Abstract: This study presents a novel human action recognition method based on the sequences of depth maps, which provide additional body shape and motion information for action recognition. First, the authors divide each depth sequence into a number of sub-sequences. All these sub-sequences are of uniform length. By controlling vague boundary (VB), they construct a VB-sequence which consists of an original sub-sequence and its adjacent sequences. Then, each depth frame in a VB-sequence is projected onto three orthogonal Cartesian planes, and the absolute value of the difference between two consecutive projected maps is accumulated to form a depth motion map (DMM) to describe the dynamic feature of a VB-sequence. Finally, they concatenate the DMMs of all the VB-sequences in one video sequence to describe an action. Collectively, they call them VB division of depth model. For classification, they apply robust probabilistic collaborative representation classification. The recognition results applied to the MSR Action Pairs, MSR Gesture 3D, MSR Action3D, and UTD-MHAD datasets indicate superior performance of their method over most existing methods.

1 Introduction
In computer vision and pattern recognition, human action recognition is an active branch. There are many potential applications in human–machine interactivity including video analysis, surveillance systems, robotics, and so on. In the past, research has mainly focused on learning and recognising actions from image sequences taken by visible light cameras [1–3]. There are inherent limitations of this type of data source, e.g. it is sensitive to colour and illumination changes, occlusions, and background clutter. With the recent advent of the cost-effective Kinect, depth cameras have received a great deal of attention from researchers. At present, there are major two categories to extract features from depth map sequence: the skeleton joint points and the original depth maps. Recently, many recognition methods based on the estimated three-dimensional (3D) skeleton joints are presented [4–6]. However, these methods need reliable skeleton data while the estimation cannot guarantee the skeleton joint points are always right. Better than the 3D skeleton joint points, the original depth maps not only provide additional body shape but also offer motion information to distinguish actions. So, the original depth maps maybe the better choice to deal with the recognition problem.

In this paper, we focus on recognising human actions by using the original depth map sequences. A new descriptor based on vague boundary division of depth model (VBDDM) is presented and proved to be an effective way to solve human recognition problem. Depth motion maps (DMMs) of an entire video [7] may lose the time information in an action [e.g. actions of ‘pick up’ and ‘put down’ in Action Pairs dataset share similar motions and shapes, yet, the co-occurrence of the object shape and the hand motion are presented in different spatiotemporal order (see Fig. 1)]. To further model the time variants, in our method, we divide each depth sequence into a number of sub-sequences. All these sub-sequences are of uniform length. By controlling VB, we construct a VB-sequence which consists of an original sub-sequence and its adjacent frames. Then, DMMs in three orientations of each VB-sequence is calculated, which are resized into a fixed size in the next step. Different from [7, 8], we do not calculate histograms of the oriented gradients (HOGs) [9] of DMMs to encode a descriptor, pixel values are used directly as the feature to reduce computational complexity of the feature extraction process. Finally, we concatenate descriptors of all the VB-sequences in one video sequence to obtain the VBDDM descriptor. For the classification, we use the robust probabilistic collaborative representation classification (R-ProCRC) proposed by Yang et al. [10], which shows superior performance to many popular classifiers. The experiments on MSR Action3D [11], MSR Action Pairs [12], MSR Gesture 3D [13], and UTD-MHAD [14] datasets demonstrate that the proposed method is more robust and achieves better recognition accuracy than most existing methods.

The rest of this paper is organised as follows. In Section 2, related work is presented. In Section 3, the details of generating VBDDM feature descriptors are offered. The principle of the classification R-ProCRC is presented in Section 4. A variety of experimental results and discussion are presented in Section 5. Conclusions are finally given in Section 6.

2 Related work
Owing to the wide range of applications for action recognition, researchers have been actively studying this topic and have achieved promising results. Some excellent surveys have been published based on conventional RGB cameras. These approaches mainly focus on detection and representation of space–time volumes such as spatiotemporal features and trajectories. For example, in [15], spatiotemporal interest points coupled with an support vector machine (SVM) classifier was used to achieve human action recognition. Scale-invariant feature transform (SIFT)-feature trajectories modelled in a hierarchy of three abstraction levels were used to recognise actions in video sequences in [16]. Motion energy images and motion-history images were introduced in [17] as motion templates to model spatial and temporal characteristics of human actions in videos. However, a major shortcoming associated with using these colour-based or intensity-based methods is the sensitivity of recognition to illumination variations, limiting the recognition robustness.

With the development of computing ability and the improvement of sensor techniques, a large number of methods have been
proposed to address action recognition using various data modalities from depth sensors. Depth maps and skeletal joints are two most commonly used data modalities in this area. The work by Wang et al. [11] utilised both skeleton and point cloud information. They proposed to extract local occupancy patterns (LOPs) features at each joint and learn an actionlet ensemble model to represent each action and to capture the intra-class invariance. Luo et al. [18] proposed a new discriminative dictionary learning algorithm (DL-GSGC) that incorporated both group sparsity and geometry constraints to better represent skeleton features. Xia et al. [5] proposed a feature called histogram of 3D joint locations using modified spherical coordinates. Skeleton-based features are easier to extract and require less computational cost and memory than depth map-based features, while it can hardly work when the human body is partly in view, and the estimation is not reliable or can fail when the person touches the background or when the person is not in an upright position.

In the area of using original depth data, Oreifej and Liu [12] proposed to describe the depth sequence using a histogram capturing the distribution of the surface normal orientation in the 4D space of time (HON4D), depth, and spatial coordinates. Yang et al. [7] proposed to project depth frames onto three orthogonal Cartesian planes for the purpose of characterising the motion of an action and then the HOGs computed from DMMs were used as feature descriptors. Chen et al. [19] also employed DMMs to capture motion cues and use local binary patterns (LBPs) to gain a compact feature representation. Two types of fusion consisting of feature-level fusion and decision-level fusion are considered. Yang and Tian [20] proposed to subdivide a depth video into a set of space-time grids, and then adopted a novel scheme of aggregating the low-level polynomials into the super normal vector. Cai and Zhang computed DMMs and depth static maps (DSMs) to obtain the temporal pyramid of depth model (TPDM) [8], then spatial pyramid HOG (SPHOG) is computed from the TPDM for the representation of an action.

Chen and Liu [21] divided depth sequence into temporally overlapping depth segments which are used to generate three DMMs.

Multiple frame lengths of depth segments are utilised to cope with speed variations in actions. Then, the LBP descriptor is exploited to characterise local rotation invariant texture information in those patches.

3 Proposed method

A depth map can be used to capture the 3D structure and shape information. However, human action consists of a sequence of human postures and varies with time. Some actions even share similar postures yet differ in the order that the postures appear. To solve this problem, in this work we propose a new model named VBDDM. Framework about VBDDM for human recognition on depth maps is demonstrated in Fig. 2.

3.1 VB division

To model the time variation of the human action, we first divide the depth map sequence into a number of sub-sequences. Given a depth sequence \( X = \{X_i\}_{i=1}^N \) of an action sample, \( X_i \) denotes the \( i \)th depth map. \( N \) is number of the depth maps. The depth map sequence \( X \) is divided into \( \text{DIV} \) sub-sequences \( \{P_i\}_{i=1}^\text{DIV} \). Each sub-sequence can be denoted as \( P_i = \{X_{i(j-1)N/\text{DIV}}+1, \ldots, X_{iN/\text{DIV}}\} \). Clearly, in our method, we choose to segment the depth sequence of equal length. In this situation, the last segment maybe shorter or longer to make sure all the segments cover the depth sequence. By division strategy, each sub-sequence contains different characteristics for different action phases. Encoding features based on divided sub-sequences can better capture the change rule of the depth information. However, not all the actions performed by different subjects follow the same regularly changing rhythm. Human action recognition are one of the most complex recognition problem, the major reason is that human actions are closely influenced by different culture, personal character, and emotion shift. The speed that a same action performed by different subjects, or even performed by same subject at different time, can be different. For a video capture system which adopts fixed sampling rate, simply dividing the depth map sequence into a number of sub-sequences may neglect the potential speed variants.

To address this problem, we introduce a VB division method. As shown in Fig. 2. By controlling VB, we construct a VB-sequence which consists of an original sub-sequence and its adjacent frames. The \( j \)th VB-sequence of the \( k \)th action video \( X \) is denoted by: \( P_j = \{X_{i(j-1)N/\text{DIV}}+1−\alpha N/\text{DIV}, \ldots, X_{iN/\text{DIV}}+\alpha N/\text{DIV}\}_{i\in[1,\text{DIV}]} \). \( \alpha \) is the VB factor which controls the number of frames shared between adjacent sub-sequences. By considering the forward and backward depth information simultaneously, each VB-sequence is more robust to action speed. Now, we convert the action depth sample \( X \) into a number of VB-sequences \( \{P_j\}_{j=1}^\text{DIV} \).

3.2 VBDDM descriptor

Inspired by the work in [22], we calculate the DMM for each VB-sequence. A depth map is a 2D image taken by some kind of depth camera. Each pixel in the depth map indicates the distance from the camera to the surface of the object. Different from the 3D model, the 3D information of the unseen surface are missing in the depth map. Yet, depth map contains rich information to help us understand the object and it is quite easy to convert the depth map into point cloud. Thus, by using point cloud, all the depth maps in the action depth sample \( X \) are converted from 2D images into 3D space. Then, we project each point cloud derived from a depth map into three views: front view \( X_{F} \), side view \( X_{S} \), and top view \( X_{T} \). We get the motion energy for each frame by calculating the absolute difference between two consecutive maps without thresholding, which is same as [22]. Then, the DMM is obtained by stacking the motion energy across a...
between 0 and 1 to avoid large pixel values dominating the view. DMMs features through whole video sequence.

Confusion matrices for 3D Action Pairs dataset

Fig. 4 Confusion matrices for 3D Action Pairs dataset

a) DMMs features through whole video sequence

b) Proposed method

Table 1 Recognition rate on the 3D Action Pairs dataset (bold values indicate best results)

|                | Accuracy, % |
|----------------|-------------|
| Chen et al. [22]| 50.6        |
| (skeleton + LOP + pyramid) [11]| 82.2        |
| HON4D + Disc [12]| 97.9        |
| ours | 97.2 |

feature set. The normalised DMM\(_{j,v}\) is denoted by \(\overline{\text{DMM}}_{j,v}\). Now, we can obtain a new descriptor of the \(j\)th VB-sequence: \(H_j = [\text{vec}(\overline{\text{DMM}}_{j,fi}), \text{vec}(\overline{\text{DMM}}_{j,si}), \text{vec}(\overline{\text{DMM}}_{j,ti})]\), which concatenates three vectored \(\overline{\text{DMM}}_{j,v}\). vec(·) indicates the vectorisation operator. Meanwhile, we can have the descriptor of the whole depth sample \(X^T = [H_1, \ldots, H_j, \ldots, H_{\text{DIV}}]\) by concatenating the descriptors of all the VB-sequences, \(T\) is the matrix transpose.

Different from [7, 22], we calculate DMM just from the VB-sequences not the whole sequence. Cai and Zhang calculated DMM on sub-action which is similar with ours [8]; however, Xu et al. [8] also computed DMM and DSMs on the whole video sequence to form a spatial–temporal pyramid model, which is high-computational. Furthermore, the method in [8] divided the action video into sub-actions which have definite boundaries. While most actions are consecutive, information in the time domain will be lost wherever we divide the action. So, in our method every VB-sequence shares information from its adjacent sequence. In [7, 8], HOG descriptors of the DMM are computed to get the final representation of an action. However, in our method pixel values of DMM are used as features directly. Suppose \(M\) training examples are given, each depth sample sequence are processed to obtain the VBDDM feature \(H^{(i)}\). The final feature matrix for the \(M\) training examples can be denoted as \(\{H^{(i)}\}_{i \in \{1, \ldots, M\}}\). To keep the classification task more efficient, we perform principal component analysis (PCA) to reduce the dimensionality. We choose to retain 99% of the variance. Thus, we get a compact VBDDM descriptor set \(\{H^{(i)}\}_{i \in \{1, \ldots, M\}}\).
Table 2 Recognition rate (%) on the Gesture 3D dataset (bold values indicate best results)

| Method                  | Accuracy, % |
|-------------------------|-------------|
| Chen et al. [22]         | 90.7        |
| HonD + Dacc [12]         | 92.5        |
| Yang and Tian [20]       | 94.7        |
| Chen and Liu [21]        | 98.2        |
| ours                    | 97.3        |

Table 3 Action subsets and tests used in our experiments

| AS1                | AS2                | AS3                |
|--------------------|--------------------|--------------------|
| horizontal wave (2)| high wave (1)      | high throw (6)     |
| hammer (3)         | hand catch (4)     | forward kick (14)  |
| forward punch (5)  | draw x (7)         | side kick (15)     |
| high throw (6)     | draw tick (8)      | jogging (16)       |
| hand clap (10)     | draw circle (9)    | tennis swing (17)  |
| bend (13)          | two hand wave (11)| tennis serve (18)  |
| tennis serve (18)  | forward kick (14) | golf swing (19)    |
| pickup throw (20)  | side boxing (12)   | pickup throw (20)  |

4 Action classification using robust ProCRC classifier

Lin et al. presented a ProCRC-based classifier, named ProCRC [10], which employs a ProCRC framework to jointly maximise the probability that a test sample belongs to each class. ProCRC effectively makes use of the training samples from all classes to deduce the class label of a test sample. It possesses a clear probabilistic interpretation, and is very efficient to solve. Therefore, here we use this newly proposed classifier to solve our action recognition problem.

According to the preceding part of this paper, we have a collection of training samples from C classes \( \mathbf{H} = [\mathbf{H}_1, \ldots, \mathbf{H}_i, \ldots, \mathbf{H}_C] \), where \( \mathbf{H}_i \) is the data matrix of the \( i \)th class and each column of \( \mathbf{H}_i \) is a VBDDM feature calculated from a training sample. We view \( \mathbf{H}_i \) as the data matrix of an expanded class, and denoted by \( i \mu \) the label set of all candidate classes in \( \mathbf{H}_i \). Denoted by \( S \) the linear subspace collaboratively spanned by all samples in \( \mathbf{H}_i \). Then for each test sample \( y \) in the collaborative subspace \( S \), it can be represented as a linear combination of samples in \( \mathbf{H}_i \): \( y = \mathbf{H}_i \beta \), where \( \beta \) is the representation vector. See [10], the optimal representation vector \( \hat{\beta} \) can be calculated as

\[
\hat{\beta} = \arg\min_\beta \left\| y - \mathbf{H}_i \beta \right\|_2^2 + \lambda \left\| \beta \right\|_2^2 + \gamma \sum_{c=1}^C \left\| \mathbf{H}_c \beta - \mathbf{H}_c \beta_c \right\|_2^2
\]

(4)

In visual classification, partial corruption or occlusion often degrades the performance. It is well-known that the robustness of classification tasks can be enhanced by using \( l_1 \)-norm to characterise the loss function [10]. Therefore, (4) can be easily extended to its robust version R-ProCRC:

\[
\hat{\beta} = \arg\min_\beta \left\| y - \mathbf{H}_i \beta \right\|_1 + \lambda \left\| \beta \right\|_2^2 + \gamma \sum_{c=1}^C \left\| \mathbf{H}_c \beta - \mathbf{H}_c \beta_c \right\|_2^2
\]

(5)

Although the R-ProCRC model is convex, there is no closed-form solution to it. Yang et al. [10] adopted an iteratively reweighted least squares (IRLS) algorithm to compute \( \beta \). The coefficient vector \( \beta \) can be updated by

\[
\hat{\beta} = \left( \mathbf{H}_i^T \mathbf{W}_i \mathbf{H}_i + \gamma \sum_{c=1}^C \left( \mathbf{H}_c^T \mathbf{H}_c \right) \right)^{-1} \mathbf{H}_i^T \mathbf{W}_i y
\]

(6)

where \( \mathbf{W}_i \) refers to diagonal weighting matrix and \( \mathbf{H}_i(i,:) \) is the \( i \)th row of \( \mathbf{H}_i \). The diagonal weighting of \( \mathbf{W}_i \) can be calculated as

\[
W_{ii}(i,:) = 1/||\mathbf{H}_i(i,:) \beta - y||
\]

(7)

With the optimal solution vector \( \hat{\beta} \) estimated using the training samples for classification, the probability of a test sample \( y \) classified as class \( c \) can be computed by

\[
P(\hat{l}(y) = c) \propto \exp \left( - \left\| y - \mathbf{H}_i \beta \right\|_1 + \lambda \left\| \hat{\beta} \right\|_2^2 + \gamma \sum_{c=1}^C \left\| \mathbf{H}_c \beta - \mathbf{H}_c \beta_c \right\|_2^2 \right)
\]

(8)

Note that \( \left\| y - \mathbf{H}_i \beta \right\|_1 + \lambda \left\| \hat{\beta} \right\|_2^2 \) is the same for all classes, thus we can omit this part in computing \( P(\hat{l}(y) = k) \). The classification rule

Table 4 Performance of our method on Action3D dataset, compared with previous approaches (bold values indicate top two results)

| Space–time occupancy pattern [25] | Chen et al. [22] | DMM-LBP-FF [19] | TPDM-SPHOG [8] | Ours       |
|----------------------------------|-----------------|-----------------|---------------|------------|
| Test One                         |                 |                 |               |            |
| AS1                              | 98.2            | 97.3            | 96.7          | 98.0       |
| AS2                              | 94.8            | 96.1            | 100           | 96.7       |
| AS3                              | 97.4            | 98.7            | 99.3          | 100        |
| average                          | 96.8            | 97.4            | 98.7          | 98.2       |
| Test Two                         |                 |                 |               |            |
| AS1                              | 99.1            | 98.6            | 100           | 98.6       |
| AS2                              | 97.0            | 98.7            | 100           | 100        |
| AS3                              | 98.7            | 100             | 100           | 100        |
| average                          | 98.3            | 99.1            | 100           | 99.5       |
| Cross Test                       |                 |                 |               |            |
| AS1                              | 84.7            | 96.2            | 98.1          | 99.1       |
| AS2                              | 81.3            | 83.2            | 92.0          | 92.9       |
| AS3                              | 88.4            | 90.5            | 94.6          | 96.4       |
| average                          | 84.8            | 90.0            | 94.9          | 96.1       |
can then be formulated as

$$l(y) = \arg\max_c \exp(-\|H_{\hat{c}}b - H_{\hat{c}}b_c\|^2_2)$$ (9)

5 Experiment and discussion

We extensively experimented on the proposed ideas using four standard 3D activity datasets including MSR Action Pair [12], MSR Gesture 3D [13], MSR Action3D [11], and UTD-MHAD [14]. For each dataset, we perform experiments to analyse the influence of parameter $\alpha$. In all experiments, each action sequence is divided into $\text{DIV VB}$-sequences. The value of $\text{DIV}$ is different according to various human action datasets. Then, three DMMs on the orthogonal Cartesian planes are calculated for every VB-sequence. So, we will have $\text{DIV} \times 3$ DMM, for one complete action sequence. Meanwhile, the comparison on the classifier SVM [23], $l_1$-norm CRC [24], R-ProCRC, and $l_2$-norm CRC [22] is made to show the superiority of R-ProCRC.

5.1 MSR Action Pairs 3D dataset

The MSR Action Pairs 3D [12] is a paired-activity dataset of depth sequences captured by a depth camera. It contains 12 activities of 10 subjects with each subject performing each activity three times. This dataset is collected to investigate how the temporal order affects activity recognitions.

The evaluation setup same as [12] is used in our experiment. Before extracting feature, we remove the first and last four frames of each sequence, in which subjects are mostly standstill. Then, we set $\alpha = 0.05$ for this dataset and divide them into six subsets. The first five actors are used for testing and the rest for training. The detailed comparison to other approaches is demonstrated in Table 1. We compare our method with three methods on this dataset. First, we compute motion maps for the whole action sequence, which are vectored as feature and use the $l_2$-norm CRC [22] to train them. Second, we compare to skeleton-based pair-wise features and LOP features as described in [11], which are enhanced by applying a temporal pyramid. The accuracy has reached to 82.2%. Finally, we compare our method to HON4D [12], though we only have just 0.5% improvement, we can consider our method has reached a relatively better result.

Confusion matrices are shown in Fig. 4. Fig. 4a presents the confusion matrices of the method [22] which extracts DMMs features through whole video sequence. Fig. 4b is the confusion matrices of the proposed method. There are many methods utilising DMMs to extract features, many of them use DMMs calculated from whole video (e.g. [7, 22]), which may lose temporal

Fig. 6 Confusion matrices for Action3D subsets on Cross Test

a AS1
b AS2
c AS3

This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)
Table 5 Human actions in UTD-MHAD

| Action Pair | Gesture 3D and UTD-MHAD |
|-------------|-------------------------|
| 1           | right arm swipe to the left (swipe_left) |
| 2           | right arm swipe to the right (swipe_right) |
| 3           | right-hand wave (wave) |
| 4           | two hand front clap (clap) |
| 5           | right arm throw (throw) |
| 6           | cross arms in the chest (arm_cross) |
| 7           | basketball shoot (basketball_shoot) |
| 8           | right-hand draw x (draw_x) |
| 9           | right-hand draw circle (clockwise) (draw_circle_CW) |
| 10          | right-hand draw circle (counter clockwise) (draw_circle_CCW) |
| 11          | draw triangle (draw_triangle) |
| 12          | bowling (right hand) (bowling) |
| 13          | front boxing (boxing) |
| 14          | tennis swing from right (tennis_swing) |
| 15          | arm curl (two arms) (arm_curl) |
| 16          | tennis serve (tennis_server) |
| 17          | two hand push (push) |
| 18          | right-hand knock on door (knock) |
| 19          | right-hand catch an object (catch) |
| 20          | right-hand pick up and throw (pickup_throw) |
| 21          | Wearable inertial sensor on right thigh |
| 22          | jogging in place (jog) |
| 23          | walking in place (walk) |
| 24          | sit to stand (sit2stand) |
| 25          | stand to sit (stand2sit) |
| 26          | forward lunge (left foot forward) (lunge) |
| 27          | squat (two arms stretch out) (squat) |

Table 6 Performance of our method on UTD-MHAD

| Method | Accuracy, % |
|--------|-------------|
| Chen et al. [22] | 66.1 |
| ours | 85.1 |

The Gesture 3D dataset [13] is a hand gesture dataset of depth sequences captured by a depth camera. It contains a set of dynamic gestures defined by American Sign Language. There are 12 gestures in the dataset: ‘bathroom’, ‘blue’, ‘finish’, ‘green’, ‘hungry’, ‘milk’, ‘past’, ‘pig’, and ‘store’, ‘where’, ‘j’ and ‘z’. Sequences of depth maps sampled from ‘milk’ are shown in Fig. 5. In total, the dataset contains 333 depth sequences, and is considered challenging mainly because of self-occlusion issues.

We follow the experiment setup in [12] and obtain the accuracies described in Table 2. Note that, before we do our experiment on this dataset, we removed the all-zero frames in these sequences. Since when we divide the video sequence into three parts, there are so many all-zero frames and we cannot calculate the DMM, correctly. Meanwhile, we removed the first three frames which are almost static. In this experiment, DIV = 3 and α = 0.2.

As we can see in Table 2, our method has obtained a relatively high accuracy on this dataset. Among these methods, Chen and Liu [21] achieve the best performance. In [21], a different strategy of temporally overlapping depth segments was proposed which adopts three temporal levels and then LBPs descriptor was exploited to characterise local rotation invariant texture information in those patches. In our method, all the depth map sequences are divided into DIV sub-sequences. Our method just uses one temporal level and does not need to calculate LBP descriptor which leads to relative lower computational complexity.

5.3 MSR Action3D dataset

The MSR Action3D [11] is a public dataset with sequences of depth maps captured by an RGBD camera. It includes 20 action categories performed by 10 subjects facing to the camera during performance. Each action was performed two or three times by each subject. The depth maps are with the resolution of 320 × 240. To facilitate a fair comparison, we follow the same experimental settings as [8, 22] to split 20 categories into three subsets as listed in Table 3. As for each subset, there are three different tests, i.e. Test One (One), Test Two (Two), and Cross Subject Test (Cr-Sub). In Test One, 1/3rd of the

![Fig. 7 Influence of α on different datasets](image)

*a Test one on MSR Action3D dataset
*b Action Pair, Gesture 3D and UTD-MHAD datasets

This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)
subset is used as training and the rest as testing; in Test Two, 2/3rd of the subset is used as training and the rest as testing; in Cross Subject Test, we take subjects 1, 3, 5, 7, 9 for training and 2, 4, 6, 8, 10 for testing. The Cross Subject Test is challenging, because different subjects have their own styles to perform actions, so there are large variations among training and testing actions.

In all the three subsets, we set the DIV = 3 and $\alpha = 0.8$. Meanwhile, the fix sizes of DMMs are totally the same with [22]. In this experiment, we choose $\alpha = 0.8$. Then, we compare our method with other methods on three subsets, and the overall accuracies are also provided for each test, as shown in Table 4. Chen et al. [22] directly use the pixels of the DMMs (DMM$_v$) calculated from a whole video sequence. Xu et al. [8] proposed a spatio-temporal pyramid model, in which DMM$_v$ and DSMs are calculated and then HOG [9] is used to represent these maps. Chen et al. [19] employed DMMs to capture motion cues and use LBPs to gain a compact feature representation. In Table 4, we compare the fusion-level fusion approach (DMM-LBP-FF) with our method which has a higher average accuracy than decision-level fusion approach (DMM-LBP-DF). TPDM-SPHOG [8] is an efficient feature and has reached a state-of-the-art performance in Action3D dataset; however, the disadvantage of this method is with great amount of computational complexity. In our method, we only need to calculate DMMs (DMM$_v$$_v$) of those sub-sequences and pixels of DMM$_v$ (not the HOG features) are used directly to obtain the feature. As we can see in Table 4, our method has obtained comparable performance with these methods and is sometimes superior to them.

The confusion matrices of action set 1 (AS1), AS2, and AS3 on Cross Subject Test are shown in Fig. 6. We can see the proposed method on AS2 has a relatively low accuracy than the other two subsets. The mainly confused actions are hand catch (4), draw x (7), draw tick (8), and draw circle (9). For these actions, only the moving of arms are critical for action classification. Yet DMMs capture the variants of the whole depth map. The moving of the body and the unrelated trunks introduce large amount of noises. The DMMs of these actions are hard to distinguish. In Fig. 6c, the only two confused actions are tennis swing (17) and tennis serve (18); they have a high similarity and are hard to distinguish as well.

5.4 UTD-MHAD dataset

This dataset [14] consists of four temporally synchronised data modalities. These modalities include RGB videos, depth videos, skeleton positions, and inertial signals from a Kinect camera and a wearable inertial sensor for a comprehensive set of 27 human actions performed by 8 subjects (4 females and 4 males). We only use depth videos here. Each subject repeated each action four times. After removing three corrupted sequences, the dataset includes 861 data sequences. The 27 actions are listed in Table 5. This dataset is larger than the other three datasets mentioned above and has a relatively bigger size in terms of action number and sample number.

In the experiments conducted, the data performed by the subjects 1, 3, 5, 7 were used for training, and the data performed by the subjects 2, 4, 6, 8 were used for testing, which are the same as [14]. We set the DIV = 3 and $\alpha = 0.05$. As a newly published dataset, there are fewer works on UTD-MHAD dataset comparing with the other three datasets. We compare our method with [22] on depth videos, details are shown in Table 6. Although our method is much better than [22], there is enormous exploration potential in this dataset.

5.5 Discussion

The influence of the VB factor $\alpha$ is presented in Fig. 7. As we can see in Fig. 7a, the variation of parameter $\alpha$ has a little effect (<5%) on the classification performance. Our method achieves the
best performance when $\alpha = 0.8$. In Fig. 7b, the influence of the parameter $\alpha$ on Gesture 3D dataset is relatively stable and reach the best performance when $\alpha = 0.2$. On Action Pair and UTD-MHAD dataset the accuracy has a downturn with the increase of $\alpha$ and obtain the best result when $\alpha = 0.05$. This experiment shows that the VB factor $\alpha$ can help to improve the recognition performance comparing with the results when $\alpha = 0$. Obviously, optimal $\alpha$ changes on different actions and different datasets, so how to choose the optimal $\alpha$ is a valuable and challenging question in the future work.

We also compare the recognition rates using different classifiers. The results are shown in Fig. 8. The $l_2$-regularised CR classifier (L2-CRC) is introduced in [22] and has been proved to perform better than $l_2$-CRC [24] and SVM [23]. As exhibited in this figure, in our experiment, the R-ProCRC classifier presents stronger discrimination than the other three classifiers on all the three subsets on Action3D dataset.

To demonstrate how much performance improvement is attributed to the proposed new descriptor, VBDDM using CRC is compared with DMM using CRC [22] for all datasets in Fig. 9. The proposed descriptor VBDDM has about 45% improvement on Action Pair dataset and 3–15% on the other three datasets.

6 Conclusion

In this paper, a computationally efficient DMM-based human action recognition method using R-ProCRC was proposed. We divide a whole video sequence into several sub-sequences and DMMs on three orthogonal Cartesian planes, which are calculated from VBs is used to generate VBDDM. In addition, the utilization of R-ProCRC was shown to be better than $l_2$-CRC which has been proved to perform better than $l_1$-CRC and SVM. The recognition results on MSR Action3D, MSR Action Pairs, MSR Gestures 3D and UTD-MHAD datasets indicate superior performance of our method over most existing methods.

7 Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (61632030, 61201429), the China Postdoctoral Science Foundation (2015M581720,2016M600360), the Jiangsu Postdoctoral Science Foundation (1601216C), and the Technology Research Project of the Ministry of Public Security of China (2014JYSJBJ007).

8 References

[1] Bian W., Tao D., Rui Y.: ‘Cross-domain human action recognition’, IEEE Trans. Syst. Man Cybern. B, Cybern. A, Publ. IEEE Syst. Man Cybern. Soc., 2012, 42, (2), pp. 298–307
[2] Niebles J.C., Wang H., Fei-Fei L.: ‘Unsupervised learning of human action categories using spatial–temporal words’, Int. J. Comput. Vis., 2008, 79, (3), pp. 299–318
[3] Laptev I.: ‘On space-time interest points’, Int. J. Comput. Vis., 2005, 64, (2), pp. 107–123
[4] Jiang M., Kong J., Bebis G., et al.: ‘Informative joints based human action recognition using skeleton contexts’, Image Commun., 2015, 33, (C), pp. 29–40
[5] Xia L., Chen C.C., Aggarwal J.K.: ‘View invariant human action recognition using histograms of 3D joints’, IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, June 2012, pp. 20–27
[6] Li W., Zhang Z., Liu Z.: ‘Action recognition based on a bag of 3D points’, IEEE Conf. Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, June 2010, pp. 9–14
[7] Yang X., Zhang C., Tian Y.L.: ‘Recognizing actions using depth motion maps-based histograms of oriented gradients’, ACM Int. Conf. on Multimedia, 2012, pp. 1057–1060
[8] Xu H., Chen E., Liang C., et al.: ‘Spatiotemporal pyramid model based on depth maps for action recognition’. MmSp2015 IEEE Int. Workshop on Multimedia Signal Processing, 2015
[9] Liang B., Zheng L.: ‘3D motion trail model based pyramid histograms of oriented gradient for action recognition’, Int. Conf. on Pattern Recognition, 2014, pp. 1952–1957
[10] Yang W., Wang Z., Sun C.: ‘A collaborative representation based projections method for feature extraction’, Pattern Recognit., 2015, 48, (1), pp. 20–27
[11] Wang J., Liu Z., Wu Y., et al.: ‘Mining actionlet ensemble for action recognition with depth cameras’, IEEE Conf. on Computer Vision and Pattern Recognition, 2012, pp. 1290–1297
[12] Oreifej O., Liu Z.: ‘HON4D: histogram of oriented 4D normals for activity recognition from depth sequences’, IEEE Conf. on Computer Vision and Pattern Recognition, 2013, vol. 9, no. 4, pp. 716–723
[13] Wang J., Liu Z., Chorowski J., et al.: ‘Robust 3D action recognition with random occupancy patterns’, European Conf. Computer Vision (ECCV), Florence, Italy, October 2012
[14] Chen C., Jafari R., Kehtarnavaz N.: ‘UTD-MHAD: a multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor’. IEEE Int. Conf. on Image Processing, 2015
[15] Schultd C., Laptev I., Caputo B.: ‘Recognizing human actions: a local SVM approach’. Int. Conf. on Pattern Recognition, 2004, vol. 3, pp. 32–36
[16] Sun J., Wu X., Yan S., et al.: ‘Hierarchical spatiotemporal context modeling for action recognition’, IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, June 2009, pp. 2004–2011
[17] Bobick A.F., Davis J.W.: ‘The recognition of human movement using temporal templates’, IEEE Trans. Pattern Anal. Mach. Intell., 2001, 23, (3), pp. 257–267
[18] Luo J., Wang W., Qi H.: ‘Group sparsity and geometry constrained dictionary learning for action recognition from depth maps’, IEEE International Conf. Computer Vision (ICCV), Sydney, Australia, December 2013, pp. 1809–1816
[19] Chen C., Jafari R., Kehtarnavaz N.: ‘Action recognition from depth sequences using depth motion maps-based local binary patterns’. IEEE Winter Conf. on Applications of Computer Vision, 2015, pp. 1092–1099
[20] Yang X., Tian Y.L.: ‘Super normal vector for activity recognition using depth sequences’, IEEE Conf. on Computer Vision and Pattern Recognition, 2014, pp. 804–811
[21] Chen C., Liu M.: ‘3D action recognition using multi-temporal depth motion maps and Fisher vector’. Int. Joint Conf. on Artificial Intelligence, 2016
[22] Chen C., Liu K., Kehtarnavaz N.: ‘Real-time human action recognition based on depth motion maps’, J. Real-Time Image Process., 2013, 12, (1), pp. 155–163
[23] Chen P.H., Lin C.J., Scholkopf B.: ‘A tutorial on v-support vector machines’, Appl. Stoch. Models Bus. Ind., 2005, 21, (2), pp. 111–136
[24] Zhang L., Yang M., Feng X.: ‘Group sparsity and geometry constrained representation: Which helps face recognition?’, Int. Conf. on Computer Vision, IEEE Computer Society, 2011, pp. 471–478
[25] Vieira A.W., Nascimento E.R., Oliveira G.L., et al.: ‘STOP: space-time occupancy patterns for 3D action recognition from depth map sequences’, Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, Buenos Aires, Argentina, September 2012, pp. 252–259