IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 29, 2018, accepted April 26, 2018, date of current version June 26, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2832290

A Novel SURF Based on a Unified Model of Appearance and Motion-Variation

YANSHAN LI[1,2,3], CONGZHU YANG[1], LI ZHANG[3], RONGJIE XIA[1], LEIDONG FAN[1], AND WEIXIN XIE[1]

[1]ATR National Key Laboratory of Defense Technology, Shenzhen University, Shenzhen 518060, China
[2]Guangdong Key Laboratory of Intelligent Information Processing, College of Information Engineering, Shenzhen University, Shenzhen 518060, China
[3]College of Information Engineering, Shenzhen University, Shenzhen 518060, China

Corresponding author: Yanshan Li (lys@szu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grants 61771319 and 61401286, in part by the Natural Science Foundation of Guangdong Province under Grant 2017A030313343, and in part by the Shenzhen Science and Technology Project under Grants JCYJ20160520173822387 and JCYJ20160307143441261.

ABSTRACT The speeded-up robust features (SURFs) algorithm is the best and most efficient local invariant feature algorithm for application to 2-D images and is widely applied in the fields of 2-D image processing and computer vision. Compared to 2-D images, a video has motion information in addition to its appearance information. Here, to make full use of the video appearance and motion information, we use geometric algebra as the mathematical calculation and analysis framework to obtain the embedded appearance and motion information on a local area of a video. In our proposed model of appearance and motion variation (UMAMV), we developed SURF feature detection and description algorithms operating on the spatio-temporal domain with video appearance and motion information. First of all, a model of appearance and motion variation, which contains video appearance and motion information in the framework of geometric algebra, is proposed. Then, based on this model, we propose a novel detection algorithm, the UMAMV-SURF detector, which mainly contains Hessian matrix construction, Hessian matrix determinant approximation calculation, and non-maximal suppression determination feature points as its key steps. Then, we introduce the UMAMV-SURF description algorithm, which mainly includes determining the dominant orientation of UMAMV-SURF feature points and generating the UMAMV-SURF feature descriptors. Finally, by experimenting with the Weizman and UCF101 datasets, the experimental results show that the proposed UMAMV-SURF algorithm can detect those SURF feature points which can have unique appearance information in the spatial domain and reflect motion change in the temporal domain. Moreover, it offers a higher accuracy than other spatio-temporal interest point algorithms in human behavior recognition of video footage.

INDEX TERMS Spatio-temporal interest point (STIP), SURF, appearance and motion-variation, geometric algebra.

I. INTRODUCTION

Since its origin in 2003 [1], spatio-temporal interest point (STIP) has been exploited as an important video feature in various video processing fields requiring the intelligent analysis, such as human action recognition, video indexing, anomalous traffic detection, and video monitoring [2]–[16]. Most of the current STIP detectors are extended from the detectors for 2D images to three-dimensions (3D) by adding the temporal component [1], [9], [17]. They detect STIPs by searching for the position where the pixel values of video have significant local variations in 3D video cube. The detected STIPs exhibit the largest appearance change in the video and thus can reflect local structures in the spatio-temporal domain. However, these detectors only implicitly capture motion information, leaving a significant performance gap to fill. Since only interest points with sufficient motion will provide the necessary information for video action recognition [42], MoSIFT was proposed to detect distinctive local features by using local appearance and motion. By detecting distinctive appearances, we obtain the candidate points, and spatio-temporal local features are selected if the candidate points contain significant movement. However, it treats spatial and temporal dimensions separately, which splits the correlation of video pixels in the spatio-temporal

domain. Therefore, we aim to synthetically analyze appearance and motion information in the process of STIPs detection, and explicitly analyze the motion information. As we know, the highly successful Speeded-Up Robust Features (SURF) [28] for object recognition detects many interest points in an image and it has a good local invariant property such as rotation invariance, scale invariance and affine invariance. Moreover, the video can be regarded as a 3D structure and it is effective to use geometric methods for video analysis and processing. Inspired by the SURF algorithm which detects abundant interest points for 2D images, we utilize geometric algebra to develop a novel STIP detector and descriptor exploiting appearance and motion information synthetically for videos.

II. RELATED WORK

In 2003, Laptev *et al.* [1] extended the Harris corner detector [18] intended for 2D images to the Harris 3D spatio-temporal corner detector by combining spatial and temporal information to detect STIPs in videos [1]. Dollár *et al.* [19] proposed a cuboid detector wherein a response function was defined using linearly separable filters, for example, 2D Gaussian filters in the spatial domain and Gabor filters in the temporal domain. The final locations of the STIPs were then obtained by adjusting the spatial and temporal scale parameters of the response function. Ke *et al.* [20] proposed the use of a volumetric STIP detector which could produce an abundant number of STIPs, each of which was scale invariant; however, the detector requires a large number of calculations to be undertaken in the course of the detection procedure. As a remedy, Oikonomopoulos *et al.* [21] utilized the entropy information of optical flow to detect STIPs. Laptev *et al.* [22], on the other hand, used a method based on the local motion of events. Experimental results proved that both methods in [21] and [22] realized scale selection. Shabani *et al.* [23] proposed a non-linear scale-space filtering approach to detecting STIPs in real-world videos. Liu *et al.* [24] improved the non-linear anisotropic-diffusion filters proposed by Weickert *et al.* [25] and used them to detect STIPs in scenes with cluttered backgrounds arising from camera movement. The V-FAST detector, proposed by Yu *et al.* [26], is a STIP-detection algorithm based on the image-corner detection. They extended the fast corner detector [27] from the spatial domain to the spatio-temporal domain by using the local appearance of moving objects and their structural information. The Hes-STIP detector proposed by Willems *et al.* [30] is a spatio-temporal extension of the blob detector of a 2D image based on the Hessian matrix [17], [28], [29]. The ST-SIFT detector was proposed by Guo *et al.* [31] as a spatio-temporal extension of the SIFT detector [26], [32] for STIP detection in videos. Following the idea of detecting distinctive local features through local appearance and motion, Chen and Hauptmann [42] proposed an algorithm called MoSIFT, which detects interest points and encodes not only their local appearance but also explicitly models local motion. In 2014, Li *et al.* [33] presented a robust method for human action recognition based on the use of multi-velocity spatio-temporal interest points (MVSTIPs). From the foregoing, the traditional STIP detectors directly extend the local invariant feature detectors for 2D images to the spatio-temporal domain by adding the temporal component, which only implicitly captures motion information. Subsequently, Chen and Hauptmann [42] developed a MoSIFT algorithm that detects spatio-temporal local features which contain distinctive appearance and explicit movement, but it treats spatial and temporal dimensions separately, instead of unifying them for video analysis and processing.

To describe the STIP, Laptev *et al.* [22] proposed the HOG/HOF descriptor, it uses spatial HOG descriptors and HOF descriptors to describe the feature points by calculating the spatial gradient and the optical flow histogram near the feature points, but this method requires a large amount of computational effort because of the need to calculate the optical flow field, and its performance depends on the selected regularization method. The HOG3D descriptor was proposed by Kläser *et al.* [10] to obtain the HOG3D descriptor vector by calculating the 3D HOG and gradient histograms of the video. Dollár *et al.* proposed a cube descriptor which is centered on the spatio-temporal interest points detected by the Dollár detector. They created a cube and the description vector is generated by calculating the values for pixels within the cube. Ullah *et al.* [46] proposed a novel action recognition method by processing the video data using convolutional neural network (CNN) and deep bidirectional LSTM (DB-LSTM) network. Uddin *et al.* [47] introduced a novel feature descriptor, namely, adaptive local motion descriptor (ALMD) by considering motion and appearance. two different kinds of algorithms that using spatio-temporal feature are proposed by Hou *et al.* [48] and Li *et al.* [49] respectively.

As mentioned above, the existing STIP detection and description algorithm cannot adequately reflect the spatio-temporal correlation of video footage, and it is difficult to detect feature points in the spatial domain and reflect motion-variation in the temporal domain at the same time. Therefore, geometric algebra is used to establish an appearance and motion-variation (AMV) model with which to analyze video image appearance information and motion information. Based on the AMV model, a new SURF algorithm based on a unified model of appearance and motion-variation (UMAMV-SURF) for videos is proposed. The main contributions of this paper are summarized as follows:

(a) Based on the AMV model, a new STIP detector for videos that considers appearance and motion information is proposed. This UMAMV-SURF detector is developed under the traditional SURF framework. The new detector inherits the advantages of previous SURF detectors (e.g., strong invariance-resistance and speed of calculation). Moreover, feature points with significant variations in motion can be extracted.

(b) At the same time, UMAMV-SURF descriptor is proposed: this mainly includes the determination of the dominant orientation of UMAMV-SURF feature points

and generation of UMAMV-SURF feature descriptors, which makes UMAMV-SURF features more robust and improves the accuracy of action recognition.

The rest of this paper is organized as follows: Section 3 introduces the proposed general geometric algebra model and UMAMV for videos, in Sections 4 and 5, the UMAMV-SURF detection and description algorithms are introduced individually, the experimental data and results are discussed in Section 6, and Section 7 summarizes the research findings.

III. THE UNIFIED MODEL OF APPEARANCE AND MOTION-VARIATION FOR VIDEO

A. GENERAL GEOMETRIC ALGEBRA MODEL OF SPATIO-TEMPORAL DOMAIN FOR VIDEOS

In [34], the short video sequence is expressed as a video cube containing spatial domain information (x, y) and time-domain information t. Therefore, a n-frame video with spatical size of $M \times N$ can be formed as:

$$F = f(x, y, t) \tag{1}$$

where $f(x, y, t)$ represents a function for the video, (x, y, t) denotes the 3D coordinate, and t $(0 < t \leq n)$ is the coordinate in the temporal domain. Moreover, x and y refer to coordinates in the spatial domain such that $0 < x \leq M$ and $0 < y \leq N$.

In recent years, geometric algebra has proven to be an effective tool for analyzing geometric problems in the information processing field [35]–[41]. In this project, the modified geometric algebra named Clifford algebra is utilized to form the mathematical framework for representing and analyzing videos, which is developed on the basis of Clifford and Grassmann algebras. Geometric algebra allows geometric problems to be solved by converting them into algebraic problems. It also provides a powerful algebra framework for geometric analysis. The representation of video sequences under the geometric algebra framework is illustrated below.

Supposing that $\mathbb{R}^3$ is the 3D Euclidean space consisting of the spatial and temporal domains of the video, and that $\{e_1, e_2, e_3\}$ is an orthonormal basis of this space. Therefore, the algebraic space on $\mathbb{R}^3$ spanned by the orthonormal basis through the geometric product is denoted by $\mathcal{G}_3(\mathbb{R}^3)$. In this study, $\mathcal{G}_3(\mathbb{R}^3)$, or $\mathcal{G}_3$ for the sake of brevity, is considered to be the 3D geometric algebra space of the videos. A group of orthonormal bases for it can be constructed as follows:

$$\begin{aligned} E^3 &:= \{E_i | i = 0, 1, 2, \cdots, 2^3 - 1\} \\ &= \{1, e_1, e_2, e_3, e_1 \wedge e_2, e_2 \wedge e_3, e_1 \wedge e_3, e_1 \wedge e_2 \wedge e_3\} \end{aligned} \tag{2}$$

where $\wedge$ represents the exterior product of the geometric algebra and $e_1 \wedge e_2$, $e_2 \wedge e_3$, and $e_1 \wedge e_3$ are three (independent) exterior products which separately represent the planes expressed by two vectors within $\mathcal{G}_3$ geometrically. Similarly, $e_1 \wedge e_2 \wedge e_3$ refers to an exterior product which corresponds geometrically to the directed geometry obtained by moving the exterior product $e_1 \wedge e_2$ along the vector e_3. In addition, the vector $\{e_1, e_2, e_3\}$ can be considered as the basis vector $\{x, y, t\}$ for a 3D vector subspace of $\mathcal{G}_3$.

The geometric product $e_1e_2e_3$ is written as I. As $e_i^2 = 1$, we have $e_1e_2 = Ie_3$, $e_2e_3 = Ie_1$, and $e_3e_1 = Ie_2$. These products also satisfy

$$(e_1e_2)^2 = (e_2e_3)^2 = (e_3e_1)^2 = -1 \tag{3}$$

If point $p \in \mathcal{G}_3$ and $p = xe_1 + ye_2 + te_3$, then a video can be expressed as follows:

$$F = f(p) \tag{4}$$

where $f(p)$ refers to the pixel values of the pixels in the video F at p.

If $p_1, p_2 \in \mathcal{G}_3$, with $p_1 = x_1e_1 + y_1e_2 + t_1e_3$ and $p_2 = x_2e_1 + y_2e_2 + t_2e_3$, then their geometric product can be expressed as:

$$p_1p_2 = p_1 \cdot p_2 + p_1 \wedge p_2 \tag{5}$$

That is, the geometric product of the two vectors is composed of the sum of the interior product $(p_1 \cdot p_2)$ and the exterior product $(p_1 \wedge p_2)$.

The distance between p_1 and p_2 in $\mathcal{G}_3$, is denoted by Δp and is given by,

$$\Delta p = p_1 - p_2 = (x_1 - x_2)e_1 + (y_1 - y_2)e_2 + (t_1 - t_2)e_3 \tag{6}$$

This represents a vector pointing from p_2 to p_1. Clearly, Δp measures the distance between the two pixels which also reflects the change in motion conditions of the pixels in the video sequence.

B. THE UNIFIED MODEL OF APPEARANCE AND MOTION-VARIATION FOR VIDEO

To take full advantage of the motion information of videos detecting STIPs, a new unified model of appearance and motion-variation is first established for spatio-temporal analysis and processing under the framework of geometrical algebra. Firstly, a geometric algebra vector for describing the appearance and motion information of video is constructed as follows.

Definition 1 (Motion Vector): It is assumed that $p_0, p_1 \in \mathcal{G}_3$, such that $p_0 = x_ie_1 + y_je_2 + t_ke_3$ and $p_1 = x_ie_1 + y_je_2 + (t_k + 1)e_3$. Let S be a set of points in the neighborhood of $(l \times l)$ with the center of p_1 on the plane $t = t_k + 1$. Thus, the motion information v_{p_0} of the pixel at p_0 in $\mathcal{G}_3$ can be defined as follows:

$$v_{p_0} = p_r - p_0 \tag{7}$$

where, $p_r = \arg\min_{p_r \in S} [f(p_r) - f(p_0)]$. Thus, v_{p_0} reflects the motion information of pixels including motion direction and speed.

Definition 2 (Motion-Variation Vector): Assuming that $p_0, p_1, p_2 \in \mathcal{G}_3$, where $p_0 = x_ie_1 + y_je_2 + t_ke_3$, $p_1 = x_ie_1 + y_je_2 + (t_k + 1)e_3$, and $p_2 = x_ie_1 + y_je_2 + (t_k - 1)e_3$.

The motion-variation vector $\mathrm{d}v_{p_0}$ of the pixel at p_0 in $\mathcal{G}_3$ is defined as:

$$\mathrm{d}v_{p_0} = v_{p_1} - v_{p_2} \tag{8}$$

where, v_{p_1} and v_{p_2} refer to the motion vectors at p_1 and p_2, respectively. The vector dv_{p_0} reflects the variation of motion at p_0, including variation of motion direction and speed, and its modulus reflect the overall variation in amplitude. Generally, the larger the variation in motion direction of a pixel at p_0, the larger the modulus of dv_{p_0} is. Similarly, the larger the variation in speed, the larger the modulus dv_{p_0} is, and vice versa.

In this study, a new geometric algebraic vector $f'(p_0)$ is defined to represent the appearance and motion-variation information of the pixel at $p_0 \in \mathcal{G}_3$ as following definition:

Definition 3 (Appearance and Motion-Variation Vector): Assuming that $p_0 \in \mathcal{G}_3$ while $f(p_0)$ and dv_{p_0} refer to the appearance value and motion-variation vector at p_0, respectively. The appearance and motion-variation vector (AMVV for short) $f'(p_0)$ is defined as:

$$f'(p_0) = f(p_0) + \mathrm{d}v_{p_0} \tag{9}$$

The vector $f'(p_0)$ contains both scalar and vector information, and reflects not only appearance information but also variation in motion direction and speed.

Based on the above definitions, the unified model of appearance and motion-variation (UMAMV) for video cube F is constructed as follows:

$$F' = f'(p) \tag{10}$$

where, $f'(p)$ is a function wherein $p \in \mathcal{G}_3$ and is regarded as the independent variable. The value of $f'(p)$ is the AMVV. Consequentially, AMVV consists of appearance information of video, and it also reflects the local motion information including motion direction and speed.

IV. HESSIAN MATRIX IN $\mathcal{G}_3$

To make full use of the appearance and motion information in the video image, a 3D Hessian matrix based on UMAMV is presented: this use appearance information and motion information.

If $p \in \mathcal{G}_3$, then the Hessian matrix with the scale of σ at point p is defined as follows:

$$H(p,\sigma) = \begin{bmatrix} L_{xx}(p,\sigma) & L_{xy}(p,\sigma) & L_{xt}(p,\sigma) \\ L_{xy}(p,\sigma) & L_{yy}(p,\sigma) & L_{yt}(p,\sigma) \\ L_{xt}(p,\sigma) & L_{yt}(p,\sigma) & L_{tt}(p,\sigma) \end{bmatrix} \tag{11}$$

where, $L_{xx}(p,\sigma)$ is the convolution of Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(p,\sigma)$ with $f'(p)$ at point p and $g(p,\sigma)$ is the Gaussian function in $\mathcal{G}_3$, $\sigma = e_1/\sigma_x + e_2/\sigma_y + e_3/\tau$. Then, a Gaussian function in $\mathcal{G}_3$ can be defined by:

$$G(p,\sigma) = \frac{1}{3(2\pi)^{3/2}} (\sigma \wedge \sigma \wedge \sigma) \exp\left(-|p \cdot \sigma|^2\right) \tag{12}$$

where σ represents the scale factor of the Gaussian function in $\mathcal{G}_3$.

Thus, $L_{xx}(p,\sigma)$ can be written as:

$$\begin{aligned} L_{xx}(p,\sigma) &= \frac{\partial^2}{\partial x^2} g(\sigma) \otimes f'(p) \\ &= \frac{\partial^2}{\partial x^2} g(\sigma) \otimes (f(p) + v_p) \\ &= \frac{\partial^2}{\partial x^2} g(\sigma) \otimes f(p) + \frac{\partial^2}{\partial x^2} g(\sigma) \otimes v_p \end{aligned} \tag{13}$$

The first part of the above equation is a Gaussian second order derivative convolution with the original video, the second part is the convolution of the Gaussian second order derivative with the motion vector at point p, which reflects the motion information of point p and its neighborhood in the video. Similarly, other items in the Hessian matrix can be written as:

$$\begin{aligned} L_{yy}(p,\sigma) &= \frac{\partial^2}{\partial y^2} g(\sigma) \otimes f'(p) \\ &= \frac{\partial^2}{\partial y^2} g(\sigma) \otimes f(p) + \frac{\partial^2}{\partial y^2} g(\sigma) \otimes v_p \end{aligned} \tag{14}$$

$$\begin{aligned} L_{xy}(p,\sigma) &= \frac{\partial^2}{\partial x \partial y} g(\sigma) \otimes f'(p) \\ &= \frac{\partial^2}{\partial x \partial y} g(\sigma) \otimes f(p) + \frac{\partial^2}{\partial x \partial y} g(\sigma) \otimes v_p \end{aligned} \tag{15}$$

$$\begin{aligned} L_{tt}(p,\sigma) &= \frac{\partial^2}{\partial t^2} g(\sigma) \otimes f'(p) \\ &= \frac{\partial^2}{\partial t^2} g(\sigma) \otimes f(p) + \frac{\partial^2}{\partial t^2} g(\sigma) \otimes v_p \end{aligned} \tag{16}$$

$$\begin{aligned} L_{yt}(p,\sigma) &= \frac{\partial^2}{\partial y \partial t} g(\sigma) \otimes f'(p) \\ &= \frac{\partial^2}{\partial y \partial t} g(\sigma) \otimes f(p) + \frac{\partial^2}{\partial y \partial t} g(\sigma) \otimes v_p \end{aligned} \tag{17}$$

$$\begin{aligned} L_{xt}(p,\sigma) &= \frac{\partial^2}{\partial x \partial t} g(\sigma) \otimes f'(p) \\ &= \frac{\partial^2}{\partial x \partial t} g(\sigma) \otimes f(p) + \frac{\partial^2}{\partial x \partial t} g(\sigma) \otimes v_p \end{aligned} \tag{18}$$

The determinant of the Hessian matrix in $\mathcal{G}_3$ can be written as:

$$\begin{aligned} \det(H) &= L_{xx}L_{yy}L_{tt} - L_{xx}L_{yt}^2 - L_{xy}^2 L_{tt} + L_{xy}L_{xt}L_{yt} \\ &\quad + L_{xt}L_{xy}L_{yt} - L_{xt}^2 L_{yy} \\ &= L_{xx}(L_{yy}L_{tt} - L_{yt}L_{yt}) + L_{xy}(L_{xt}L_{yt} - L_{xy}L_{tt}) \\ &\quad + L_{xt}(L_{xy}L_{yt} - L_{xt}L_{yy}) \end{aligned} \tag{19}$$

V. SURF DETECTOR AND DESCRIPTOR BASED ON UMAMV

Based on the hessian matrix in $\mathcal{G}_3$, a novel SURF detector and descriptor based on UMAMV were proposed as follow:

A. DETECTOR

The steps of the UMAMV-SURF detector mainly include: 3D Hessian matrix, 3D Hessian matrix determinant approximation, and non-maximal suppression to select the feature points.

Since the calculation of each element in the Hessian matrix in the above equation is complicated, the algorithm uses the integral video [35] and the box filter to simplify the calculation in Eq. (16).

1) INTEGRAL VIDEO ON UMAMV

The concept of integral image is used in the SURF algorithm. By means of the integral image, the filtering of the image and Gaussian second order derivative template is transformed into the addition and subtraction of the integral image. First, we give the definition of the integral video in UMAMV.

Definition 4 (Integral Video on UMAMV): Setting the UMAMV of video F as $F^{'}$, V is the integral video of $F^{'}$, Then the value $v(p_0)$ of V at point p is expressed as:

$$v(p_0) = \sum_{p \in S_0} f^{'}(p) \tag{20}$$

Where, $p_0 \in \mathcal{G}_3$ ($p_0 = x_0e_1 + y_0e_2 + t_0e_3$), S_0 is a cuboid formed from the origin O to the p_0, $p = xe_1 + ye_2 + te_3$ and $p \in S_0$, that is $0 \le x \le x_0, 0 \le y \le y_0, 0 \le t \le t_0, f'(p)$ is the AMVV value at point p. That is, the integral value $V(p_0)$ of point p in UMAMV is obtained by accumulating the AMVV values of all points in the cube from the origin O of the video $F^{'}$ to the point p_0.

2) BOX FILTER

In the traditional SURF algorithm, the box filter is used to approximate the convolution of the 2D image and the Gaussian second order derivative, which is able to simplify the calculation and improve the efficiency. Therefore, the box filter on UMAMV is designed to simplify the convolution of the Gaussian two-order derivative template and integral UMAMV, as shown in Fig. 1.

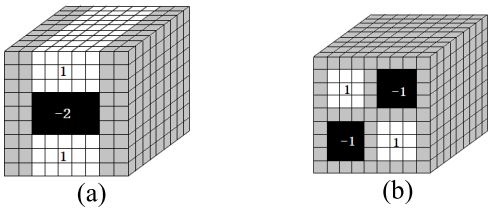

FIGURE 1. Gaussian second order partial derivative approximation filter on UMAMV.

The box filter designed in Fig. 1 is a 3D structure, mostly shown by the figure in the white and black rectangular area. The rectangular box is filled with the same value N ($N \in \{-2, 1, 0, 1\}$), Fig. 1(a) is an approximate box filter for the Gaussian second-order partial derivative in the y-direction (D_{yy}), each cuboid region is sized to $3 \times 5 \times 9$, Fig. 1(b) shows an approximate box filter for the Gaussian second-order partial derivative in the xy-direction (D_{xy}). The cubes are separated by one pixel and the blank is filled with a value of zero.

To get different scale features, we divide the scale space into octaves. Each octave contains a series of response values for filtering the same input UMAMV with a gradually enlarged filter template. Specifically, each octave comprises several layers, the minimum scale change between the two layers is determined by the positive and negative spot length l of the Gaussian second order derivative filter, where l is one third of the box filter template size. Here, the minimum filter template for the box filter is $9 \times 9 \times 9$, then l is 3, the response length of the next layer should be at least two more pixels on the basis of l, to ensure that one pixel is added at each side, and $l = 5$. Then, the size of the next layer of the box filter is $15 \times 15 \times 15$, and so on, thus allowing us to draw the layers of the box filter size as shown in Fig. 2.

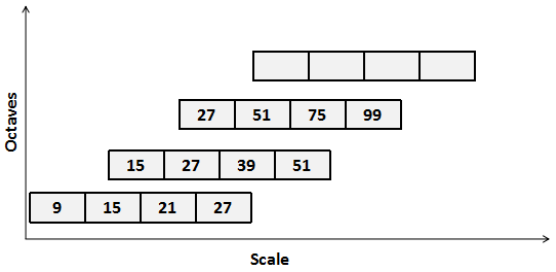

FIGURE 2. The size of three octaves of filters.

In Fig. 2, the horizontal axis represents the size of the box filter size, the vertical axis represents the number of octaves. To cover all possible scales, there is a scale overlap between the octaves and the latter set of filter sizes increases twice compared to that of the previous group (6, 12, . . ., $6 \times 2^{i-1}$, where i is the number of the octave).

3) THE DETERMINANT APPROXIMATION OF THE HESSIAN MATRIX IN $\mathcal{G}_3$

In the approximate calculation of the integral video and the box filter, let $D_{ij}(i, j \in \{x, y, t\})$ be the result of the convolution of the integral video with the corresponding box filter, the response values in each of the rectangular regions are calculated only by the addition and subtraction of the value of each vertex of the cuboid. The rectangular cuboid is shown in Fig. 3.

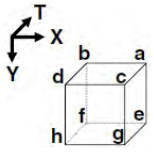

FIGURE 3. Cuboid unit in a box filter.

As shown in Fig. 3, a, b, c, d, e, f, and g are the integral values V in the integral video corresponding to each vertex in the box filter, and the response value of each cuboid and integral video in the box filter is $S_n = \mathrm{N} * (\mathrm{e} - \mathrm{a} - \mathrm{f} - \mathrm{g} + \mathrm{b} + \mathrm{c} + \mathrm{h} - \mathrm{d})$. Therefore, $D_{ij}(i, j \in \{x, y, t\})$ is the sum of the response values of eachcuboid of

the box filter, that is:

$$D_{xx} = \frac{1}{s_{xx}}\sum_{n=1}^{3} S_n, \quad D_{yy} = \frac{1}{s_{yy}}\sum_{n=1}^{3} S_n, \quad D_{tt} = \frac{1}{s_{tt}}\sum_{n=1}^{3} S_n,$$
$$D_{xy} = \frac{1}{s_{xy}}\sum_{n=1}^{4} S_n, \quad D_{xt} = \frac{1}{s_{xt}}\sum_{n=1}^{4} S_n, \quad D_{yt} = \frac{1}{s_{yt}}\sum_{n=1}^{4} S_n$$

Where $S_{ij}(i, j \in \{x, y, t\})$ and A is the volume of the box filter. Based on the above analyses, Eq. (19) can be written as:

$$\begin{aligned}\det(H) &= \frac{L_{xx}L_{yy}L_{tt}}{D_{xx}D_{yy}D_{tt}} \times D_{xx}(D_{yy}D_{tt} - D_{yt}D_{yt} \times \frac{L_{yt}L_{yt}}{D_{yt}D_{yt}} \\ &\quad \times \frac{D_{yy}D_{tt}}{L_{yy}L_{tt}}) + \frac{L_{xy}L_{yt}L_{xt}}{D_{xy}D_{yt}D_{xt}} \times D_{xy}(D_{yt}D_{xt} - D_{tt}D_{xy} \\ &\quad \times \frac{L_{tt}L_{xy}}{D_{tt}D_{xy}} \times \frac{D_{yt}D_{xt}}{L_{yt}L_{xt}}) + \frac{L_{xt}L_{xy}L_{yt}}{D_{xy}D_{yt}D_{xt}} \times D_{xt}(D_{yt}D_{xy} \\ &\quad - D_{yy}D_{xt} \times \frac{L_{yy}L_{xt}}{D_{yy}D_{xt}} \times \frac{D_{yt}D_{xy}}{L_{yt}L_{xy}}) \\ &= C_1(D_{xx}D_{yy}D_{tt} - D_{xx}D_{yt}D_{yt}Y_1) + C_2(D_{xy}D_{yt}D_{xt} \\ &\quad - D_{xy}D_{tt}D_{xy}Y_2) + C_3(D_{xy}D_{yt}D_{xt} - D_{xt}D_{yy}D_{xt}Y_3)\end{aligned} \tag{21}$$

Where

$$C_1 = \frac{L_{xx}L_{yy}L_{tt}}{D_{xx}D_{yy}D_{tt}}, \quad C_2 = \frac{L_{xy}L_{yt}L_{xt}}{D_{xy}D_{yt}D_{xt}}, \quad Y_1 = \frac{L_{yt}L_{yt}}{D_{yt}D_{yt}} \times \frac{D_{yy}D_{tt}}{L_{yy}L_{tt}},$$
$$Y_2 = \frac{L_{tt}L_{xy}}{D_{tt}D_{xy}} \times \frac{D_{yt}D_{xt}}{L_{yt}L_{xt}}, \quad Y_3 = \frac{L_{yy}L_{xt}}{D_{yy}D_{xt}} \times \frac{D_{yt}D_{xy}}{L_{yt}L_{xy}}.$$

Let $A_1 = D_{xx}D_{yy}D_{tt}$, $B_1 = D_{xx}D_{yt}D_{yt}$, $A_2 = D_{xy}D_{yt}D_{xt}$, $B_2 = D_{xy}D_{tt}D_{xy}$, $A_3 = D_{xy}D_{yt}D_{xt}$, $B_3 = D_{xt}D_{yy}D_{xt}$.

This yields

$$\det(H) = C_1(A_1 - B_1Y_1) + C_2(A_2 - B_2Y_2 + A_3 - B_3Y_3) \tag{22}$$

When calculating $\det(H)$, the minimum scale of Gaussian second order derivative filtering is $\sigma = 1.2$, and the minimum template size is $9 \times 9 \times 9$. Thus, in Eq. (21)

$$C_1 = \frac{|L_{xx}(1.2)|_F \left|L_{yy}(1.2)\right|_F |L_{tt}(1.2)|_F}{|D_{xx}(9)|_F \left|D_{yy}(9)\right|_F |D_{tt}(9)|_F} = 0.0018 \approx 0.002,$$

where $|\bullet|_F$ is the Frobenius norm. Notice that, for theoretical correctness, the weighting changes depend on the scale. In practice, we keep this factor constant, as this did not have a significant effect on the results in our experiments. Similarly, $C_2 = 0.038$, $Y_1 = 7.52$, $Y_2 = Y_3 = 0.37$.

From Eq. (9) it can be seen that, the result of $\det(H)$ can be divided into the sum of the convolution values of $f(p)$ and dv_p with the box filter respectively. As a result, one part of the calculation is scalar, which reflects the change of pixel values in video images; another part is the vector and it reflects the movement of the points. Here, the l2-norm of the two parts is used as the response value of the sampling point p.

$$\det(H_p) = \|\det(H)\|_2 \tag{23}$$

After calculating the response value of each sampling point, the feature point was selected by non-maximal suppression.

B. DESCRIPTOR

UMAMV-SURF description consists of two steps: determining the dominant orientation of the feature point, and generating the feature descriptor. *1) Determining the dominant orientation*

Let point $M(\mathrm{p}_0, \sigma_x, \sigma_t)$ be the feature point detected by UMAMV-SURF detector in the geometric algebra space, where p is the coordinates of the feature point in the geometric algebraic space, $p_0 = x_0e_1 + y_0e_2 + t_0e_3$, σ_x and σ_t are the scale of the feature points in the spatial and temporal domains, respectively. A dominant orientation is assigned to each feature point to maintain the rotational invariance of descriptor. We take the Haar wavelet response to the video in the cylindrical region with the radius of $6\sigma_x$, the height of σ_t, and take point p as the center, the Haar wavelet templates used in this paper are shown in Fig. 4.

The dominant orientation is estimated by calculating the sum of all responses within a sliding orientation window W using the feature point as the center: the sliding window W measures $\pi/3$, and changes in 0.2 radian increments. The horizontal and vertical responses F_x, F_y within the window are summed to get a feature vector (m_w, θ_w):

$$m_w = \sum_w F_x + \sum_w F_y \tag{24}$$

$$\theta_w = \arctan(\sum_w F_x / \sum_w F_y) \tag{25}$$

Where the dominant orientation is the maximum Haar response to the cumulative value of the corresponding orientation, thus, the longest vector corresponding to the orientation is:

$$\theta = \theta_w | \max\{m_w\} \tag{26}$$

When there is another peak corresponding to 80% of the main peak energy, this orientation is considered to be the auxiliary orientation of the feature. One feature point may be specified with multiple orientations, including a dominant orientation and multiple auxiliary orientations, which can enhance the robustness of the feature. If the feature point exists in two orientations, we copy it into two feature points. One of the dominant orientations is the orientation corresponding to the maximum response value, and the other dominant orientations is the orientation corresponding to the second response value.

1) GENERATING THE FEATURE DESCRIPTOR

After determining the dominant orientation of each feature point, we then calculate the Haar wavelet response of each sub-region and generate the descriptor by its statistical response value.

The SURF descriptor based on UMAMV is a vector representation of the neighborhood of the feature point and the Haar wavelet response statistics: the feature descriptor is related to the scale of the feature point, so the Haar wavelet response should be calculated on the integral image corresponding to the scale of the feature point. Here, the selected neighborhood size G′ is $20\sigma_x \times 20\sigma_x \times 3\sigma_t$.

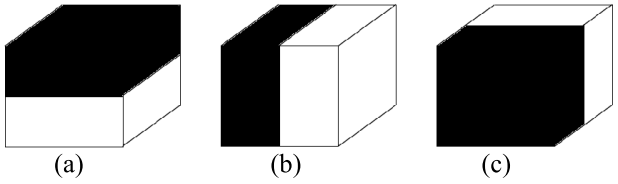

FIGURE 4. (a), (b), and (c) are t, x, y Haar wavelet templates, respectively (the black part of the value is −1, the other part has value 1).

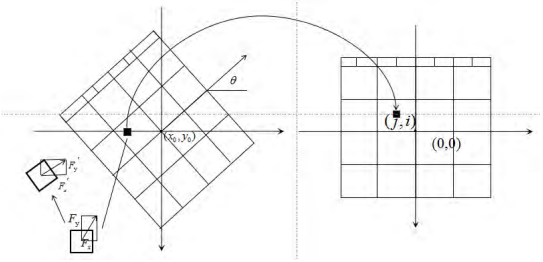

FIGURE 5. The image before, and after, rotation: as F_t is unchanged, drawing a two-dimensional picture allows us to represent it.

The neighborhood of the feature point is divided into $4 \times 4 \times 3$ sub-regions, and each sub-area contains $5\sigma_x \times 5\sigma_x \times \sigma_t$ pixels. The response values are calculated for each sub-region by using a Haar template measuring $2\sigma_x \times 2\sigma_x \times \sigma_t$, then getting the response values in the x-, y-, and t-directions, respectively. Then, the response values in each sub-region are calculated to obtain the values of $\sum F_x$, $\sum |F_x|$, $\sum F_y$, $\sum |F_y|$, $\sum F_t$, $\sum |F_t|$. To be invariant to image rotation, we rotate the original video to its dominant orientation; Nevertheless, we use the Haar wavelet template directly on the integral video to get the response value to avoid the complex calculation caused by the rotation and the full use of the integrated video, which was generated during feature detection.. Then, the response value F_x, F_y, F_t are rotated according to the dominant orientation, resulting in a rotated F'_x, $F'_y F'_t$, that is,

$$F'_x = -F_x \times \sin(\theta) + F_y \times \cos(\theta) \tag{27}$$

$$F'_y = F_x \times \cos(\theta) + F_y \times \sin(\theta) \tag{28}$$

$$F'_t = F_t \tag{29}$$

This transform is shown in Fig. 5.

Thus the response vector V for each sub-block is:

$$V = [\sum F'_x, \sum |F'_x|, \sum F'_y, \sum \left|F'_y\right|, \sum F'_t, \sum |F'_t|] \tag{30}$$

Since there are $4 \times 4 \times 3$ sub-blocks, and each sub-block produces six gradient statistics, the feature descriptors proposed in this paper have a total of $4 \times 4 \times 3 \times 6 = 288$ dimensional feature vectors. Finally, to guarantee the contrast invariance of the descriptor, we need to normalize the descriptor of the generated 228 dimension, which is:

$$V' = \frac{V}{\|V\|} \tag{31}$$

Where $\|\bullet\|$ is the modulus of the vector and V' is the normalized descriptor. The algorithm is described as follows:

Require: Video V$(M \times N \times L)$:
Calculate F', $v(P_0)$
for i $= 1 \rightarrow$ M **do**
  for i $= 1 \rightarrow$ N **do**
    for i $= 1 \rightarrow$ L **do**
      Calculate $A_1 \sim A_3, B_1 \sim B_3$
      $\det(H_p) = \|\det(H)\|_2$
    end for
  end for
end for
Searching feature points X(x, y, t) by non-maximal suppresion
for each in X
  $m_w = \sum_w F_x + \sum_w F_y$
  $\theta_w = \arctan(\sum_w F_x / \sum_w F_y)$
  Calculate $F'_x, F'_y F'_t$
  $V = [\sum F'_x, \sum |F'_x|, \sum F'_y, \sum \left|F'_y\right|, \sum F'_t, \sum |F'_t|]$
  Normilization
end for
Return V'

VI. EXPERIMENTAL WORK

In this part, we first introduce the experimental settings and the process used, and then compared the experimental results with those obtained by the use of different algorithms. Finally, we ran a video classification experiment on the UCF101 dataset.

A. EXPERIMENTAL SETTINGS

Two datasets are used to evaluate our proposed methods: Weizman and UCF101. The Weizman dataset consists of 10 different types of action videos, each type of action has nine or 10 clips, and, because the number of videos is small, we use it to test the efficiency of different algorithms detecting and describing features. The UCF101 dataset includes 101 actions and 13,320 clips which have been collected on YouTube. It is an extension of UCF50 dataset generated by adding another 51 action classes and it is divided into five types: sports, playing musical instrument, human-object interaction, body-motion only, and human-human interaction. The clips of one action class are divided into 25 groups which contain between four and seven clips each. The clips in one group share some common characteristics, such as the background or actors. All clips have a fixed frame rate and a resolution of 320×240.

In this experiment, the overall experimental framework process is as shown in Fig. 6. Firstly, as a basic part of the whole experiment, we detect and describe the spatio-temporal points of interest. In the experiment, we use the UMAMV-SURF algorithm to detect the feature points and

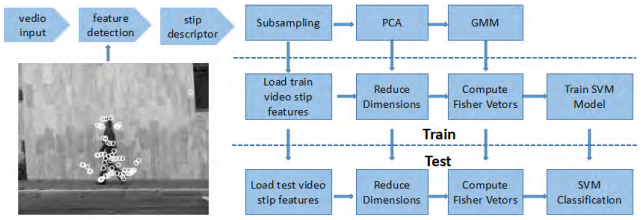

FIGURE 6. Action recognition framework.

get 228-dimensional descriptors. In the pre-processing of the obtained descriptors, we first sample the descriptors, and each video will obtain 2000 feature descriptors, if the number of feature descriptors in some videos is less than 2000, we will take all the descriptors of the video as our experimental sample. Then, the dimensions of the sub-sampled feature descriptors were reduced to half of their original dimensions by principal component analysis (PCA). Then, the Gaussian mixture models were trained by using the half-sized features. We used the Gaussian mixture model (GMM) pre-processed results, including weight, mean, covariance, and so on, as the input for coding. In this experiment, we used the Fisher vector coding method [36] to encode the selected descriptors, and then, used part of the video as training data to train the Support Vector Machine (SVM) model, finally utilized the trained SVM model to classify the test videos.

Here, the measurement of video classification accuracy is divided into one-class classification accuracy of video recognition and mean classification accuracy of multi-class video recognition. The classification accuracy of the i-th class is recorded as $acc(i)$.

$$acc(i) = \frac{cornum(i)}{testnum(i)} \tag{32}$$

Where $testnum(i)$ is the number of videos for testing and $cornum(i)$ is the number of videos correctly classified. Then, the mean accuracy of the n-class classification process is given by:

$$acc = \frac{\sum cornum(i)}{\sum testnum(i)} \tag{33}$$

B. EXPERIMENTAL RESULTS

Here, we first use different algorithms to detect feature points and perform feature point distribution experiments; Secondly, we classify the ten types of videos randomly selected from the UCF101 dataset, and test them by using different algorithms and compare the experimental results; in the third experiment, we test five major types of actions, *i.e.*, sports, playing musical instrument, human-object interaction, body-motion only, and human-human interaction, respectively to test the scene adaptability of our proposed algorithm; Fourthly, we tested the entire UCF101 and HDMB51 dataset to ascertain the mean classification accuracy of the proposed algorithm; Finally, the time required to run the different algorithms was compared from the perspective of the efficiency of the algorithm;

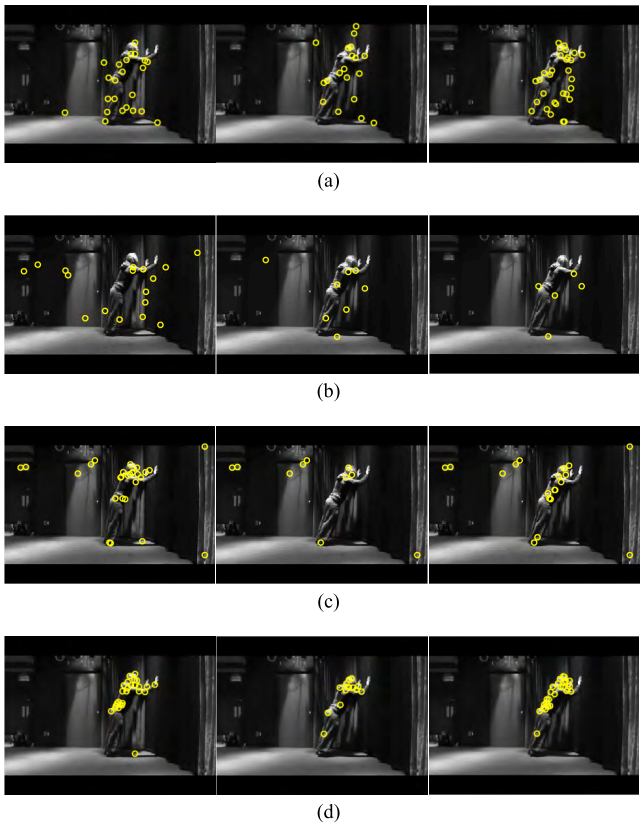

FIGURE 7. Using 3D Harris, 3D SIFT, SURF, and the algorithm proposed in this paper to detect the feature points of a video named "v_WallPushups_g03_c05" in UCF101 and the position distribution in video frames 8, 35, 43, and 63. (a) Feature points detected by 3D Harris algorithm. (b) Feature points detected by 3D SIFT algorithm. (c) Feature points detected by traditional SURF algorithm. (d) Feature points detected by UMAMV-SURF algorithm.

1) FEATURE POINT POSITION DISTRIBUTION EXPERIMENT

In this experiment, we focus on the position of the UCF101 dataset. We use different algorithms to detect the feature points of a video named "v_WallPushups_g03_c05" in UCF101, and the position distribution in video frames 8, 35, and 63, which are randomly selected from the dataset.

The detection results are shown in Fig. 7, which shows the distributions of the STIPs in the 8th, 35th, 43rd, and 63rd frames (randomly selected?) determined using the 3D Harris, 3D SIFT, the traditional SURF, and UMAMV-SURF algorithms, respectively. It can be seen from the experimental results that the 3D Harris detection algorithm is used to detect abundant feature points which are mostly found on the moving target, but there remain a few STIPs in the video background. Fewer feature points are detected by the 3D SIFT detection algorithm, and some of them are located in the video background. Compared to the previous three algorithms, most of the feature points extracted using the proposed UMAMV-SURF algorithm are on the moving target with low noise, and most points are concentrated in the upper part of the human body. This is because the algorithm proposed in this paper not only takes into account the local appearance information of

the video but also considers the correlation between the video frames and the motion information. Therefore, the feature points that contained the motion information can be detected from the video, and the background noise is effectively suppressed and it can better represent significant changes in the video. In summary, the UMAMV-SURF algorithm added the motion information under the geometric algebraic framework preserves the original robustness and effectiveness of the SURF algorithm, and most points can reflect motion changes in the video.

2) TEN RANDOMLY SELECTED CLASSES OF VIDEO CLASSIFICATION EXPERIMENTS

First of all, we randomly select 10 classes of videos from the UCF101 dataset, and use different algorithms to classify the videos. In this experiment, the videos we detected are ApplyLipstick, BandMarching, BreastStroke, CuttingInKitchen, HorseRiding, JumpingJack, PlayingGuitar, StillRings, Taichi, and YoYo. Fig. 8 shows a comparison of the classification accuracies. The mean classification accuracies of the 3D Harris, 3D SIFT, and the algorithm proposed in this paper are 81.16%, 78.67%, and 90.17%, respectively.

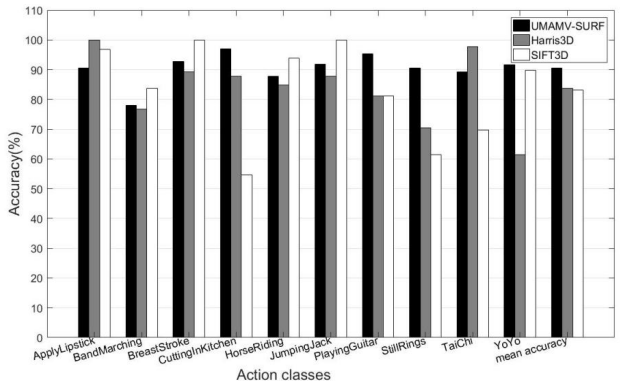

FIGURE 8. Using different algorithms to test the classification results of ten randomly selected types of video.

From the comparison of the classification results, it can be seen that, although the classification accuracy of the algorithm proposed is not the best in every class in the classification experiment, but achieves the highest average accuracy of outperform the other two algorithms, respectively. The reason can be seen clearly from the experiment involving feature point distribution: most of the feature points detected by the proposed algorithm are focused on the moving target, but in the other two algorithms, the proportion of feature points in the background is significantly higher than that of the proposed algorithm. Furthermore, the other two algorithms only deal with appearance information, and do not take any motion information into account. In addition, in this paper, the UMAMV-SURF feature description algorithm uses geometric algebra and the means of rotation response values and their statistical accumulation, which makes the descriptor more robust. Based thereon, the algorithm has higher classification accuracy for the ten types of random video experiments.

TABLE 1. Comparison of classification accuracies.

Method	STIP+BoVW[37]	UMAMV-SURF
Sports	50.54%	**80.32%**
PMI	37.42%	**72.81%**
HOI	38.52%	**71.56%**
BMO	36.26%	**75.38%**
HHI	44.14%	**78.12%**

3) CLASSIFICATION RESULTS FOR THE FIVE DIFFERENT ACTION GROUPS

The UCF101 dataset is divided into sports, playing a musical instrument, human-object interaction, body-motion only, and human-human interaction. The proposed method is compared with STIP-BoVW. Both two methods achieves best accuracy on sports group, since the group of sports films involve a majority of typical, simple behaviors, as well as simpler backgrounds, compared to other videos. Hence, they are easier to classify. Different from sports group, the video background of human object interaction is quite complex therein. In addition, the movement of other objects in the video also imposes difficulties during classification which is why human-object interaction returns such low classification accuracy. The result of the proposed method in this paper is shown in Table 1. Compared to experimental results [37], we can see that our algorithm has advantages in the five groups of classification experiments.

In this part of the experiment, the results obtained by using the UMAMV-SURF algorithm are significantly better than other results [37]. One possible reason is that we use the spatio-temporal UMAMV-SURF algorithm to detect feature points with their motion information under the geometric algebra computational framework. From the experimental results in Experiment 6.2.1, we can see that the distribution of feature points detected by our algorithm is more concentrated in positions of larger motion amplitude and better represent motion invariance. Therefore, the higher amplitude of the motion, the higher the classification accuracy. The STIP + BoW-based approach only takes into account the appearance information of the video, it does not describe the motion information in video very well.

4) CLASSIFICATION RESULTS AND ANALYSES FOR THE UCF101 DATASET

In this section, we will classify all 101 classes in the entire dataset by using the proposed algorithm. The mean classification accuracies of some algorithms proposed in recent years on the UCF101 dataset are listed in Table 2.

It can be seen that the algorithm proposed in this paper is 79.17% accurate for the UCF101 dataset. The algorithm classification accuracy in this paper is 35.27% higher than the algorithm used elsewhere [37]. Although the feature points

TABLE 2. Comparison with state-of-the-art action classification models.

Method	Mean accuracy in UCF101
STIP+BoVW [37]	43.9%
"Slow fusion" spatio-temporal ConvNet [38]	63.3%
LRCN [39]	71.1%
C3D [40]	72.3%
Spatial stream ConvNet [41]	73.0%
UMAMV-SURF	**79.17%**

are both detected with local features, the other method [37] uses the 3D Harris feature combined with the Bag of Visual Words (BOVW) method; but, the method proposed in this paper is based on the geometric algebraic framework of a video, and uses a local feature detection and description algorithm which are more robust. Therefore, our proposed algorithm has higher classification accuracy on the dataset classification test, and the effect of this algorithm in classification is better than that proposed elsewhere [37]–[41]. Moreover, it is well known that the favorable effects arising from the use of deep-learning methods depend on the availability of large quantities of data: they do not perform well if the size of the data sample is small.

5) CLASSIFICATION RESULTS AND ANALYSES FOR HDMB51 DATASET

In this part, we use the mainstream deep learning algorithm and the algorithm proposed in this paper to classify the HDMB 51 dataset. Table 3 is the experimental result in HDMB 51 dataset. Although deep learning has got good results in the behavior recognition of large data sets, it is not fit well during the training process because of too much parameters in training process when it run in the datasets that relatively small such as hdmb51, However, the local feature based on geometric algebra proposed in this paper is still able to achieve good results in the HDMB51 dataset because the feature extraction is more accurate and controllable.

TABLE 3. Comparison of results obtained for the HMDB51 dataset with those using neural network models.

Method	HMDB51(%)
Adaptive RNN-CNNs[43]	61.1
Rank Pooling+CNN[44]	65.8
Three-stream CNNs [45]	68.3
UMAMV-SURF	**75.7**

6) COMPUTATIONAL TIME COMPARISON

In this subsection, we compare the computational time of our proposed UMAMV-SURF with other hand-craft algorithm of 3D Harris, 3D SIFT on the relatively small dataset of Weizman for convenience, The evaluation computer has an CPU of 4 GHz and 16 GB random-access memory.

Table 4 lists the times taken to detect and describe 2000 points for each video in the dataset. As shown in Table 3, the time spent by using 3D Harris and 3D SIFT are 8994 s and 10,106 s, respectively, and the time taken by our proposed algorithm is only 3170 s.

TABLE 4. Time demand for different algorithms to detect and describe the Weizman dataset.

	3D Harris	3D SIFT	UMAMV-SURF
Detection& description time	8994 s	10106s	3170 s

The reason our proposed method achieves the shortest time is that the calculation process of detection and description is simplified by using approximate calculation methods such as integral video and box filter. While, the computational times of 3D Harris and 3D SIFT are linearly proportional to the number of detected feature points. Moreover, in the process of feature description, the integral video generated during the process of feature detection is used to reduce the time spent in the process of description. Therefore, the UMAMV-SURF algorithm proposed in this paper bestows a significant advantage in the time consumed.

VII. CONCLUSION

In this paper, based on the geometric algebra model (UMAMV) of a video, the feature detection and description algorithms, UMAMV-SURF detector and UMAMV-SURF descriptor, for uniformly analyzing video spatial information and motion information are proposed. Experimental results from behavior classification experiments show that the algorithm proposed in this paper does achieve a better classification effect than traditional algorithms. Although deep learning algorithms have seen many breakthroughs in video action classification in recent years, there remain several problems: complex and long-time calculation and requiring large training data. Therefore, deep learning methods are obviously not suitable for small sample datasets. The traditional STIP + SVM method just compensates for recognition and classification problems on small sample datasets, therefore, our study also provides meaningful research directions for future work in the cognate area of intelligent video analysis.

REFERENCES

[1] I. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. Int. Conf. Comput. Vis.*, vol. 1, Oct. 2003, pp. 432–439.
[2] A. Kovashka and K. Grauman, "Learning a hierarchy of discriminative space-time neighborhood features for human action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, Jun. 2010, pp. 2046–2053.
[3] L. Liu, L. Shao, X. Li, and K. Lu, "Learning spatio-temporal representations for action recognition: A genetic programming approach," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 158–170, Jan. 2015.
[4] X. Li, Z. Wang, and X. Lu, "Surveillance video synopsis via scaling down objects," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 740–755, Feb. 2016.
[5] X. Zhen, L. Shao, and X. Li, "Action recognition by spatio-temporal oriented energies," *Inf. Sci.*, vol. 281, pp. 295–309, Oct. 2014.
[6] Y. Li, W. Liu, and Q. Huang, "Traffic anomaly detection based on image descriptor in videos," *Multimedia Tools Appl.*, vol. 75, no. 5, pp. 2487–2505, Mar. 2015.

[7] Q. Ling, S. Deng, F. Li, Q. Huang, and X. Li, "A feedback-based robust video stabilization method for traffic videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 3, pp. 561–572, Mar. 2016.
[8] X. Jiang, F. Zhong, Q. Peng, and X. Qin, "Online robust action recognition based on a hierarchical model," *Vis. Comput.*, vol. 30, no. 9, pp. 1021–1033, 2014.
[9] H. Zhang and L. E. Parker, "4-dimensional local spatio-temporal features for human activity recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2011, pp. 2044–2049.
[10] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *Int. J. Comput. Vis.*, vol. 103, no. 1, pp. 60–79, May 2013.
[11] C. Wang and C. Dong, "Spatial-temporal words learning for crowd behavior recognition," *Int. J. Sci. Eng. Investigations*, vol. 1, no. 3, 2012.
[12] Y. Li, W. Liu, Q. Huang, and X. Li, "Fuzzy bag of words for social image description," *Multimedia Tools Appl.*, vol. 75, no. 3, pp. 1371–1390, 2016.
[13] F. Jiang, J. Yuan, S. A. Tsaftaris, and A. K. Katsaggelos, "Anomalous video event detection using spatiotemporal context," *Comput. Vis. Image Understand.*, vol. 115, no. 3, pp. 323–333, 2011.
[14] Y. Li, Q. Huang, W. Xie, and X. Li, "A novel visual codebook model based on fuzzy geometry for large-scale image classification," *Pattern Recognit.*, vol. 48, no. 10, pp. 3125–3134, Oct. 2015.
[15] S. Oh *et al.*, "A large-scale benchmark dataset for event recognition in surveillance video," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, 2011.
[16] C. Chattopadhyay and A. K. Maurya, "Multivariate time series modeling of geometric features of spatio-temporal volumes for content based video retrieval," *Int. J. Multimedia Inf. Retrieval*, vol. 3, no. 1, pp. 15–28, 2014.
[17] M. Al Ghamdi, L. Zhang, and Y. Gotoh, "Spatio-temporal SIFT and its application to human action classification," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 301–310.
[18] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. 4th Alvey Vis. Conf.*, 1988, vol. 15, no. 50, pp. 147–151.
[19] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. IEEE Int. Workshop Vis. Surveillance Perform. Eval. Tracking Surveillance*, Oct. 2005, pp. 65–72.
[20] Y. Ke, R. Sukthankar, and M. Hebert, "Efficient visual event detection using volumetric features," in *Proc. ICCV*, Oct. 2005, pp. 166–173.
[21] A. Oikonomopoulos, I. Patras, and M. Pantic, "Spatiotemporal salient points for visual recognition of human actions," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 3, pp. 710–719, Jun. 2005.
[22] I. Laptev, B. Caputo, C. Schüldt, and T. Lindeberg, "Local velocity-adapted motion events for spatio-temporal recognition," *Comput. Vis. Image Understand.*, vol. 108, no. 3, pp. 207–229, 2007.
[23] H. Shabani, D. A. Clausi, and J. S. Zelek, "Towards a robust spatio-temporal interest point detection for human action recognition," in *Proc. Can. Conf. Comput. Robot Vis. (CRV)*, May 2009, pp. 237–243.
[24] C. Liu, Y. Chen, and M. Wang, "Spatio-temporal interest point detection in cluttered backgrounds with camera movements," *J. Image Graph.*, vol. 18, no. 8, pp. 982–989, 2013.
[25] J. Weickert, "A review of nonlinear diffusion filtering," in *Scale-Space Theory in Computer Vision*. Berlin, Germany: Springer, 1997, pp. 1–28.
[26] T.-H. Yu, T.-K. Kim, and R. Cipolla, "Real-time action recognition by spatiotemporal semantic and structural forest," in *Proc. BMVC*, 2010, pp. 52.1–52.12.
[27] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2006, pp. 430–443.
[28] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008.
[29] K. Mikolajczyk and S. Cordelia, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.
[30] G. Willems, T. Tuytelaars, and L. Van Gool, "An efficient dense and scale-invariant spatio-temporal interest point detector," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2008, pp. 650–663.
[31] Y. Guo, "Spatio-temporal SIFT interest points detection in videos," Zhejiang Univ., Hangzhou, China, Tech. Rep., 2009.
[32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
[33] C. Li, B. Su, J. Wang, H. Wang, and Q. Zhang, "Human action recognition using multi-velocity STIPs and motion energy orientation histogram," *J. Inf. Sci. Eng.*, vol. 30, no. 2, pp. 295–312, 2014.
[34] J. Lasenby, W. J. Fitzgerald, A. N. Lasenby, and C. J. L. Doran, "New geometric methods for computer vision: An application to structure and motion estimation," *Int. J. Comput. Vis.*, vol. 26, no. 3, pp. 191–213, Feb. 1998.
[35] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001, pp. I-511–I-518.
[36] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed Fisher vectors," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3384–3391.
[37] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, vol. abs/1212.0402, 2012.
[38] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. CVPR*, Jun. 2014, pp. 1725–1732.
[39] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2625–2634.
[40] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "C3D: Generic features for video analysis," *CoRR*, vol. abs/1412.0767, Dec. 2014.
[41] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568-576.
[42] M.-Y. Chen and A. Hauptmann, "MoSIFT: Recognizing human actions in surveillance videos," Dept. Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-CS- 09-161, 2009.
[43] M. Xin, H. Zhang, H. Wang, M. Sun, and D. Yuan, "ARCH: Adaptive recurrent-convolutional hybrid networks for long-term action recognition," *Neurocomputing*, vol. 178, pp. 87–102, Feb. 2016.
[44] B. Fernando, E. Gavves, J. Oramas M, A. Ghodrati, and T. Tuytelaars, "Rank pooling for action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 773–787, Apr. 2017.
[45] L. Wang, L. Ge, R. Li, and Y. Fang, "Three-stream CNNs for action recognition," *Pattern Recognit. Lett.*, vol. 92, pp. 33–40, Jun. 2017.
[46] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2017.
[47] M. A. Uddin, J. B. Joolee, A. Alam, and Y. K. Lee, "Human action recognition using adaptive local motion descriptor in spark," *IEEE Access*, vol. 5, pp. 21157–21167, 2017.
[48] Y. Hou, S. Wang, P. Wang, Z. Gao, and W. Li, "Spatially and temporally structured global to local aggregation of dynamic depth information for action recognition," *IEEE Access*, vol. 6, pp. 2206–2219, 2018.
[49] Y. Li, R. Xia, Q. Huang, W. Xie, and X. Li, "Survey of spatio-temporal interest point detection algorithms in video," *IEEE Access*, vol. 5, no. 2, pp. 10323–10331, Feb. 2017.

YANSHAN LI received the M.Sc. degree from the Zhejiang University of Technology in 2005 and the Ph.D. degree from the South China University of Technology, China, in 2015. He is currently an Associate Professor with the ATR National Key Laboratory of Defense Technology, Shenzhen University, China. His research interests cover computer vision, machine learning, and image analysis.

CONGZHU YANG is currently pursuing the M.S. degree in signal and information processing with the ATR National Key Laboratory of Defense Technology, Shenzhen University. His research interests include machine learning, video processing, and pattern recognition.

LI ZHANG received the bachelor's degree in radio engineering and the master's degree in communications and information systems from the Harbin Institute of Technology in 1997 and 1999, respectively, and the doctorate degree in communications and information systems from the South China University of Technology in 2002. He is currently a Professor with the School of Information Engineering, Shenzhen University. His research interests include digital signal processing and information security.

RONGJIE XIA received the B.E. degree in information and engineering from Shenzhen University, Shenzhen, China, in 2017, where he is currently pursuing the M.S. degree in signal and information processing with the ATR National Key Laboratory of Defense Technology. His research interests include intelligent information processing, video processing, and pattern recognition.

LEIDONG FAN received the B.E. degree in information and engineering from Shenzhen University, Shenzhen, China, in 2015, where he is currently pursuing the M.S. degree in signal and information processing with the ATR National Key Laboratory of Defense Technology. His research interests include image processing and pattern recognition.

WEIXIN XIE received the Degree from Xidian University, Xi'an. In 1965, he joined Xidian University as a Faculty Member. From 1981 to 1983, he was a Visiting Scholar with the University of Pennsylvania, USA. In 1989, he was invited to the University of Pennsylvania, as a Visiting Professor. He is currently with the School of Information Engineering, Shenzhen University, China. His research interests include intelligent information processing, fuzzy information processing, image processing, and pattern recognition.