing of optical flow field. (a) The optical flow field collected on the original video. (b) The ROI extracted by Attention Pooling from (a). (c) and (d) is the divergence characteristic of continuous frame. (e) and (f) is the curl characteristic of continuous frame.

conclusion: as the number of sub-blocks is larger, the approximation of the boundary of the motion area is more accurate, and the recognition result is higher, but the more the number of full connection layer parameters will be needed. Therefore, we need to select an appropriate number of sub-blocks, and achieve better recognition results as much as possible under the premise of fewer parameters. In order to find such an appropriate division method, we selected the eight scales to divide MF, the recognition performance as show in Fig. 6.

From the results shown in Fig. 6, we noticed that the with the increase of sub blocks' number, the growth rate of recognition performance gradually decreases. This also means that we do not need to put too much computational burden to get the highest recognition effect. Considering that every time the number of sub blocks is increased, the subsequent parameters will show an exponential growth trend. Therefore, the 32×64 scale is finally adopted as the division method of blocks. Besides, We list the detailed data in Figure 6 in Table 3 to make this result more prominent.

From Table 3, we can also see that the recognition results obtained by the three division methods of 32×64, 64×64 and 64×128 do not have much difference, but obviously the latter requires much larger parameter scale than the former. This also more clearly proves that the division method we

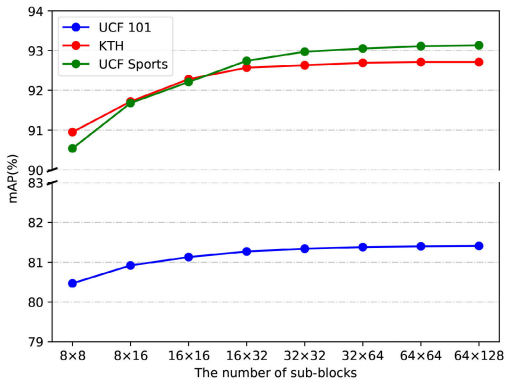

FIGURE 6. The recognition performance of different division methods on KTH, UCF Sports and UCF 101 datasets.

TABLE 3. Recognition performance of different division method.

Division methods	KTH	UCF Sports	UCF 101
8×8	90.95%	90.54%	80.47%
8×16	91.72%	91.68%	80.92%
16×16	92.28%	92.21%	81.13%
16×32	92.57%	92.74%	81.27%
32×32	92.63%	92.97%	81.34%
32×64	92.69%	93.05%	81.38%
64×64	92.71%	93.11%	81.40%
64×128	92.71%	93.13%	81.41%

chose is the most appropriate, that is, the sub-blocks obtained by the 32×64 division can already approach the motion boundary very well, and construct the MF that can reflect motion trend.

3) EVALUATION OF FUSION STRATEGIES

In Section III, we design two different features, i.e. MF and MVF, which need to be fused to make them as the input of classifier. In this subsection, we design 7 fusion strategies in total, where **Feature Sum** (we denoted by $+$) is to map features to a unified scale and then add them together. **Scores Average** (we denoted by $\boxplus$) is to classify two features by using two classifiers respectively, and then weighted sum the classification scores to obtain the final classification score. **Concatenation** (we denoted by $\oplus$) is to splice two features in one dimension, and then send the spliced features into the classifier to calculate their scores. The recognition results are shown in the Table 4.

From the results shown in Table 4, we can see that the effect of adopting **Scores Average** ($\boxplus$) as fusion strategy is not ideal (As shown in the second and seventh rows in the Table 4), we believe that this is because MF and MVF cannot fully characterize the information in video, and it is easier to produce contradictory judgments and cause false interference to the final recognition performance. Besides, since the dimensions between MF and MVF do not match, or even far from each other, it leads to the loss of more information when two features are projected on the same dimension. Therefore, the **Feature Sum** ($+$) fusion strategy

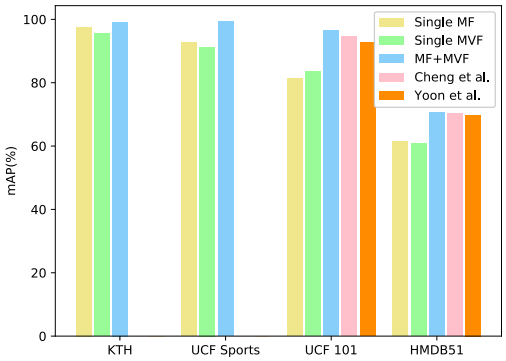

FIGURE 7. The recognition performance of different division methods on KTH and UCF Sports datasets.

has not been able to achieve superior performance (As shown in the third row of the Table 4). Besides, although good results have been achieved by the strategy of $(v_{Div} \oplus MVF) + (v_{Curl} \oplus MVF)$ (As shown in the fourth row in the Table 4), since its calculation process is more complicated than the method in row 6, we did not choose this method. Under the trade-off, we finally choose the full **Concatenation** fusion strategy $((v_{Div} \oplus v_{Curl}) \oplus MVF)$ due to its superior performance on most datasets, especially on complex datasets. We think that this is also because the direct **Concatenation** complements the information in MF and MVF and retains all the information in feature fusion.

4) EVALUATION OF UNIT n-SPHERE PROJECTION METHOD

We have introduced the Riemannian manifold learning method in the previous work, but whether this method can bring about performance improvement remains to be discussed. In order to verify its effectiveness, we conducted experiments, mainly on Four datasets (KTH, UCF Sports, UCF 101, HMDB51). Three types of features are used: single MF feature vectors, single MVF feature vectors and MF $+$ MVF fusion vectors. Besides, we also list two state-of-the-art spatio-temporal fusion feature methods, Cheng *et al.* [39] and Yoon *et al.* [40]. The experimental results are shown in Fig. 7.

It can be seen from the results shown in Fig. 7 that the recognition performance of the fusion feature is significantly higher than the other two separate features, and this difference is more prominent on a complex data set. We believe that this also supports the highly one-sided view of individual manual features. Neither MF nor MVF can fully characterize all the information contained in the action. Not only that, the individual motion performance in the HMDB51 and JHMDB datasets is more complex, which leads to lose more information single feature by single features. In addition, we also compared with the most advanced spatio-temporal feature fusion methods [39], [40]. The superior performance also verified that our MF $+$ MVF is more discriminative than the temporal and spatial features extracted from the depth network.

TABLE 4. The comparison of recognition performance of different module combinations.

Methods	KTH	UCF Sports	UCF 101	HMDB51
$(v_{Div} + v_{Curl}) \oplus MVF$	97.86%	96.26%	92.72%	67.73%
$(v_{Div} + v_{Curl}) \boxplus MVF$	95.49%	93.78%	89.14%	65.29%
$(v_{Div} + v_{Curl}) + MVF$	96.42%	95.39%	94.33%	66.74%
$(v_{Div} \oplus MVF) + (v_{Curl} \oplus MVF)$	**99.47%**	96.83%	96.27%	69.35%
$(v_{Div} + MVF) + (v_{Curl} + MVF)$	96.87%	96.87%	94.15%	67.24%
$(v_{Div} \oplus v_{Curl}) \oplus MVF$	99.31%	**97.83%**	**97.34%**	**71.28%**
$(v_{Div} \oplus v_{Curl}) \boxplus MVF$	96.61%	94.37%	87.35%	66.89%

TABLE 5. The computational complexity comparison with different methods.

Method	KTH	UCF Sports	UCF 101	HMDB51	JHMDB
HDL [34]	32.13M	44.29M	58.41M	53.43M	36.27M
FF-BFS [35]	24.52M	43.61M	50.39M	47.16M	28.40M
STCM [36]	38.19M	55.36M	68.41M	65.12M	41.72M
SOD-SSD [37]	20.84M	36.73M	45.24M	41.98M	26.55M
iGDA [38]	3.81M	5.87M	6.94M	6.58M	5.24M
TEA [41]	3.81M	5.87M	6.94M	6.58M	5.24M
Girdhar et al. [23]	1.76M	3.05M	3.87M	3.63M	2.14M
MF+MVF (ours)	1.97M	3.18M	3.92M	3.76M	2.25M

D. EVALUATION ON COMPUTATIONAL COMPLEXITY

In this subsection, we compared the computational complexity on five classical datasets. We trained the proposed framework on NVIDIA TitanX GPU to evaluate the computational complexity. The experimental results are shown in Table 5.

From the results shown in Table 5, we can see that our proposed recognition framework based on MF and manifold learning is significantly different from other existing methods in terms of parameter scale. Especially compared to the state-of-the-art method [41], the overall complexity of our method is much smaller than the result obtained by its structure that highly stacked network depth. It is worth noting that our method is still slightly larger in complexity than [23]. This is also because we have expanded the method on the basis of [23] and added a small number of parameters. Nevertheless, we have achieved a recognition effect much higher than [23], and only increased the amount of calculation that can be ignored.

For the results in Table 5. We conclude that proposed framework achieve faster recognition speed, which is due to the following three points: (1) The AP is adopted to reduce the interference caused by the complex background, and only using the extracted ROI as the subsequent input also makes the dimension of the input sample much smaller than other methods; (2) In the field of fluid mechanics, divergence, curl and gradient are effective representations of flow field changes. The representation of the entire flow field through these three features further reduces the dimension of the feature; (3) The n-unit sphere is applied to measure the rate of change between features. We finally reduced the dimensionality of MVF by taking the mean, maximum, minimum and variance of the change sequence of the entire video sequence.

TABLE 6. The comparison of recognition performance of different module combinations.

Method	KTH	UCF Sports	UCF 101	HMDB51	JHMDB
HDL [34]	96.30%	89.01%	89.30%	61.53%	-
FF-BFS [35]	100%	99.18%	-	-	84.19%
STCM [36]	-	59.57%	49.39%	66.37%	86.77%
SOD-SSD [37]	-	96.6%	56.6%	57.19%	82.59%
iGDA [38]	-	-	84.67%	66.84%	-
Gong et al. [19]	-	94.13%	90.75%	-	-
Abdelkader et al. [18]	96.42%	-	92.93%	58.37%	-
Kiruba et al. [42]	97.6%	-	91.3%	-	86.39%
Zong et al. [43]	-	-	93.5%	69.42%	86.67%
Dinesh et al. [44]	95.35%	-	96.36%	67.93%	-
Mandal et al. [45]	-	-	95.7%	70.4%	-
Nagrani et al. [46]	-	-	-	71.3%	-
Feichtenhofer et al. [47]	-	-	96.82%	70.65%	-
Shou et al. [48]	-	-	95.97%	69.83%	-
MF+MVF (ours)	99.31%	97.83%	97.34%	71.28%	87.76%

E. PERFORMANCE COMPARISON WITH THE STATE-OF-THE-ART METHODS

In this subsection, we compared the proposed framework with other methods on five classic datasets (KTH, UCF Sports, UCF 101, HMDB51, JHMDB), and quantitative analysis verified the performance of our proposed framework. The comparison results shown in Table 6.

From the result shown in Table 6, we compare the performance of the proposed framework with the other eight most advanced methods, and it can be seen that our method still achieves excellent performance improvement in the face of the more complex datasets such as UCF101, HMDB51 and JHMDB. We think that this is because our designed MF and MVF can effectively represent the motion state, in which the divergence of MF reflects the local motion size distribution of the moving target, curl reflects the local motion direction distribution of the moving target, and MVF reflects the spatial position change measurement of the moving parts. Two fully connected layers are applied to train the above three features, effectively learning the state of motion. However, it is worth noting that our framework is limited by two-dimensional spatial modeling defects, that is, it is unable to extract similar features from the same action from different perspectives. Therefore, we will conduct a more in-depth study on this issue.

V. CONCLUSION

In this paper, we proposed a novel action recognition framework based on optical and manifold learning, our framework combines the physical characteristics of the flow field with the knowledge of manifold learning, and models the movement of the target from multiple angles. Compared with most

mainstream action recognition algorithms at this stage, our proposed framework has the advantages of simple network structure, strong versatility and high scalability. However, as mentioned earlier, our model is also limited by the shortcomings of 2-dimensional modelling, that is, it is unable to extract similar motion features from the same behaviour in different perspectives in the three-dimensional space. Therefore, we will conduct further research on the construction of motion features in three-dimensional space.

REFERENCES

[1] J. Wang, Z. Xu, and Y. Liu, "Texture-based segmentation for extracting image shape features," in *Proc. 19th Int. Conf. Autom. Comput.*, 2013, pp. 1–6.

[2] H. Sui, Z. Song, D. Gao, and L. Hua, "Automatic image registration based on shape features and multi-scale image segmentation," in *Proc. 2nd Int. Conf. Multimedia Image Process. (ICMIP)*, Mar. 2017, pp. 118–122.

[3] H. Akbari, H. M. Kalkhoran, and E. Fatemizadeh, "A robust FCM algorithm for image segmentation based on spatial information and total variation," in *Proc. 9th Iranian Conf. Mach. Vis. Image Process. (MVIP)*, Nov. 2015, pp. 180–184.

[4] Y. Li, J. Zhang, P. Gao, L. Jiang, and M. Chen, "Grab cut image segmentation based on image region," in *Proc. IEEE 3rd Int. Conf. Image, Vis. Comput. (ICIVC)*, Jun. 2018, pp. 311–315.

[5] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun, "Object detection networks on convolutional feature maps," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1476–1481, Jul. 2017.

[6] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[7] S.-C. Hsu, Y.-W. Wang, and C.-L. Huang, "Human object identification for human-robot interaction by using fast R-CNN," in *Proc. 2nd IEEE Int. Conf. Robotic Comput. (IRC)*, Jan. 2018, pp. 201–204.

[8] K. Wang, Y. Dong, H. Bai, Y. Zhao, and K. Hu, "Use fast R-CNN and cascade structure for face detection," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2016, pp. 1–4.

[9] J. Zhang and H. Hu, "Domain learning joint with semantic adaptation for human action recognition," *Pattern Recognit.*, vol. 90, pp. 196–209, Jun. 2019.

[10] H. Wang, C. Schmid, and C. L. Liu, "Action recognition by dense trajectories," in *Proc. IEEE/CVPR Comput. Soc. Conf. Comput. Vis. Pattern Recognit. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 3169–3176.

[11] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3551–3558.

[12] B. Jiang, X. Yin, and H. Song, "Single-stream long-term optical flow convolution network for action recognition of lameness dairy cow," *Comput. Electron. Agricult.*, vol. 175, Aug. 2020, Art. no. 105536.

[13] L. Wang, H. Xiao, S. Luo, J. Zhang, and X. Liu, "A weighted feature extraction method based on temporal accumulation of optical flow for micro-expression recognition," *Signal Process., Image Commun.*, vol. 78, pp. 246–253, Oct. 2019.

[14] Y. Li, Q. Miao, K. Tian, Y. Fan, X. Xu, Z. Ma, and J. Song, "Large-scale gesture recognition with a fusion of RGB-D data based on optical flow and the C3D model," *Pattern Recognit. Lett.*, vol. 119, pp. 187–194, Mar. 2019.

[15] J. Xu, R. Song, H. Wei, J. Guo, Y. Zhou, and X. Huang, "A fast human action recognition network based on spatio-temporal features," *Neurocomputing*, vol. 441, pp. 350–358, Jun. 2021.

[16] Y. Yi, A. Li, and X. Zhou, "Human action recognition based on action relevance weighted encoding," *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115640.

[17] S. Tanberk, Z. H. Kilimci, D. B. Tükel, M. Uysal, and S. Akyokus, "A hybrid deep model using deep learning and dense optical flow approaches for human activity recognition," *IEEE Access*, vol. 8, pp. 19799–19809, 2020.

[18] M. F. Abdelkader, W. Abd-Almageed, A. Srivastava, and R. Chellappa, "Silhouette-based gesture and action recognition via modeling trajectories on Riemannian shape manifolds," *Comput. Vis. Image Understand.*, vol. 115, no. 3, pp. 439–455, Mar. 2011.

[19] D. Gong and G. Medioni, "Dynamic manifold warping for view invariant action recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 571–578.

[20] J. Gall, A. Yao, and L. Van Gool, "2D action recognition serves 3D human pose estimation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2010, pp. 425–438.

[21] R. Slama, H. Wannous, M. Daoudi, and A. Srivastava, "Accurate 3D action recognition using learning on the Grassmann manifold," *Pattern Recognit.*, vol. 48, no. 2, pp. 556–567, Feb. 2015.

[22] F. Martínez Carrillo, M. Gouiffès, G. G. Villamizar, and A. Manzanera, "A compact and recursive Riemannian motion descriptor for untrimmed activity recognition," *J. Real-Time Image Process.*, vol. 1, pp. 1–14, Jan. 2021.

[23] R. Girdhar and D. Ramanan, "Attentional pooling for action recognition," 2017, *arXiv:1711.01467*. [Online]. Available: http://arxiv.org/abs/1711.01467

[24] H. Sang, Z. Zhao, and D. He, "Two-level attention model based video action recognition network," *IEEE Access*, vol. 7, pp. 118388–118401, 2019.

[25] X.-H. Chen and J.-H. Lai, "Detecting abnormal crowd behaviors based on the div-curl characteristics of flow fields," *Pattern Recognit.*, vol. 88, pp. 342–355, Apr. 2019.

[26] F. Wang, G. Wang, Y. Huang, and H. Chu, "SAST: Learning semantic action-aware spatial-temporal features for efficient action recognition," *IEEE Access*, vol. 7, pp. 164876–164886, 2019.

[27] Y. Yun and I. Y.-H. Gu, "Human fall detection in videos by fusing statistical features of shape and motion dynamics on Riemannian manifolds," *Neurocomputing*, vol. 207, pp. 726–734, Sep. 2016.

[28] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, 2004, pp. 32–36.

[29] M. Rodriguez, "Spatio-temporal maximum average correlation height templates in action recognition and video summarization," *Electron. Theses Dissertations*, vol. 1, pp. 2004–2019, May 2010.

[30] K. Soomro, A. Roshan Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*. [Online]. Available: http://arxiv.org/abs/1212.0402

[31] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2556–2563.

[32] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black, "Towards understanding action recognition," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 3192–3199.

[33] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Proc. Scandin. Conf. Image Anal.* Cham, Switzerland: Springer, 2003, pp. 363–370.

[34] N. Jaouedi, N. Boujnah, and M. S. Bouhlel, "A new hybrid deep learning model for human action recognition," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 32, no. 4, pp. 447–453, May 2020.

[35] F. Afza, M. A. Khan, M. Sharif, S. Kadry, G. Manogaran, T. Saba, I. Ashraf, and R. Damaševičius, "A framework of human action recognition using length control features fusion and weighted entropy-variances based feature selection," *Image Vis. Comput.*, vol. 106, Feb. 2021, Art. no. 104090.

[36] W. Xu, Z. Miao, J. Yu, and Q. Ji, "Action recognition and localization with spatial and temporal contexts," *Neurocomputing*, vol. 333, pp. 351–363, Mar. 2019.

[37] Y. Zhang, M. Ding, Y. Bai, D. Liu, and B. Ghanem, "Learning a strong detector for action localization in videos," *Pattern Recognit. Lett.*, vol. 128, pp. 407–413, Dec. 2019.

[38] L. S. Souza, B. B. Gatto, J.-H. Xue, and K. Fukui, "Enhanced Grassmann discriminant analysis with randomized time warping for motion recognition," *Pattern Recognit.*, vol. 97, Jan. 2020, Art. no. 107028.

[39] S. Cheng, G. Qin, S. Li, M. Xie, and Z. Ma, "VLAD-SSTA: VLAD with soft spatio-temporal assignment for action recognition," in *Proc. 16th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process.*, Dec. 2019, pp. 217–221.

[40] Y. Yoon, J. Yu, and M. Jeon, "Spatio-temporal feature extraction and distance metric learning for unconstrained action recognition," in *Proc. 16th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Sep. 2019, pp. 1–8.

[41] Y. Li, B. Ji, X. Shi, J. Zhang, B. Kang, and L. Wang, "Tea: Temporal excitation and aggregation for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 909–918.

[42] K. Kiruba, E. D. Shiloah, and R. R. C. Sunil, "Hexagonal volume local binary pattern (H-VLBP) with deep stacked autoencoder for human action recognition," *Cognit. Syst. Res.*, vol. 58, pp. 71–93, Dec. 2019.
[43] M. Zong, R. Wang, X. Chen, Z. Chen, and Y. Gong, "Motion saliency based multi-stream multiplier ResNets for action recognition," *Image Vis. Comput.*, vol. 107, Mar. 2021, Art. no. 104108.
[44] D. K. Vishwakarma and T. Singh, "A visual cognizance based multi-resolution descriptor for human action recognition using key pose," *AEU - Int. J. Electron. Commun.*, vol. 107, pp. 157–169, Jul. 2019.
[45] D. Mandal, S. Narayan, S. K. Dwivedi, V. Gupta, S. Ahmed, F. S. Khan, and L. Shao, "Out-Of-distribution detection for generalized zero-shot action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9985–9993.
[46] A. Nagrani, C. Sun, D. Ross, R. Sukthankar, C. Schmid, and A. Zisserman, "Speech2Action: Cross-modal supervision for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10317–10326.
[47] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 6202–6211.
[48] Z. Shou, X. Lin, Y. Kalantidis, L. Sevilla-Lara, M. Rohrbach, S.-F. Chang, and Z. Yan, "DMC-net: Generating discriminative motion cues for fast compressed video action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1268–1277.

JUN WANG received the Ph.D. degree in control science and engineering from Central South University, in 2018. He is currently an Associate Professor with the Zhongshan Institute, University of Electronic Science and Technology of China. His main research interests include computer vision, pattern recognition, and behavior analysis.

LIMIN XIA received the Ph.D. degree in control science and engineering from Central South University, China, in 2000. He is currently a Professor with Central South University. He has published more than 150 articles in this field. His main research interests include computer vision, pattern recognition, and behavior analysis.

WENTAO MA was born in China, in 1996. He received the B.S. degree in automation from Central South University, Hunan, China, in 2019, where he is currently pursuing the M.S. degree in control science and engineering. His research interests include computer vision, pattern recognition, and behavior analysis.