IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 41, No. 3, June, 2024, pp. 1461-1472
Journal homepage: http://iieta.org/journals/ts

Enhanced Depth Motion Maps for Improved Human Action Recognition from Depth Action Sequences

Dustakar Surendra Rao[1,2], L. Koteswara Rao[1*], Vipparthi Bhagyaraju[3], Goh Kam Meng[4]

[1] Department of ECE, Koneru Lakshmaiah Education Foundation, Hyderabad 500075, India
[2] Department of ECE, Guru Nanak Institutions Technical Campus, Hyderabad 501506, India
[3] Department of ECE, Siddhartha Institute of Engineering and Technology, Hyderabad 501506, India
[4] Centre for Multimodal Signal Processing, Faculty of Engineering and Technology, Tunku Abdul Rahman University of Management and Technology, Kuala Lumpur 53300, Malaysia

Corresponding Author Email: koteswararao@klh.edu.in

https://doi.org/10.18280/ts.410334

Received: 24 May 2023
Revised: 29 September 2023
Accepted: 16 February 2024
Available online: 26 June 2024

Keywords:
human action recognition, depth action sequences, deep learning, convolutional neural networks, depth maps, depth motion maps and spatial window

ABSTRACT

Human action recognition based on depth action sequences is a well-known study that can be used in many fields. Compared to RGB Videos, depth action videos are more robust as they are not affected by changes in lighting. This study proposes the Enhanced Depth Motion Map (EDMM), a new action descriptor to overcome the challenges of the conventional DMM, which cannot handle the presence of some undefined regions in depth maps. Contrary to DMM's global motion description, the EDMM meticulously scans individual pixels and then accurately identifies those in motion. We extracted the EDMM from a series of video sequences and then used a convolutional neural network (CNN) model to simplify the motions accurately. The CNN model, equipped with nine layers, accurately recognizes activities based on maximum movement similarity. The method underwent testing using two standard and publicly available datasets; MSR Action 3D and UTD-MHAD. The test results through True Positive Rate (TPR), Positive Predictive Value (PPV) or Precision, False Discovery Rate (FDR), False Negative Rate (FNR), F1-score, and accuracy demonstrated the superiority of the proposed method over numerous state-of-the-art methods like DMM, DMM with Local Binary Pattern and DMM with Histogram of Oriented Gradients (HOGs).

1. INTRODUCTION

Human Action Recognition (HAR) has recently gained a lot of research attention due to its integrated nature in numerous applications such as human-computer interface (HCI) [1], motion analysis, intelligent monitoring [2], virtual reality, and some computer vision-related applications like intelligence surveillance [3-5] and content-based video retrieval. Applying HAR enables a better understanding of people's actions and habits through video monitoring and pattern observation. Human Action Recognition (HAR) employs diverse data modalities, such as RGB, infrared, and depth, to detect, localize, and recognize activities. The primary goal of action recognition involves categorizing a data sequence or video into predefined classes by extracting representative features that describe the characteristics of actions across multiple data modalities. After the feature extraction, the trained model is employed to recognize irrespective of the subjects (same person or different person).

Previously, most studies on HAR have concentrated on RGB videos acquired by standard cameras. Yet, these image sequences are often affected by environmental changes, shadows, and variations in illumination. Consequently, the introduction of depth-video-based HAR aims to mitigate the influence of lighting, shadows, color variations, and other environmental factors. Notably, a depth-based camera possesses the capability to capture high-resolution videos even under extremely low illumination conditions [6-8].

Moreover, the color and texture variations are less impactful in detecting moving objects and humans from clustered backgrounds in HAR. Further, the traditional RGB videos can't provide any information about the motion cues. Additionally, depth cameras can provide 3D structural information about objects in the scene. RGBD-HuDaAct is one of the datasets having 3D structural information [9]. Due to recent technological advances, a particular type of camera (Microsoft Kinect) is now available to capture depth videos [10].

Multiple HAR models have been specifically crafted around depth action videos [11-13] owing to their array of advantages. Among these models, the Depth Motion Map (DMM) stands out as a widely utilized and straightforward method for depicting activity within 2D spatial images. Its computation is notably simple, revolving around the disparity between identical pixels in consecutive frames. Nonetheless, the current iteration of DMM has got its own set of limitations.

1. In some depth action videos, certain regions are undefined [14], and these areas don't correspond to any actual motion in the video. However, they still manifest in the Depth Motion Map (DMM). Due to the presence of unnecessary

movements, the DMM considers them also as motion-related regions. In addition to that, the body shaking movements and Ghost Shadows appeared.

2. Due to low-quality video, depth action videos consist of noises that might simulate artificial motions. For instance, unnecessary actions and clothing may result in unnatural and wrongly moving pixels in the DMM [15-17].

In response to these challenges, we introduce these groundbreaking recognition systems designed to master these obstacles, emphasizing its key contributions as follows:

1. The EDMM is proposed to eliminate unclear regions and identify fake moving signals.

2. Furthermore, we introduce a sophisticated grouping model based on deep learning which aims to derive discriminative and efficient features for enhanced and productive classification.

In the previous work, EDMM has been applied to gesture recognition [18]. Gestures consider only a few parts of the body's facial expression, fingers, legs, etc. However, action recognition considers whole body parts. To our exhaustive search, no previous literature uses EDMM for HAR. As a result, we contributed the very first EDMM-based approach and deep learning for action recognition.

The paper is structured as follows: Section 2 delves into previous depth data-based methods for action recognition. Section 3 offers the specifics of the proposed action recognition system. Section 4 examines the experimental results, and Section 5 wraps up with concluding remarks.

2. LITERATURE SURVEY

The recognition of depth data-based actions has been handled in various ways over the years [14, 18]. Hence in this section, we briefly go through several existing methods to explain the trends and the technological changes in HAR. Figure 1 depicts a depth action (High Wave) video with a total of 54 depth frames that have only movements of action.

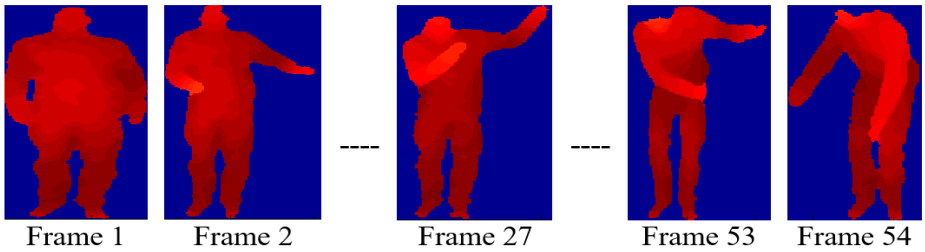

Figure 1. Sample frames of a depth action video of Golf Swing Action

2.1 DMM-based approaches

Chen et al. [15] derived three views of depth action video by projecting it onto three orthogonal planes: top, side, and front. Then they measured the DMM by accumulating global action. Over the obtained DMMs, they applied the Histogram of Oriented Gradients (HOGs); hence, the descriptor is called DMM-HOG. Each plane used its HOGs-HOGf, HOGs, and HOGt.

A Linear SVM is thus employed to classify the actions based on their HOG features. An accuracy of approximately 85% for images was obtained with frontal views. However, HOG features are sensitive to image rotations making them not a good choice for classifying actions that often appear from different angles.

On the other hand, Chen et al. [16] considered only two views: the front and the side. From these views, two action descriptors viz. "Depth Motion History (DMH)" and "Depth Motion Appearance (DMA)" are extracted. Descriptors are fed into the SVM, which then classifies these descriptors. Similarly, Chen et al. [17] also rated DMM from three different planes. The absolute difference between successive frames is accumulated in each projected plane for each view. They employed an L2-regularized classifier with a collaborative representation of the distance-weighted Tikhonov Matrix for classification. However, despite its utility, DMM encounters inefficiencies when handling small body-shaking movements and shadows as moving features.

Beddiar et al. [18] enhanced their earlier version in a segmented DMM computation. Initially, they applied segmentation over an entire video sequence and partitioned it into several overlapping segments. Further, they measured each element and produced one DMM for each aspect. They then applied Local Binary Patterns (LBPs) on DMMs to explore the texture information of action. At last, they employed Fisher Kernel (FK) for encoding the LBP descriptor and then passed it to "Kernel-Based Extreme Learning Machine (KELM)" to get the action class label.

Kim et al. [19] proposed a further enhancement by contributing at the fusion level. They applied two different fusion scenarios: decision-based fusion and feature-based fusion. They fused the decisions obtained at the former's output while combining the LBP features for feature-level fusion. At the fusion of judgment, they used the softmax rule over the obtained probability scores of each action. However, the segment size must be adaptive because each action video has its length. The standard segment size is not an appropriate solution for an effective HAR. Moreover, they also used simple DMM, which is not robust for external effects.

Al-Faris et al. [20] applied fuzzy logic over the segmented DMMs to determine each segment's motion's significance. They applied a weight function in three directions-central, reverse, and linear. This approach involved designing a novel CNN model rooted in deep learning for classification.

Recognizing the time-dependency of motion significance, they highlighted that initial frames might lack substantial motion details at the action's outset. However, as time advances, movement intensifies. They emphasized that aligning segmentation with time could yield a more effective solution.

2.2 4D approaches

Vieira et al. [21] developed a "Space-Time Occupancy Pattern (STOP)" named descriptor with a 4D grid after the segmentation of action video through the space-time axis.

STOP effectively preserves spatial and temporal information, making the system robust to intra-class variations.

Wang et al. [22] introduced another 4D-based action descriptor named 'Random Occupancy Patterns (ROP),' employing sparse coding techniques to address noise and occlusion challenges. ROP utilizes data sampling methods to effectively explore a broader sampling space and encodes features using sparse coding.

Both these 4D descriptors proficiently manage issues commonly found in action videos, such as occlusions and noise, without the need for additional parameter tuning.

2.3 STIP-based approaches

Following the massive success of "Spatio-Temporal Interest Points (STIPs)" in recognizing human actions from RGB videos, Xia and Aggarwal [23] introduced an extended version of STIPs called "Depth STIPs (DSTIPs)" to detect the interest points from depth action videos. However, DSTIPs are more susceptible to noise. Additionally, they also proposed a new feature named "Depth Cuboid Similarity Feature (DCSF)," which explores the local 3D cuboid depth surrounding the DSTIP through a correct size. Every action is characterized through a bag of words (BoW) set, and it constructs a codebook after clustering all the DCSFs through the most popular K-means clustering algorithm. Every codeword is determined with the cluster centre, and every feature vector is assigned to a code word with the help of Euclidean distance.

2.4 Other approaches

Li et al. [24] applied CNNs to represent action videos by using DMMs on three orthogonal planes. They adapted for Multi-view CNN (MV-CNN) composed of three CNN distinct architectures each for one view. With each Channel, the fully connected layer generates a complete set of action scores further applied to the softmax regression layer to predict the score of action present in the given input action video.

By extending the LBP, Xia and Aggarwal [23] suggested a new variant, namely "Local Ternary Pattern (LTP)." They initially project action videos onto three orthogonal planes and represent each view with DMM. Then LTP is applied to each DMM to effectively differentiate the actions with similar movements. Finally, they adopted for CNN model for classification.

Arivazhagan et al. [25] aimed to classify human action by combining the salient features from both Depth and RGB cameras. They generated a Salient Information Map from both RGB and Depth action videos, sign positioning the significant motion region of the video. They extracted sign, magnitude, and centre descriptors from the map representing the complete LBP. Sargin et al. [26] consolidated these features, they utilized canonical correlation analysis for dimensionality reduction and subsequently fed them into a Multiclass SVM algorithm for classification.

In summary, current methods are very vulnerable to external factors such as a) background noises caused by body shaking movements and b) Ghost Shadows. Their poor recognition performance in noisy environments stems from the lack of pixel-level analysis to differentiate between motion and non-motion pixels. While some techniques characterized each pixel as a 4D vector, they struggled to accurately distinguish between real, genuine, and false fake motions.

3. PROPOSED METHOD

3.1 Overview of the method

This section outlines the particulars of our proposed HAR mechanism. Under this mechanism, we introduce a new action descriptor called Enhanced DMM (EDMM), an extended version of DMM. EDMM is more effective in representing the action under several real-time constraints. Our developed HAR system initially means it in a 2D spatial representation image using DMM and is fed to a deep learning model. Additionally, we propose another deep learning model tailored to effectively recognize actions with numerous repeatable elements. The result is maximum probability through the softmax regression layer. The overall block schematic of the developed HAR system is demonstrated in Figure 2.

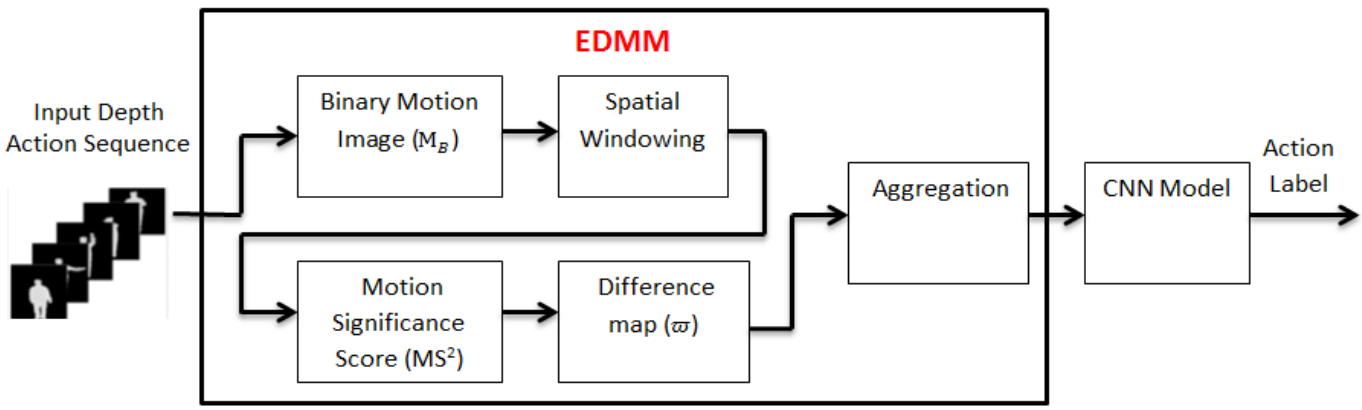

Figure 2. Overall block schematic of Human action recognition system

3.2 Enhanced DMM

In the context of action recognition from videos, the action representation holds crucial importance which describes the key features of action through its movements Human Action Recognition initially represents the action through the motion features that encode the motion information. Among different methods of motion representation, a depth motion map is one of the most popular methods. DMM was initially introduced by Chen et al. [15] and is derived from the motion energy accumulation of an entire video sequence. Chen et al. [17] also developed DMM by calculating motion energy measured as a

mean summation of differences between consecutive frames. The main difference between these two methodologies is the motion energy calculation by thresholding the difference [15]. Here, we adapted the DMM proposed by Chen et al. [17]. For an input depth action video with N number of frames, let $I=\{I_1, I_2, ...I_N\}$, the DMM is computed as in Eq. (1):

$$DMM = \sum_{t=0}^{N-2} |I(i,j,t) - I(i,j,t-1)| \quad (1)$$

where, $I(i,j,t)$ and $I(i,j,t-1)$ represent the intensities of pixels of the frames at time instances *t* and *t-1*, respectively; the value of *t* varies from 0 to *N-2*. DMM is highly effective in detailing the shape and motion cues of the input action video by producing a 2D-spatial energy distribution map that helps in distinguishing various actions. However, most of the depth action videos are acquired under specific environments such as noisy, unstable reflections from depth camera, etc. Such environments produce videos with some undefined regions after subjecting them to DMM. Some areas intermittently appear in a few frames, like shadows surrounding object boundaries with undefined depth pixel intensities—instances unique to specific frames. Additionally, small body shaking movements between successive frames introduce erroneous edges into the DMM.

These false edges are sensitive to the body's size and don't deliver much helpful evidence regarding action motion. Several image processing methods like median filtering and mathematical morphology are proposed to overcome these problems [27]. These methods may result in DMM missing genuine motion information. Thus, we proposed a new variant of DMM called Enhanced DMM, which measures the motion weights at each pixel in DMM. The proposed EDMM relies on a D spatial windowing centered at each pixel location. Before processing them for spatial windowing, a binary image is generated from every two successive frames based on pixel intensities. Consider $I(i,j,t)$ and $I(i,j,t+1)$ as the intensities of pixels of the frames at time instances *t* and *t+1*, respectively; the binary image is constructed using Eq. (2):

$$\mathrm{M}_B(i,j,t) = \begin{cases} 1, & if\ I(i,j,t) \neq I(i,j,t+1) \\ 0, & Otherwise \end{cases} \quad (2)$$

Referring to the expression shown in Eq. (2), for a given action video with N number of frames, we derive a total N-1 binary images. Subsequently, these binary images are subjected to spatial windowing by locating each pixel (i,j,t) at the center to decide whether this pixel belongs to a fake moving pixel or a real motion pixel. In general, the fake moving pixels are located alone in the region. For a natural motion, the human body generates the pixels with non-zero values for a larger area in the binary image M_B. In other words, as much as the motion at a pixel (i,j,t) it is considered necessary as the number of pixels with appropriate action is more significant in the 2D spatial window. To identify such significance (weight), we compute a Motion Significance Score (MSS or MS^2) for each pixel in the binary image M_B as in Eq. (3):

$$\mathrm{M}_S(i,j,t) = \frac{1}{(h+1)(w+1)} \sum_{x=i-(h/2)}^{i+(h/2)} \sum_{y=j-(w/2)}^{j+(w/2)} \mathrm{M}_B(i,j,t) \quad (3)$$

where, w and h are the weight and height of the spatial window, respectively. The major motivation behind applying the spatial windowing is to analyze each pixel efficiently with respect to its neighboring pixels. The values of $\mathrm{M}_S(i,j,t)$ range in binary form, where 0 indicates the fewer MS^2 to reflect fake moving pixel and 1 shows high MS^2, which reflects the nature of real moving pixel.

Figure 3 represents an illustration depicting noise-removed motion regions between successive frames in a depth action sequence. In our experiments, we fixed the width and height of the spatial window as 7×7. As 5×5 and 3×3 are smaller, the differentiation between motion and noisy pixels becomes tough. To provide perfect discrimination between real and fake motion pixels, we set a threshold (ψ) for MS^2. The MS^2 value less than ψ will be considered fake moving pixels and vice versa. In our experiments, we fix the threshold value as 0.6. The major reason is that the proposed system can discriminate the actions with minor movements. From Eq. (3), the range of Motion significance is derived as [0 1], where 0 denotes the most negligible relevance, and 1 denotes the maximum importance. For the lower values of 0.6, the required motion information-related pixels are also discarded; hence, we set it for 0.6. Next, based on the MS^2 and threshold, a new and intermediate map is measured as in Eq. (4):

$$\varpi(i,j,t) = \begin{cases} |I(i,j,t) - I(i,j,t-1)|, & if\ \mathrm{M}_S(i,j,t) > \psi \\ 0, & Otherwise \end{cases} \quad (4)$$

The above expression indicates that the difference map (ϖ) is generated as a difference between the same pixels in successive frames if the MS^2 of a pixel is greater than the threshold. Once the difference map is computed for all consecutive images resulting maps are accumulated to get the final EDMM, as described in Eq. (5):

$$EDMM = \sum_{t=0}^{N-2} \varpi(i,j,t) \quad (5)$$

Figure 4 (a) shows an example of input depth action frame, Figure 4 (b) shows the result of DMM and Figure 4 (c) shows the result of proposed EDMM. From Figure 4 (c) we can observe a clear spatial energy distribution map with no fake moving pixels or narrow edge boundaries.

3.3 CNN model

The EDMM described in Eq. (5) is then rescaled to 112×112 and fed as an input to the CNN model effectively representing the motion within the action footage. Our proposed CNN model consists of five Conv layers, two PL layers, and a FCL layer to establish complete connectivity. The Conv layers have extracted features while pooling layers decrease the dimensionality of the components. Our use of depth information results in maximum or minimum pixel relations.

As a result, max pooling is the pooling technique most suited to dimensionality reduction. When it comes to the number of activities, we propose a CNN model with a single fully linked layer of size $1 \times n$. This new CNN model's structure is depicted in Figure 5. This CNN model is a customized CNN model and it was inspired from AlexNet architecture. Due to this reason, we followed same number of layers and almost similar filter sizes.

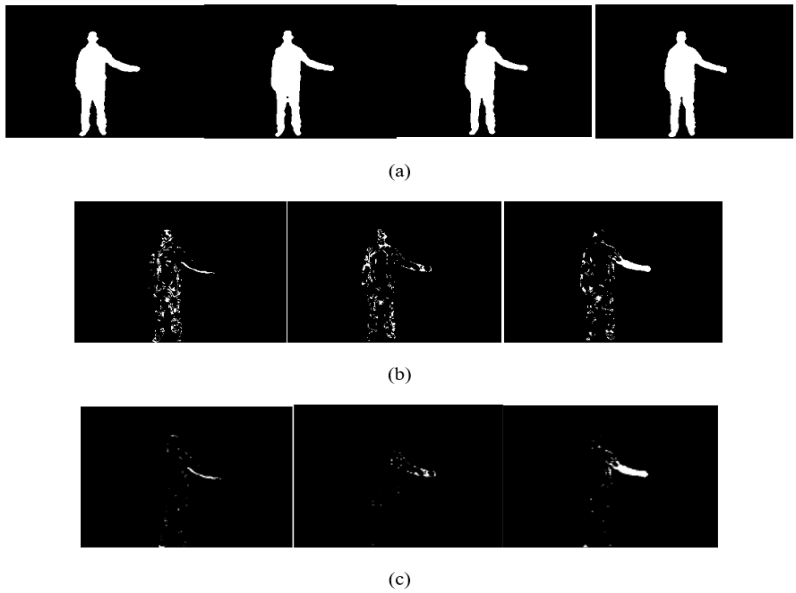

Figure 3. (a) Depth sequence, (b) Motion regions between successive frames, and (c) Motion regions between successive frames after removal of noise

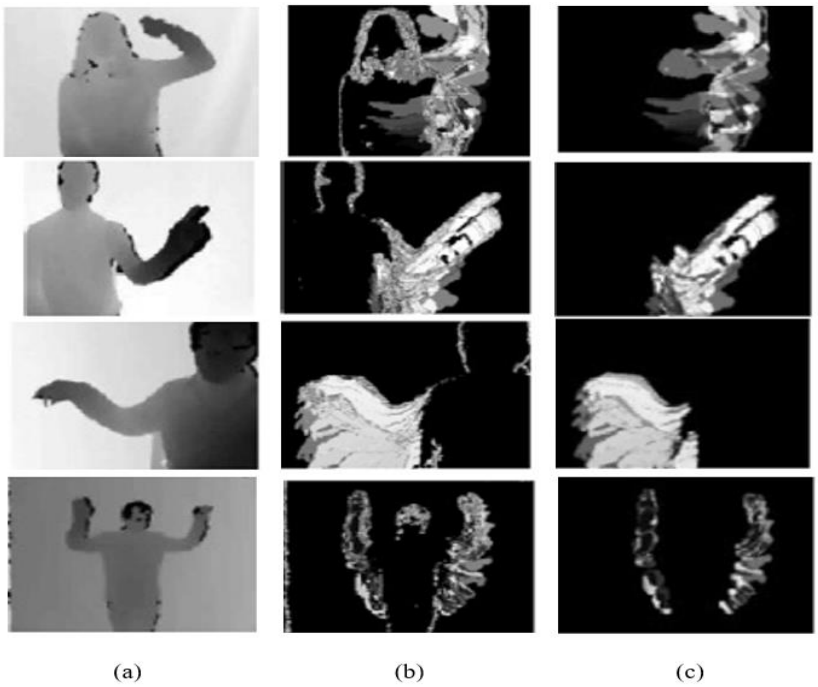

Figure 4. (a) Depth Frame, (b) DMM, and (c) EDMM

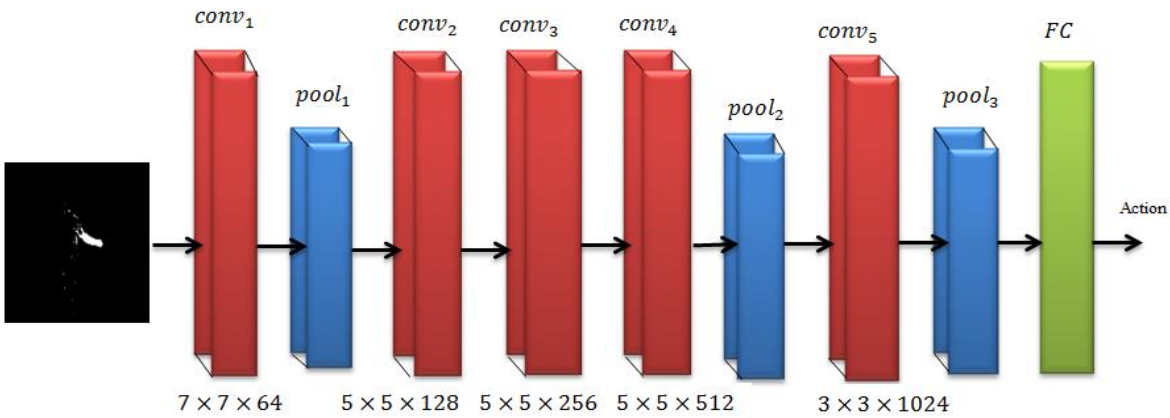

Figure 5. CNN model

Table 1. CNN model attributes

Layer	Filter Count	Size	Filter Size	Stride
$Conv_1$	64	112×112	7×7	2×2
$Pool_1$	-	-	-	2×2
$Conv_2$	128	56×56	5×5	1×1
$Conv_3$	256	56×56	5×5	1×1
$Conv_4$	512	56×56	5×5	1×1
$Pool_2$	-	-	-	2×2
$Conv_5$	1024	28×28	3×3	2×2
$Pool_3$	-			2×2

Table 1 indicates the specifications for each convolutional layer in the architecture, Conv1 has applied 64 convolutional filters, each sized 7×7. Subsequently, Conv2, Conv3, and Conv4 utilize 128, 256, and 512 filters respectively, each with a 5×5 dimension per filter. At Conv5, each convolutional filter is 3×3 in size, totaling 1024 filters. This higher number of filters in Conv5 is attributed to their smaller size [27]. When the size of convolutional filters is influenced by two distinct actions through EDMM, extracting unique features becomes difficult for the system. For example, a 3×3 filter size applied to the image at the outset is inefficient, as the characteristics will be shared by two action images within the small-sized region. Therefore, 7×7 convolutional filters have been decided to be implemented in the initial convolutional layer. The filter size at the max-pooling layer is thus limited to 2×2, with the primary goal of reducing the feature map size. In this study, we employed two max-pooling layers, with the first after Conv1 and the second after Conv4. By implementing the max-pooling layer following Conv1 and Conv4, the feature map size is decreased from 112×112 to 56×56 and from 56×56 to 28×28, respectively. The feature maps are then processed by the fully connected layer (FCL), and their scale corresponds to the actual actions being tested. We produced each action score during the testing phase using the trained weights and the softmax regression layer. The action with the highest score is considered present in the video input.

4. SIMULATION EXPERIMENTS

This section explores the particulars of experimental analysis. Conducted using MATLAB software. First, we examine the details of the datasets. Next, the details of performance metrics are explored, and finally, the observed results are demonstrated.

4.1 Datasets

In our experiments, we validated our approach using two well-known benchmark datasets: MSR Action 3D and UTD-MHAD dataset. This section provides comprehensive details about these datasets, including total number of actions, actions acquired environments, the total number of subjects involved in the creation, video formats, etc.

4.1.1 MSR action 3D dataset [28]

The actions in MSRA3D are captured with the help of a depth camera, and the subject's pose is located in the front view. This dataset is acquired with the help of ten issues, and every subject performs each action two to three times. This dataset is very puzzling due to the speed variations on each topic. Also, this dataset consists of 20 steps: bend, Waving TWO HANDS, Handclap, Jog, kicking Sides, Kicking Forward, Pickup & Throw, Golf swing, Tennis Swing, High Arm wave, Horizontal Arm wave, Forward punch, High Throw, Hammer, Hand catch, Draw cross, Draw tick, Draw a circle, and Side boxing. Some actions of MSR action 3D are depicted in the following Figure 6.

4.1.2 UTD-MHAD dataset [29]

The authors used a wearable Microsoft Kinect Sensor to acquire this dataset, and all the actions are accepted in the indoor environment. Eight subjects were used for the activities; four were female, and the remaining four were male. Each subject performed each step four times; thus, the total number of action videos is 864. Since the three movements are corrupted, the total number of moves is 861. The total number of actions available in this dataset is 27; including Bowling, Drawing a triangle, Swiping Right, clapping HANDS, Swiping Left, Waving, crossing arms, drawing a Cross, Throw, drawing a Circle Counterclockwise, Basketball Shoot, drawing Circle Clockwise, Baseball Swing, Front Boxing, Squat, Arm Curl, Tennis Swing, Push, Tennis Serve, Catch, Knock, Jogging, Pick up & Throw, Sit to Stand, Walking, Stand to Sit and Lunge.

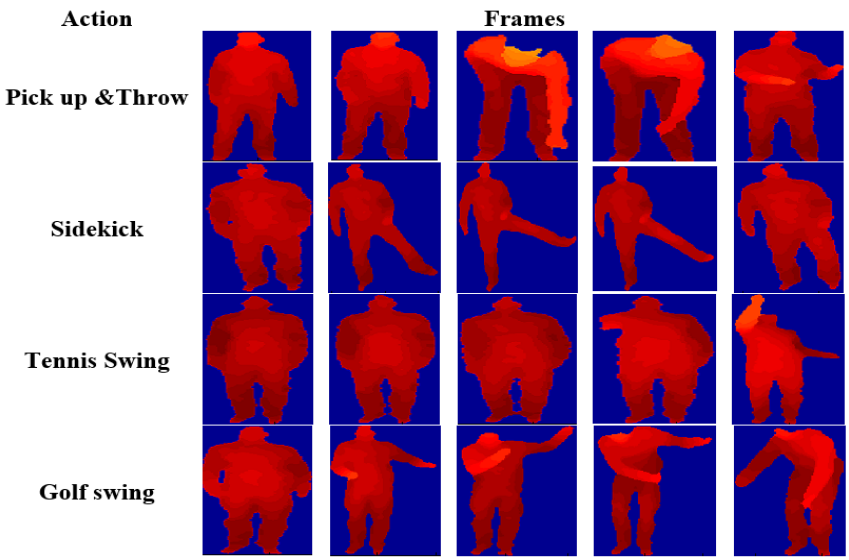

Figure 6. Several actions of MSRA3D

4.2 Performance metrics

For the performance evaluation of the developed action recognition model, we adapt for Receiver Operating Characteristics (ROC). Under ROC evaluation, a confusion matrix is created based on the recognized actions individually. For every step, there exist four measures, namely "True Positives (TPs), False Positives (FPs), True Negatives (TNs), and False Negatives (FNs)." When an action is correctly recognized, it is counted under TP; otherwise, FP or FN. Under these four measures, the following metrics are calculated: namely, Recall or True Positive Rate (TPR), Positive Predictive Value (PPV) or Precision, False Discovery Rate (FDR), False Negative Rate (FNR), F1-score, and accuracy. For all these metrics, there exists a standard mathematical formula as follows:

$$TPR = \frac{TP}{TP + FN} \tag{6}$$

$$PPV = \frac{TP}{TP + FP} \tag{7}$$

$$F1 - Score = \frac{2 \times TPR \times PPV}{TPR + PPV} \tag{8}$$

$$FNR = \frac{FN}{TP + FN} \tag{9}$$

$$FDR = \frac{FP}{TP + FP} \tag{10}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

4.3 Results

In our results evaluation we repeatedly apply our proposed methods to each of the datasets. Each time the subjects are altered, and the performance is measured through the performance mentioned earlier metrics. For example, the MSR action 3D dataset is acquired with the help of 10 subjects, and each subject performed every action two to three times. If the first five subjects are used for training in the first validation, the following five topics are used for testing. Similarly, we consider even issues for training and odd subjects for testing in the second validation. In this manner, we conduct fivefold cross-validation on each dataset, averaging the obtained values for every action. In this section, initially, we explore the results obtained over MSR action3D and then the results of UTD-MHAD.

4.3.1 Results of MSR action 3D

After the simulation of different actions of MSRA3D, we measure the above-specified performance metrics, and the results are stipulated in Table 2. The values demonstrated here are average values of five-fold cross-validation.

Table 2. Performance analysis of the proposed approach on MSRA3D

Action/Metric	TPR (%)	PPV (%)	F1-Score (%)	FNR (%)	FDR (%)
Hammer	68.7865	50.9347	58.5296	31.2135	49.0653
Horizontal Arm Wave	**93.5666**	58.4374	71.9426	**6.4334**	41.5626
High Arm Wave	81.1665	84.1466	82.6296	18.8333	15.8534
Side-Boxing	79.4602	79.0608	79.2599	20.5398	20.9392
Two-Hand Wave	78.7674	**99.3711**	87.8777	21.2326	**0.62890**
Bend	91.4621	86.2810	88.7960	8.53789	13.7190
Hand Clap	76.3669	92.1700	83.5275	23.6331	7.83000
Forward Kick	83.4246	64.1694	72.5409	16.5754	35.8306
Draw Circle	64.4554	66.0377	65.2369	35.5446	33.9623
Side Kick	76.1977	98.3722	85.8766	23.8023	1.62779
Draw Tick	65.7647	65.5115	65.6378	34.2353	34.4885
Jogging	92.4266	97.2922	**94.7970**	7.57340	2.70780
Draw Cross	66.4256	53.1602	59.0571	33.5744	46.8398
Tennis Swing	67.4346	82.4561	74.1926	32.5654	17.5439
High Throw	81.4545	59.3691	68.6799	18.5455	40.6309
Tennis Serve	66.1007	76.2733	70.8235	33.8993	23.7267
Forward Punch	90.4588	58.4831	71.0385	9.54120	41.5169
Golf Swing	70.0455	77.3672	73.5245	29.9545	22.6328
Hand Catch	62.1485	58.1498	60.0826	37.8515	41.8502
Pick Up & Throw	68.2720	84.3266	75.4547	31.7280	15.6734

The above-specified results observed the maximum and minimum recall for the Horizontal arm wave and Hand catch classes. Along with hand catch action, the three actions namely Draw tick, Draw a circle, and Draw cross, are also observed to have a minimum recall. Since these three actions have similar movements in the entire body except for fingers, they are misclassified and result in more FPs and FNs. Hence, these few classes are observed to have almost similar recall rates. This effect can be observed at higher values of FNR as they are 35.5446%, 34.2353%, and 33.5744% for Draw circle, Draw tick, and Draw cross, respectively. Next, the maximum and minimum PPV are observed at Two-Hand Wave and Hammer. The sidekick action is the only action in MSRA3D that deviates from other activities. Hence, it has gained more precision and less FDR, observed as 99.3711% and 0.6289%, respectively. Furthermore, the F1-score is computed as a harmonic mean of recall and precision. The highest value of the F1-score is 100, which indicates the perfect memory and accuracy, while the lowest value is 0, which shows either recall or precision 0. From the above table, the maximum F1-score (94.7970%) is observed for jogging action, and the minimum (58.5296%) is observed for Hammer action. In summary, the simulation experiments over MSRAction 3D declare that the better recognition performance is obtained at Jogging action while the minimum recognition performance is obtained at Hammer action. In the MSRAction 3D dataset, the

Hammer action is in resemblance with several actions like High Arm Wave, Pick up & Throw etc., the misclassification rate is high. On the other side, even though the jogging action is in resemblance with Running and Walking, the proposed approach can provide sufficient discrimination between due to the temporal analysis.

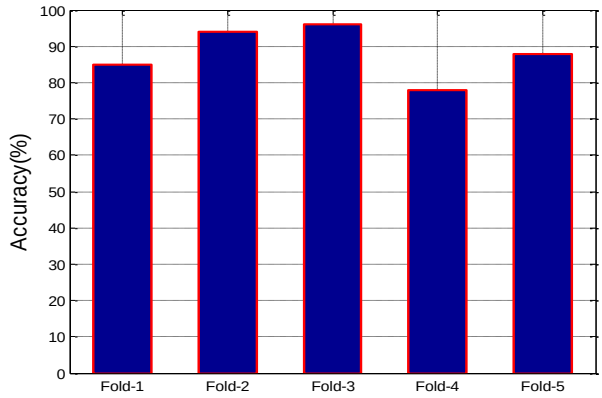

Figure 7. Accuracies at different cross-validations- MSR Action 3D

Figure 7 shows an average accuracy at different cross-validations. The subjects are changed at every validation, and accuracy is measured after testing the 20 action videos. Among these five validations, the maximum accuracy (96.0212%) is achieved at the third validation and minimum accuracy (78.0451%) at the fourth validation. Since the subjects used for training and testing differ at every validation, the accuracy at folds varies. It is dependent on the matter used to accomplish the action. From the accuracies obtained at five validations, the mean accuracy with standard deviation is 6.8920%.

Table 3 shows the MSRA3D is an extensive dataset as it contains 20 individual actions. By comparing the similarity and complexity, they are partitioned into three groups, each composed of eight steps. Among the 20 actions, the four are grouped into multiple group Pick Up & Throw forward kick, and High throw. According to the actions shown in Tab.3.the, Action Set 1 (AS1) and Action Set 2 (AS2) include similar measures, while Action Set 3 (AS3) consists of some complex steps. The significant advantage of such representation is that the recognition system's ability to recognize complicated efforts increases accurately.

Table 3. Action sets of MSRA3D

AS1	AS2	AS3
Pick Up & Throw ($\boldsymbol{AS_{11}}$)	Two-hand Wave (AS_{21})	Forward Kick (AS_{31})
High Throw ($\boldsymbol{AS_{12}}$)	Forward Kick (AS_{22})	Side Kick (AS_{32})
Hammer ($\boldsymbol{AS_{13}}$)	Draw Tick (AS_{23})	Tennis Swing (AS_{33})
Forward Punch ($\boldsymbol{AS_{14}}$)	High Arm Wave (AS_{24})	Jogging (AS_{34})
Bend ($\boldsymbol{AS_{15}}$)	Draw Cross (AS_{25})	Golf-swing (AS_{35})
Hand Clap ($\boldsymbol{AS_{16}}$)	Draw Circle (AS_{26})	High Throw (AS_{36})
Tennis Serve ($\boldsymbol{AS_{17}}$)	Hand Catch (AS_{27})	Pick-up & Throw (AS_{37})
Horizontal Arm Wave ($\boldsymbol{AS_{18}}$)	Side-boxing (AS_{28})	Tennis Serve (AS_{38})

Table 4. Confusion matrix of EDMM+CNN on MSRA3D-AS1

		$\boldsymbol{AS_{11}}$	$\boldsymbol{AS_{12}}$	$\boldsymbol{AS_{13}}$	$\boldsymbol{AS_{14}}$	$\boldsymbol{AS_{15}}$	$\boldsymbol{AS_{16}}$	$\boldsymbol{AS_{17}}$	$\boldsymbol{AS_{18}}$
TRUE LABEL	AS_{11}	**97.5245**				4.4755			
	AS_{12}		**71.5522**			18.5455	9.9028		
	AS_{13}			**98.8674**				1.1326	
	AS_{14}				**98.3325**				1.6675
	AS_{15}					**83.2411**		16.7589	
	AS_{16}		1.2155				**98.7845**		
	AS_{17}					11.5215		**88.4785**	
	AS_{18}				3.6522				**96.3478**
		PREDICTED LABEL							

Table 5. Confusion matrix of EDMM+CNN on MSRA3D-AS2

		$\boldsymbol{AS_{21}}$	$\boldsymbol{AS_{22}}$	$\boldsymbol{AS_{23}}$	$\boldsymbol{AS_{24}}$	$\boldsymbol{AS_{25}}$	$\boldsymbol{AS_{26}}$	$\boldsymbol{AS_{27}}$	$\boldsymbol{AS_{28}}$
TRUE LABEL	AS_{21}	**90.2335**			9.7665				
	AS_{22}		**87.4215**					12.5785	
	AS_{23}			**98.6698**		1.3302			1.4326
	AS_{24}				**98.5674**				
	AS_{25}			11.8323		**76.3354**	11.8323		
	AS_{26}			2.3655			**97.6345**		
	AS_{27}		22.4766					**77.5234**	
	AS_{28}			3.4522					**96.5478**
		PREDICTED LABEL							

Table 6. Confusion matrix of EDMM+CNN on MSRA3D-AS3

		AS_{31}	AS_{32}	AS_{33}	AS_{34}	AS_{35}	AS_{36}	AS_{37}	AS_{38}
TRUE LABEL	AS_{31}	**100.00**							
	AS_{32}	1.5422	**98.4578**						
	AS_{33}			**85.6641**					14.3359
	AS_{34}	2.6311			**97.4689**				
	AS_{35}					**100.00**			
	AS_{36}						**98.6936**	1.3064	
	AS_{37}						2.3001	**97.6999**	
	AS_{38}			7.5542					**92.4578**
		PREDICTED LABEL							

Table 7. Performance measured after five-fold cross-validation over UTD-MHAD

Action/Metric	TPR (%)	PPV (%)	F1-Score (%)	FNR (%)	FDR (%)
Swipe Left	94.3606	85.5172	89.7215	5.6394	14.4828
Swipe Right	94.4872	89.4874	91.9193	5.5128	10.5126
Wave	72.6746	74.7131	73.6797	27.325	25.2869
Clap	80.8035	94.7131	87.2071	19.196	5.28690
Throw	76.3555	82.2874	79.2105	23.644	17.7126
Arm Cross	**97.6945**	89.1902	**93.2488**	**2.3055**	10.8098
Basketball Shoot	95.7835	86.1908	90.7343	4.2165	13.8092
Draw X	95.4806	68.3807	79.6897	4.5194	31.6193
Draw Circle (clockwise)	87.3656	55.4147	67.8152	12.6340	44.5853
Draw Circle (counter clockwise)	74.4418	58.4628	65.4917	25.5582	41.5372
Draw Triangle	74.9257	61.8505	67.7631	25.0743	38.1495
Bowling	73.1536	72.6207	72.8861	26.8464	27.3793
Boxing	92.4468	89.2138	90.8015	7.5532	10.7862
Baseball Swing	85.3772	75.3808	80.0681	14.6228	24.6192
Tennis Swing	62.4548	80.5445	70.3554	37.5452	19.4555
Arm Curl	80.7746	74.5531	77.5392	19.2254	25.4469
Tennis Serve	89.3615	87.5645	88.4538	10.6385	12.4355
Push	80.9939	**97.6138**	88.5305	19.0061	**2.38620**
Knock	90.4872	95.5440	92.9468	9.51280	4.45600
Catch	62.8283	89.2138	73.7315	37.1717	10.7862
Pick up & Throw	93.4839	73.5512	82.3282	6.51609	26.4488
Jog	74.4508	92.2890	82.4157	25.5492	7.71100
Walk	90.4536	91.2138	90.8321	9.54640	8.78619
Sit to Stand	81.0257	89.2138	84.9228	18.9743	10.7862
Stand to Sit	89.4872	81.2156	85.1509	10.5128	18.7844
Lunge	90.5169	86.0623	88.2334	9.48309	13.9377
Squat	84.9182	89.8505	87.3147	15.0818	10.1495

Tables 4-6 show the confusion matrices of the results of the developed method on the simulation of AS1, AS2, and S3, respectively. These results show that the maximum number of actions have gained a maximum recognition rate, which is approximately 95% mainly, the activities under AS3. Most of them have a higher recognition rate for the action Pickup & Throw because it is a long-term continuous action, i.e., the candidate must pick up the ball first and then throw. This can be recorded as a combination of different steps like a bend. In such a case, it can be recognized as other actions. However, the proposed method resolves this problem by extracting its main motion features through EDMM. In both sets (AS1 and AS3), it has gained a recognition rate above 95%, proving that the proposed method is much more effective in recognizing complex actions.

4.3.2 Results of UTD-MHAD

Under the simulation of UTD-MHAD, from the accessible 861 depth action videos, we employed exactly half (i.e., 431) for training, and the remaining 431 are used for testing. In this simulation, we also considered five-fold cross-validation by interchanging the subjects. We measured the performance metrics for every action at every validation and averaged. The obtained average values are shown in Table 7.

Arm cross-class achieved maximum TPR of approximately 97.6945% from the observed performance metrics, and Tennis Swing achieved minimum TPR of roughly 62.4548%. Next, the Push action class and Draw Circle (clockwise) can achieve maximum and minimum PPV of approximately 97.6138% and 55.5147%, respectively. Further, the maximum F1-score is 93.2489%, and the minimum F1-score is observed as 65.4917%. In the dataset, a few actions, specifically Draw Triangle, Draw X, and Draw a circle (counterclockwise), have comparative developments, and subsequently they have acquired practically indistinguishable precisions. Due to their identical action movements, they have achieved more FDR. During the simulation, we observed that the Clap, Boxing, and Arms cross-actions are misclassified due to their nature of similar movements. The clap action is approximately 15% recognized as arms cross and 7% as boxing. However, the recognition accuracy is observed as above 85%, which proves the developed method's effectiveness in recognising actions even in the case of activities with only minor differences in their movements. However, the arms cross-action is completely recognized because it differs from boxing and hand clapping. Due to these reasons, the Arm-Cross has gained

maximum recognition performance and Tennis Swing achieved minimum recognition performance. In summary, we can state that, the proposed approach can provide sufficient discrimination between actions even with similar motion semantics.

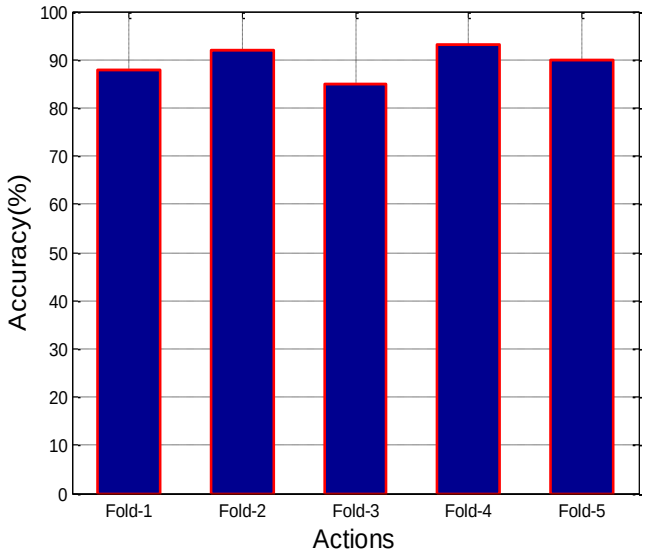

Figure 8. Accuracies at different cross-validations- UTD-MHAD

The accuracies measured at individual validations are demonstrated in Figure 8; from these accuracies, we found that the maximum accuracy is archived at the Fourth validation, and it is approximately 93.2315%. We used even subjects for training and odd subjects for testing at this validation. Even though not many works are executed on this dataset among the available methods, the proposed method gained more considerable recognition accuracy.

4.4 Comparison with state-of-the-art methods

The accuracy comparison between proposed and existing approaches is demonstrated in Table 8. The initial method, i.e., DMM developed by Yang et al. [15], has gained only 85.5218% accuracy. They proposed DMM and represented each action with Histogram features that are not robust for noises. Next, Kim et al. [16] combined the DMM with DMH and DMA and achieved an accuracy of 90.4523%, which was better than the DMM. This approach, however, is more sensitive to changes in noise and texture in motion videos. Chen et al. [18] implemented LBP over DMM to address the texture issue, expressing each action as a fisher kernel vector. Due to the accomplishment of LBP, they have gained an improved accuracy than DMM, which is approximately 89.5214%. However, they didn't concentrate on determining fake motions in the action frame. The primary reason behind the counterfeit gestures is a small body shaking movements and ghost shadows. The Traditional DMM can't provide sufficient information about such disturbances, so they have gained limited recognition accuracy. The proposed EDMM solves this problem; hence, the recognition accuracy is maximum.

Table 8. Accuracy comparison

Method	Dataset	Accuracy (%)
DMM+HOG (extracted) [15]	MSR Action 3D (MSRA3D)	85.5218
DMH+DMA+HOG (extracted) [16]	MSRA3D	90.4523
DMM+LBP+FK (extracted) [18]	MSRA3D	89.5214
Random Occupancy Patterns (extracted) 22]	MSRA3D	86.5536
DSTIPs (extracted) [23]	MSRA3D	89.3012
Bag of 3D Points (extracted) [28]	(MSRA3D)	74.7077
Salient Motion Energy Image + CCA + SVM [30]	UTD-MHAD	84.1200
EDMM-CNN	(MSRA3D)	**93.3369**
EDMM-CNN	UTD-MHAD	**85.6696**

On the other hand, the methods like ROP [22], Bag of 3D points [28], and DSTIPs have achieved better recognition accuracy; they are significantly less concentrated on the real-time issues and in-depth action sequences. They mostly worked on high-resolution videos and did not mention a tiny idea about shadows and body shaking movements. Hence, they are much more robust for a real-time environment with many problems. Our method has gained superior recognition accuracy with all these problems because the proposed EDMM nullifies the fake moving pixels and represents the action with the regions with significant movement. Hence, it has achieved a maximum accuracy (93.3369%) on MSR action 3D and 88.6696% on the UTD-MHAD dataset.

5. CONCLUSION

A system for recognizing human actions is developed here with the help of depth action videos and machine learning. The proposed new action descriptor concentrates on the fake movements and nullifies them. The unnatural movements are mainly due to several real-time problems, and the proposed descriptor effectively represents the motion by observing its importance pixel-wise. The proposed EDMM scans each pixel and analyzes its nature with the help of its neighboring pixels. Since fake moving regions are constructed with only few connected pixels, the number of neighboring pixels are less. Further, a new and straightforward CNN model has been proposed for feature extraction and classification. The proposed CNN is a customized model inspired from AlexNet named pretrained model. Two benchmark depth action datasets were used to evaluate the created system. According to the results, it was found that the suggested strategy is more efficient than several other approaches. Adding a segmentation method that divides the input action video frames into segments can further improve the work. Herein the current paper, we considered the entire frames of action video as input for EDMM. However, there exists only few frames are significant and remaining are insignificant. Segmentation of such kind of frames will improve the recognition performance further.

REFERENCES

[1] Liu, X., You, T., Ma, X., Kuang, H. (2018). An optimization model for human activity recognition inspired by information on human-object interaction. In 2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), Changsha, China, pp. 519-523. https://doi.org/10.1109/ICMTMA.2018.00131

[2] Yussiff, A.L., Suet-Peng, Y., Baharudin, B.B. (2016). Human action recognition in surveillance video of a computer laboratory. In 2016 3rd International Conference on Computer and Information Sciences (ICCOINS), Lumpur, Malaysia, pp. 418-423. https://doi.org/10.1109/ICCOINS.2016.7783252

[3] Chen, C., Kehtarnavaz, N., Jafari, R. (2014). A medication adherence monitoring system for pill bottles based on a wearable inertial sensor. In 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, USA, pp. 4983-4986. https://doi.org/10.1109/EMBC.2014.6944743

[4] Chen, C., Liu, K., Jafari, R., Kehtarnavaz, N. (2014). Home-based senior fitness test measurement system using collaborative inertial and depth sensors. In 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, USA, pp. 4135-4138. https://doi.org/10.1109/EMBC.2014.6944534

[5] Bloom, V., Makris, D., Argyriou, V. (2012). G3D: A gaming action dataset and real time action recognition evaluation framework. In 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, pp. 7-12. https://doi.org/10.1109/CVPRW.2012.6239175

[6] Zhang, H.B., Zhang, Y.X., Zhong, B., Lei, Q., Yang, L., Du, J.X., Chen, D.S. (2019). A comprehensive survey of vision-based human action recognition methods. Sensors, 19(5): 1005. https://doi.org/10.3390/s19051005

[7] Subetha, T., Chitrakala, S. (2016). A survey on human action recognition from videos. In Proceedings of the IEEE 2016 International Conference on Information Communication and Embedded Systems, Chennai, India, pp. 25-26. http://doi.org/10.1109/ICICES.2016.7518920

[8] Kong, Y., Fu, Y. (2022). Human action recognition and prediction: A survey. International Journal of Computer Vision, 130(5): 1366-1401. https://doi.org/10.1007/s11263-022-01594-9

[9] Ni, B., Wang, G., Moulin, P. (2011). Rgbd-hudaact: A color-depth video database for human daily activity recognition. In 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), Barcelona, pp. 1147-1153. https://doi.org/10.1109/ICCVW.2011.6130379

[10] Zhang, Z., Ma, X., Song, R., Rong, X., Tian, X., Tian, G., Li, Y. (2017). Deep learning based human action recognition: A survey. In 2017 Chinese automation congress (CAC), Jinan, China, pp. 3780-3785. https://doi.org/10.1109/CAC.2017.8243438

[11] Chen, L., Wei, H., Ferryman, J. (2013). A survey of human motion analysis using depth imagery. Pattern Recognition Letters, 34(15): 1995-2006. https://doi.org/10.1016/j.patrec.2013.02.006

[12] Herath, S., Harandi, M., Porikli, F. (2017). Going deeper into action recognition: A survey. Image and Vision Computing, 60: 4-21. https://doi.org/10.1016/j.imavis.2017.01.010

[13] Wang, P., Li, W., Ogunbona, P., Wan, J., Escalera, S. (2018). RGB-D-based human motion recognition with deep learning: A survey. Computer Vision and Image Understanding, 171: 118-139. https://doi.org/10.1016/j.cviu.2018.04.007

[14] Yang, X., Zhang, C., Tian, Y. (2012). Recognizing actions using depth motion maps-based histograms of oriented gradients. In Proceedings of the 20th ACM International Conference on Multimedia, pp. 1057-1060. https://doi.org/10.1145/2393347.2396382

[15] Chen, C., Liu, K., Kehtarnavaz, N. (2016). Real-time human action recognition based on depth motion maps. Journal of Real-Time Image Processing, 12: 155-163. https://doi.org/10.1007/s11554-013-0370-1

[16] Chen, C., Liu, M., Zhang, B., Han, J., Jiang, J., Liu, H. (2016). 3D action recognition using multi-temporal depth motion maps and Fisher vector. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, pp. 3331-3337.

[17] Chen, C., Jafari, R., Kehtarnavaz, N. (2015). Action recognition from depth sequences using depth motion maps-based local binary patterns. In 2015 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, pp. 1092-1099. https://doi.org/10.1109/WACV.2015.150

[18] Beddiar, D.R., Nini, B., Sabokrou, M., Hadid, A. (2020). Vision-based human activity recognition: A survey. Multimedia Tools and Applications, 79(41): 30509-30555. https://doi.org/10.1007/s11042-020-09004-3

[19] Kim, D., Yun, W.H., Yoon, H.S., Kim, J. (2014). Action recognition with depth maps using HOG descriptors of multi-view motion appearance and history. In the Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, UBICOMM, pp. 2308-4278.

[20] Al-Faris, M., Chiverton, J., Yang, Y., Ndzi, D. (2019). Deep learning of fuzzy weighted multi-resolution depth motion maps with spatial feature fusion for action recognition. Journal of Imaging, 5(10): 82. https://doi.org/10.3390/jimaging5100082

[21] Vieira, A.W., Nascimento, E.R., Oliveira, G.L., Liu, Z., Campos, M.F. (2012). Stop: Space-time occupancy patterns for 3D action recognition from depth map sequences. In Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 17th Iberoamerican Congress, CIARP 2012, Buenos Aires, Argentina, pp. 252-259. https://doi.org/10.1007/978-3-642-33275-3_31

[22] Wang, J., Liu, Z., Chorowski, J., Chen, Z., Wu, Y. (2012). Robust 3D action recognition with random occupancy patterns. In Computer Vision-ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, 12: 872-885. https://doi.org/10.1007/978-3-642-33709-3_62

[23] Xia, L., Aggarwal, J.K. (2013). Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, pp. 2834-2841. https://doi.org/10.1109/CVPR.2013.365

[24] Li, J., Ban, X., Yang, G., Li, Y., Wang, Y. (2019). Real-

time human action recognition using depth motion maps and convolutional neural networks. International Journal of High Performance Computing and Networking, 13(3): 312-320. https://doi.org/10.1504/IJHPCN.2019.098572

[25] Arivazhagan, S., Shebiah, R.N., Harini, R., Swetha, S. (2019). Human action recognition from RGB-D data using complete local binary pattern. Cognitive Systems Research, 58: 94-104. https://doi.org/10.1016/j.cogsys.2019.05.002

[26] Sargin, M.E., Yemez, Y., Erzin, E., Tekalp, A.M. (2007). Audiovisual synchronization and fusion using canonical correlation analysis. IEEE Transactions on Multimedia, 9(7): 1396-1403. https://doi.org/10.1109/TMM.2007.906583

[27] Zhang, Z., Wei, S., Song, Y., Zhang, Y. (2017). Gesture recognition using enhanced depth motion map and static pose map. In 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), Washington, DC, USA, pp. 238-244. https://doi.org/10.1109/FG.2017.38

[28] Kamel, A., Sheng, B., Yang, P., Li, P., Shen, R., Feng, D.D. (2019). Deep convolutional neural networks for human action recognition using depth maps and postures. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49(9): 1806-1819. https://doi.org/10.1109/TSMC.2018.2850149

[29] Ji, X.P., Cheng, J., Feng, W., Tao, D. (2018). Skeleton embedded motion body partition for human action recognition using depth sequences. Signal Process, 143: 56-68. https://doi.org/10.1016/j.sigpro.2017.08.016

[30] Fan, Y., Weng, S., Zhang, Y., Shi, B., Zhang, Y. (2020). Context-aware cross-attention for skeleton-based human action recognition. IEEE Access, 8: 15280-15290. http://doi.org/10.1109/ACCESS.2020.2968054