Available online at www.sciencedirect.com

ScienceDirect

Cognitive Systems Research 58 (2019) 71–93

Cognitive Systems RESEARCH

www.elsevier.com/locate/cogsys

Hexagonal Volume Local Binary Pattern (H-VLBP) with deep stacked autoencoder for Human Action Recognition

Kiruba K [a], Shiloah Elizabeth D [a,*], Sunil Retmin Raj C [b]

[a] *Department of Computer Science and Engineering, Anna University, CEG Campus, Chennai 600025, Tamil Nadu, India*
[b] *Department of Information Technology, Anna University, MIT Campus, Chennai 600044, Tamil Nadu, India*

Received 21 September 2018; received in revised form 2 February 2019; accepted 1 March 2019
Available online 15 May 2019

Abstract

Human action recognition plays a significant role in a number of computer vision applications. This work is based on three processing stages. In the first stage, discriminative frames are selected as representative frames per action to minimize the computational cost and time. In the second stage, novel neighbourhood selection approaches based on geometric shapes including triangle, quadrilateral, pentagon, hexagon, octagon and heptagon are used in Volumetric Local Binary Pattern (VLBP) to extract the features from frame sequences based on motion and appearance information. Hexagonal Volume Local Binary Pattern (H-VLBP) descriptor has been found to produce better results among all other novel geometric shape based neighbourhood selection approaches for human action recognition. However, the dimensionality of extracted feature from H-VLBP is too large. Therefore, the deep stacked autoencoder is used for dimensionality reduction with the decoder layer replaced by softmax layer for performing multi-class recognition. The developed approach is applied to four publicly available benchmark datasets, namely KTH, Weizmann, UCF11 dataset and IXMAS dataset for human action recognition. The results obtained show that the proposed approach outperforms the state-of-art techniques. Moreover, the approach has been tested with a synthetic dataset and better results have been obtained. This illustrates the effectiveness of the approach in real time environment.

Keywords: Action recognition; Local binary pattern; H-VLBP; Deep stacked autoencoder

1. Introduction

Recognizing human actions in videos has emerged as one of the significant research problems in the field of computer vision, popularly known as Human Action Recognition (HAR). It tries to imitate the human visual system by focusing on regions that involve human actions as Region of Interest (ROI) in the videos. HAR finds application in surveillance, video retrieval, robotics, abnormal activity recognition and many more fields. Generally, HAR is believed to have at least the following properties: (i) it should detect the precise boundaries of the ROI with complete information (ii) high computational efficiency high with low computational complexity. (iii) high accuracy even if the number of actions increases. The key issues concerning HAR (Nguyen, Li, & Ogunbona, 2016; Popoola & Wang, 2012) are cluttered backgrounds, changes of viewpoint, illumination changes, camera motion, variations of human clothes and postures occlusions. The challenging problem in most of the HAR methods (Popoola & Wang, 2012) are variance to geometric transformation such as rotation, scale, translation and high computational complexity.

* Corresponding author.
E-mail address: shiloah@annauniv.edu (E.D. Shiloah).

https://doi.org/10.1016/j.cogsys.2019.03.001

General HAR techniques (Nguyen et al., 2016; Popoola & Wang, 2012) are human body model based methods (2D or 3D information on human body), holistic methods (Shape and silhouette) and local feature methods. This paper fully focuses on a local feature descriptor. One of the local feature detectors is Spatio-Temporal Interest Point (STIP) detectors. Recently, researchers have developed much interest in analysing the human motion or actions in a spatio-temporal domain such as Motion Energy Image (MEI) (Ahad, Tan, Kim, & Ishikawa, 2012), Motion History Image (MHI), Optical Flow (OF) (Chaudhry, Ravichandran, Hager, & Vidal, 2009), and so on. The popular interest point detectors (Krig, 2016) are Scale Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF), Binary Robust Invariant Scalable Keypoints (BRISK), Histogram of Oriented Gradients(HOG), Harris corner detector, Features from Accelerated Segment Test (FAST), Local Binary Pattern (LBP), and so on.

Ojala, Pietikäinen, and Mäenpää (2000) have introduced the Local Binary Pattern (LBP) technique for texture classification. The traditional LBP operator labels the pixels of an image by thresholding a circular neighbourhood region. $LBP_{P,R}$ generates 2^P different values on the radius of R which represents the 2^P different binary patterns. Advantages of LBP are high discriminative power and computational simplicity with good performance. Additionally, it is invariant to gray scale changes (Illumination invariant). The main limitation of LBP is that it is not invariant to rotations. It computes the difference of pixel values only with limited structural information and ignores magnitude or orientation information. Feature vector size will vary with respect to the number of neighbours. Additionally, space and time complexity also varies with respect to the number of neighbours considered.

The effectiveness of LBP technique is extended to Volume LBP (VLBP) (Zhao & Pietikainen, 2007) for capturing dynamic information in dynamic face recognition and HAR. VLBP is used to extract the motion and appearance information of the dynamic sequences introduced for dynamic texture analysis. The temporal information extracted by the VLBP features consists of both intra and extra personal dynamics. In the VLBP procedure, gray values of frames are modelled as volume data. In the extracted frames, overlapping three continuous frames are taken as input and the three continuous frames are divided into non-overlapping $3 \times 3 \times 3$ voxels. VLBP techniques are applied to it using the difference of gray scale values in the volume data. The difference of gray value is calculated by the center gray value of the current frame of $3 \times 3 \times 3$ voxels and the neighbour values of the previous, current and next frames. If the centre value is greater than neighbour points then the values are replaced by 0's, else 1's. Five different neighbourhood selection topologies (Ojala, Pietikainen, & Maenpaa, 2002) such as circle, ellipse, parabola, hyperbola and spiral have been used. Each topology has some advantages and disadvantages. Spiral topology (Kazak & Koc, 2016) is rotation invariant but cannot exploit the anisotropic structure information. This information may be an important source of some problems. An elliptical neighbourhood topology (Nguyen et al., 2016) has been used to exploit this anisotropic structural information than circle topology. The main drawback in basic LBP and VLBP is the extraction of a large number of features. Uniform LBP overcomes the limitation of large feature vectors. Uniform pattern is used to reduce the large data into a small subset. LBP-TOP (Three Orthogonal Plane) (Kellokumpu, Zhao, & Pietikäinen, 2008), Local Ternary Pattern (LTP) (Yeffet & Wolf, 2009), Motion Binary Pattern (MBP) (Baumann, Lao, Ehlers, & Rosenhahn, 2014) and so many extended LBP techniques have been introduced in HAR.

In this paper, the H-VLBP descriptor is developed by taking into consideration the motion and temporal information. The proposed approach is introduced to improve the performance of HAR with respect to computational speed and recognition accuracy.

The contributions of this work are as follows,

1. Instead of working with all the frames or key frames, discriminative frames per action are selected as representative frames for the actions that give effective results in HAR classification.
2. The original VLBP representation is altered by a 6-point hexagonal neighbourhood selection in the volume based histogram extraction instead of using circular symmetric selection which results in computational simplicity with ease of use.
3. Some geometric neighbourhood selection topologies are introduced.
4. Binary patterns created by hexagonal neighbourhood points in VLBP is variant to rotation. Hence rotation invariance is achieved using the bitwise shift operator.
5. Deep stacked autoencoder is used for dimensionality reduction. The output layer of the autoencoder is replaced by softmax layer. The softmax layer is trained using supervised learning for multi class human action recognition.

The remainder of the paper is organized as follows: First, a comprehensive survey on related works is provided in Section 2. The proposed HAR methodology including the preprocessing, proposed neighbourhood selection topologies, H-VLBP, rotation invariance and deep stacked autoencoder is explained in Section 3. The developed approach has been applied to four benchmark datasets and one synthetic dataset. The experimental setup and results are reported in Section 4. Section 5 compares the two layer FFNN and deep stacked autoencoder with respect to different parameter initializations and the corresponding recognition accuracy. Conclusion and future work are included in Section 6.

2. Related works

Human action recognition is an important area of research in the field of computer vision. The research related to the various feature extraction, machine learning and deep learning techniques (Akula, Shah, & Ghosh, 2018) involved in human action recognition are discussed in this section.

Cheng et al. have proposed a supervised temporal t-Stochastic Neighbor Embedding (ST t-SNE) and incremental learning for human action recognition (Cheng, Liu, Wang, Li, & Zhu, 2015). The main contribution of their paper is silhouette sequential analysis based on ST t-SNE which has been introduced to preserve the intrinsic action structure with dimensionality reduction. They have extracted discriminative features to introduce class label information and temporal information into manifold learning methods. They have done the experiments on INRIA Xmas Motion Acquisition Sequences (IXMAS) dataset, Weizmann dataset and National Laboratory of Pattern Recognition (NLPR) gait dataset. They have achieved 100% recognition rate with dimensionality reduction. Their proposed approach is effective in class labelling and extraction of temporal information. Additionally, without any preprocessing, their approach gives optimal low-dimensional representation self-adaptively. But they have failed to analyse the technique in complex action or activity datasets with a dynamic background.

Chun and Lee have proposed a new motion descriptor namely, Histogram of Motion Intensity and Direction (HOMID) for human action recognition (Chun & Lee, 2016). They have estimated HOMID by using the optical flow method. They have plotted a regular grid on an image to partition the motion flow into sub-regions and the feature vector of each sub-region is framed by local flow direction and its intensity with less computational power. They have used Support Vector Machine (SVM) for action classification. They have overcome the limitation of dependency of camera view and narrow area coverage by multiple views observation with less computational power and memory. They have tested their experiment in i3DPost and IXMAS database and have achieved 98.96% and 83.03% recognition rate respectively. From the literature, it is inferred that their proposed technique has been tested only with the single action in a video with better accuracy but the presence of noise in the input increases the false positive values in the motion intensity and direction. The accuracy rate of action recognition has reduced with an increase in the number of actions in a video.

Su et al. have proposed a multi-attribute sparse-coding approach for action recognition from a single unknown viewpoint (Su, Chiang, & Lai, 2016). In their work, first, over-segmentation based background modelling and foreground detection approach have been used to extract silhouette from action videos by computing the Multi-interval Motion History Image (MMHI). Second, multi-view action video classification has been done by multi-attribute sparse representation. Finally, a random walk algorithm has been used to assign appropriate attribute values to the unlabelled actions in the training data. They have demonstrated the effectiveness of their proposed method on three public multi-view human action datasets namely, i3DPost, 3Dlife and IXMAS. They have achieved 77.2% of accuracy in i3DPost dataset, 65.6% of accuracy in 3Dlife dataset and 74.3% of accuracy in IXMAS dataset. The merit of their work is that it can be used to segment the foreground and shadow which provide more accurate human silhouettes to extract the features for human action recognition. They have tested their proposed work only with multi-view single actions.

Xu et al. have presented the two stream dictionary learning architecture for action recognition (Xu, Jiang, & Sun, 2017). In their paper, the Interest Patches (IP) detector based on human detector, background subtraction and contour detector has been used to extract IPs on human contours. Then, in order to compute the spatial and temporal streams, IP descriptors have been used to extract the pixel values, gradients and optical flow. They have trained the spatial SVM and temporal SVM based on the IP distribution histograms and the scores of these two SVM are fused to make a final decision in recognizing action. Two stream dictionaries have been created for each action in the benchmark dataset: one dictionary corresponds to the spatial stream and the other one corresponds to the temporal stream. Their work has been found capable of handling videos with camera motion and cluttered background. They have tested their two stream dictionary learning architecture in Weizmann datasets with 99.1% accuracy, KTH datasets with 95.8%, Olympic sports dataset with 88.81% and HMDB51 dataset with 59.47%. They have worked with single actions per video.

Guo et al. have proposed the 3D gradient LBP algorithm for action recognition (Guo, Wang, & Xie, 2017). The main contribution in their paper is the extraction of STIP and calculation of the gradient cuboids from dense sampling data which gives six planes in each 3D patch. The six planes are front, rear, left, right, above and below. They have compared the average value of each plane with local features in the given threshold value. Based on threshold comparison, histogram values are recorded and two histogram values are concatenated and used for classification of HAR. They have done the experiments on the KTH, Weizmann and UT interaction dataset. They have achieved 92.25% of correctness in KTH dataset, 92.88% of correctness in Weizmann dataset and 91.42% of correctness in UT-interaction dataset.

Qu and Li have proposed HAR based on improved Co-occurrence Histogram Oriented Gradients (CoHOG)-Local Quantization Code (LQC) (Qu & Li, 2017). In their work, they have fused the LQC and CoHOG features for the detection and recognition of human actions. LQC feature descriptor has been used to extract the spectral property of the image. LQC character spectral property has been used to calculate the edge characteristics. They have used Principal Component Analysis (PCA) to reduce the dimensionality. They have used histogram interaction kernel support vector machine (HIKSVM) classification for HAR. They have experimented their work in KTH, Weizmann and Hollywood2 and compared their results with the following methods, CoHOG, HOG_LBP_HIKSVM, COGMULBP and CoHOG_LQC_HIKSVM.

Al-Berry et al. have proposed the fusion of directional wavelet LBP and moments for HAR (Al-Berry, Salem, Ebeid, Hussein, & Tolba, 2016). In their work, they have combined the advantages of local and global descriptors using wavelet transformation. They have fused the Hu invariant moments as global descriptors with 3-D stationary wavelet transform and incorporated the LBP concepts. It is called Directional Wavelet-LBP (DW-LBP). They have extracted feature vectors from the DW-LBP method and have used five different directional bands individually to train separately on multi-class data. Finally, voting scheme has been used to find the best match. They have experimented their work separately with basic LBP-moments in KTH and Weizmann dataset. They have achieved 96% recognition rate in KTH dataset and 91.4% in Weizmann dataset using Decision Tree (DT) classifier.

Li et al. have proposed a method for action recognition (Li, Yu, He, Sun, & Ge, 2016). They have proposed a method based on multiple key motion history images. First, they have selected the key MHI using entropy of MHIs. Second, they have described the Spatial Pyramid Matching (SPM) for describing the spatio-temporal information of actions. Two-dimensional entropy of MHI and Zernike moments of Motion History Images Edge (MHIE) are combined as feature vectors based on SPM. SVM classification has been used to classify the human actions. They have shown the comparison of proposed work with LBP_H using the KTH dataset. They have achieved a recognition rate of 94.0% using the proposed method and 73.0% using LBP-H on multi MHIs.

Ahsan et al. have proposed a histogram of spatio-temporal LBP for HAR (Ahsan, Tan, Kim, & Ishikawa, 2014). They have used Directional MHI (DMHI) as Spatio-Temporal template and LBP to extract the features from the Spatio Temporal template and have converted the outcome histogram to feature vector. Additionally, they have extracted selective silhouettes as shape features. Finally, they have concatenated the two features and fed the feature vectors to SVM classifier. They have experimented their work in Weizmann dataset. They have validated this result using the 10-fold cross validation method. They have achieved 90.56% of accuracy using MHI_LBP_H, 90.56% using MHI_LBP_H+SF (Silhouette Features), 93.15% using DMHI_LBP_H and 94.26% using DMHI_LBP_H+SF.

Ji et al. have proposed a novel 3D CNN model for HAR (Ji, Xu, Yang, & Yu, 2013). In their work, they have proposed a 3D convolution operation to extract the spatial and temporal features from videos. Seven frames have been taken as raw input with size of 60 × 54. They have applied a set of hardwired kernels on the raw input and to generate multiple channels of information including five different channels such as gray, gradient-x, gradient-y, optflow-x and optflow-y. They have applied 3D convolutions with a kernel size of 7 × 7 × 3 on each of the five channels separately. In order to increase the feature maps, authors have used two sets of different convolutions at each location. After the three layers of convolution and two layers of subsampling, they have converted the seven input frames into a 128 dimensional feature vector. They have evaluated their experiments on TREC Video Retrieval Evaluation (TRECVID) 2008 and the KTH dataset. They have analysed the different combinations in 3D CNN and have achieved 78.28% of precision in TRECVID 2008 dataset and 90.2% of recognition accuracy in KTH dataset.

Shi et al. have introduced a three stream CNN framework for human action recognition (Shi, Tian, Wang, & Huang, 2016). In this work, they have proposed Sequential Deep Trajectory Descriptor (SDTD) to extract the dense trajectories and then these trajectories are converted into sequential trajectories, which is a long-term motion descriptor. They have projected the dense trajectories into two dimensional planes and CNN-RNN network is used to learn an effective representation for long-term motion using spatial stream, temporal stream and SDTD stream. Unlike two stream static spatial features, they have extracted short term motion and long term motion in the video. Finally, they have focused on deep neural networks namely, CNN and LSTM to learn the spatial features and capture the temporal features, respectively. They have evaluated their experiments on KTH, HMDB51 and UCF101 datasets. They have achieved 96.8% of recognition accuracy on KTH, 65.2% on HMDB and 92.2% on UCF101 dataset using three-stream CNN.

Baccouche et al. have proposed a fully automated deep model for HAR without using any prior information (Baccouche, Mamalet, Wolf, Garcia, & Baskurt, 2011). First, they have used 3DCNN to extract spatio-temporal information. Second, they have employed RNN to classify each sequence at each timestep. They have used LSTM to overcome the limitation of short term memory in RNN. They have experimented their work on the KTH dataset and achieved 92.17% of recognition accuracy. Buonamente et al. have proposed a hierarchical neural architecture to recognize human actions (Buonamente, Dindo, & Johnsson, 2016). They have used Self Organizing Maps (SOM) in each layer with different objectives. They have experimented their work on INRIA 4D repository and achieved better recognition rate.

Veeriah et al. have proposed the differential RNN (dRNN) model (Veeriah, Zhuang, & Qi, 2015). They have learnt the salient spatio-temporal representations of actions to overcome the limitation of LSTM model which fails to capture salient dynamic patterns. They have evaluated their experiments on the KTH 2D dataset and the MSR action 3D dataset. They have extracted HOG3D features from the KTH2D dataset and depth sensor information features from MSR action 3D dataset such as position, angle, offset, velocity and pairwise join distances. They have achieved 93.28% and 91.98% of accuracy on KTH-1 and KTH-2 using 1-order dRNN+HOG3D, 93.96% and 92.12% of accuracy on KTH-1 and KTH-2 using 2-order dRNN+HOG3D, 91.40% of accuracy using 1-order dRNN and 92.03% of accuracy using 2-order dRNN on MSRaction3D dataset.

Katircioglu et al. have introduced a deep learning regression architecture for structure prediction of 3D human pose from monocular images or 2D joint location heat maps (Katircioglu, Tekin, Salzmann, Lepetit, & Fua, 2018). They have combined autoencoders with CNNs to improve the dependencies between human body parts effectively and this combination improves the accuracy of pose estimation. In their approach, they have trained a stacked denoising autoencoder which learns the structural information and enforces implicit constraints about human body in its latent middle layer (latent pose representation). Then the CNN architecture maps the raw image or the 2D joint location heat map predicted from the input image to the latent representation learnt by the autoencoder. Finally, they have stacked the decoding layers of the autoencoder on top of the CNN for reprojection from the latent space to the original pose space and the entire network gets fine-tuned by updating the parameters of all the layers. They have evaluated their method on human 3.6 m, Human Eva, KTH multi view Football II and leads sports pose (LSP) datasets.

Ijjina et al. have proposed a method for classification of human action using pose based features (Ijjina, 2016). In their work, they have extracted the pose information from the input observation. They have used these pose information to compute pose-based distance measurements. Then the distance measures are evaluated by a set of fuzzy membership functions which is designed to get the value of the unique motion pattern of each action. Finally, they have given the input representations to the stacked autoencoder for classification. They have used 100 and 50 neurons in first and second layers, respectively. Then, the last layer is softmax layer which consists of n neurons to produce the classification results, where n refers to the number action classes. They have experimented their work on CMU MOCAP and Berkely MHAD datasets. They have achieved 97.47% of recognition accuracy on the CMU MOCAP dataset and 98.03% of recognition accuracy on the Berkely MHAD dataset.

Inferred from the related research work on HAR approaches, most of the research work focus on the spatio temporal information extraction from video. VLBP approach has been introduced for dynamic texture extraction and has the limitations of dimensionality in the feature vector. Dimensionality of the feature vector depends on the number of neighbours. Hence, (Ojala et al., 2000) proposed the LBP-TOP technique as an alternate to VLBP with reduced feature dimension. No researchers has focused on the dimensionality reduction in VLBP because of the improvement of LBP-TOP. In this work, deep stacked autoencoder has been used to reduce the dimensionality of the feature vector and reduce the computational complexity. Additionally, VLBP has been modified by introducing various geometric shape-based neighbourhood selection approaches to improve the recognition accuracy and hexagonal neighbourhood selection provides better results among other geometric shapes.

The proposed approach focuses on effective feature extraction using different geometric shape based neighbourhood selection approaches, and the deep stacked autoencoder has been used for dimensionality reduction, thereby reducing the computational complexity. Softmax classifier is used for multi class classification and it has the impacts in recognition accuracy. The objective of this work is to improve the recognition accuracy and reduce the computational time.

3. Proposed methodology

The proposed approach consists of three phases. The first phase involves preprocessing of frames. In this step, discriminative frames per action are selected from meaningful frames. Spatially normalized ROI frames are used as input sequences. Then in the second phase, geometric shape based neighbourhood topologies has been introduced. The Hexagonal shape based neighbourhood selection has been found to perform better experimental results among all other novel geometric shape based neighbourhood selection. The resultant H-VLBP histograms are normalized and converted as feature vectors. Finally in the third phase, feature vectors are fed to the deep stacked autoencoder for dimensionality reduction and the output layer is activated by the softmax function which is used for multi class human action recognition. The procedure of the H-VLBP approach is discussed in Section 3.2.4. Fig. 1 shows the detailed workflow of the proposed approach.

3.1. Data pre-processing

3.1.1. Extracting meaningful frames

Each video V_i is converted into n number of frames F_i. The extracted frames may contain repetitive sequences of actions, for instance, 3 to 4 times of repetitive sequences per video. Processing all the frames is computationally complex and a time-consuming process. In order to overcome this limitation, meaningful frames M_{F_i} are extracted from frames F_i. Thereby partially visible human and empty frames (without object) are selected from the frame sequences because discriminative information to recognize the action may not be present in the partially visible human body. Therefore, human detection techniques (Nguyen et al., 2016) or manual cropping process (Chun & Lee, 2016) is used for the extraction of full body human motion in the selected frames to detect the foreground Region of

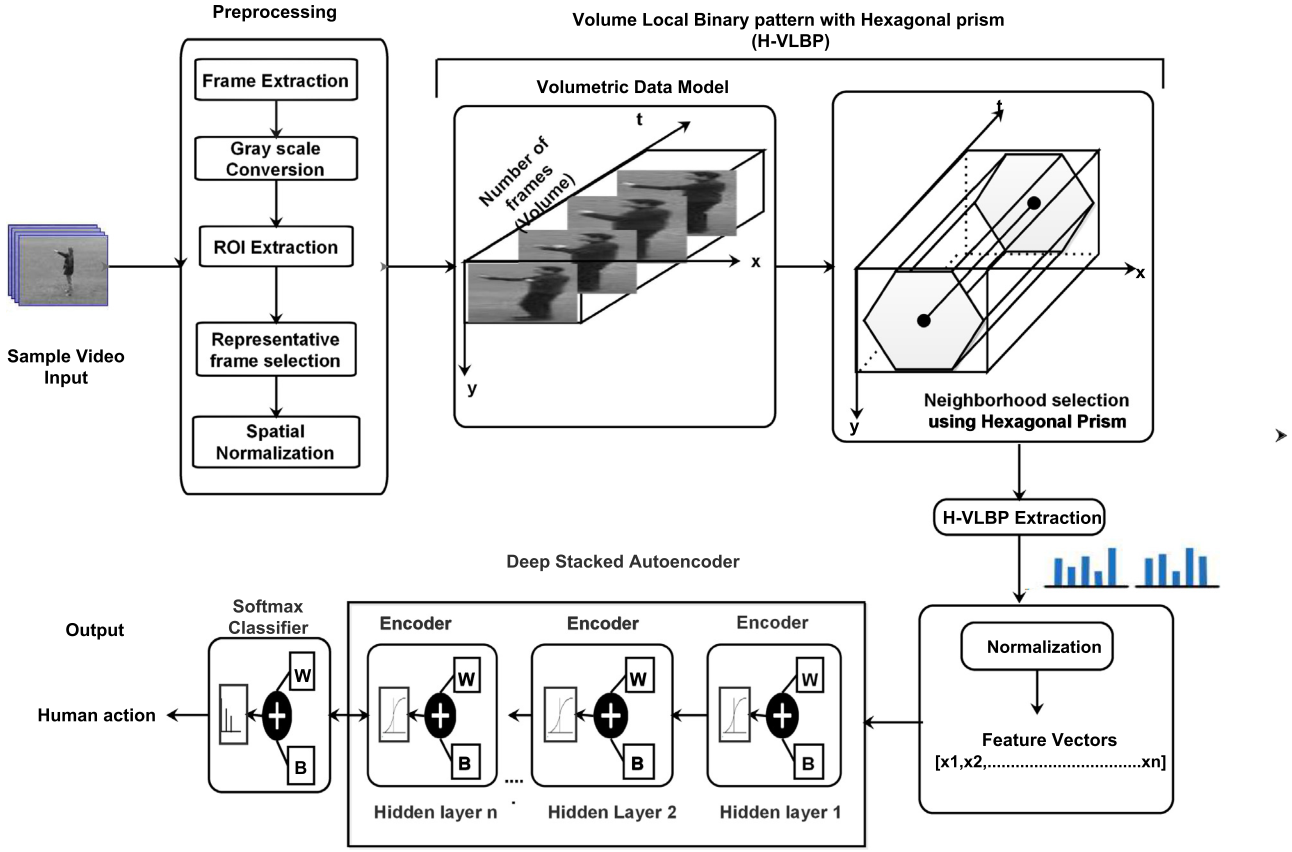


Fig. 1. The workflow of the proposed approach.

Interest (ROI). Additionally, the person size may vary depending on whether the human moves away from the camera or towards the camera. Hence the entire ROI frame sequence is resized to uniform height and weight as spatial normalization (Chun & Lee, 2016) to remove the translation and scaling variance. Spatial normalization is also used for dimensionality problems in frame sequences. The spatially normalized meaningful frames are defined using Eq. (1),

$$M_i = resize(ROI(M_{F_i})) \tag{1}$$

where M_i represents the meaningful frame which is the spatially normalized ROI.

3.1.2. Discriminative frames per action

The sequence of action will be repetitive 2–4 times in each video. A single sequence is selected and used instead of working with multiple sequences in a single video. In this paper, discriminative frames are chosen manually from the single sequence instead of working with all the frames or key frames (Sheena & Narayanan, 2015). For example, in boxing action, left hand and left leg movement, right hand and right leg movement, hand in boxing position are the discriminative frames chosen as representative frames which contains 9–10 frames. Temporal normalization is not applied in the process, hence each image sequence may consist of different number of frames.

$$D_i = Discrminative\_frames[M_i] \tag{2}$$

D_i represents the discriminative frames per action which is represented in Eq. (2). The number of discriminative frames may range from 3 to 10 frames for every action because a minimum of three frames are required to recognize the action.

3.2. VLBP with hexagonal prism

3.2.1. Volumetric data model (Voxels)

A volumetric pixel or volume pixel or voxel is the three dimensional equivalent of a pixel and the tiniest distinguishable element of a 3D projection. It is a volume element that represents a specific grid value in 3D space. Like pixels, voxels do not contain information about their position in 3D space. Rather coordinates are inferred based on their positions relative to other surrounding voxels. In this volume data model represented in Eq. (3), height, width and depth of the frames are considered. The volumetric data model,

$$v = [H; W; D] \tag{3}$$

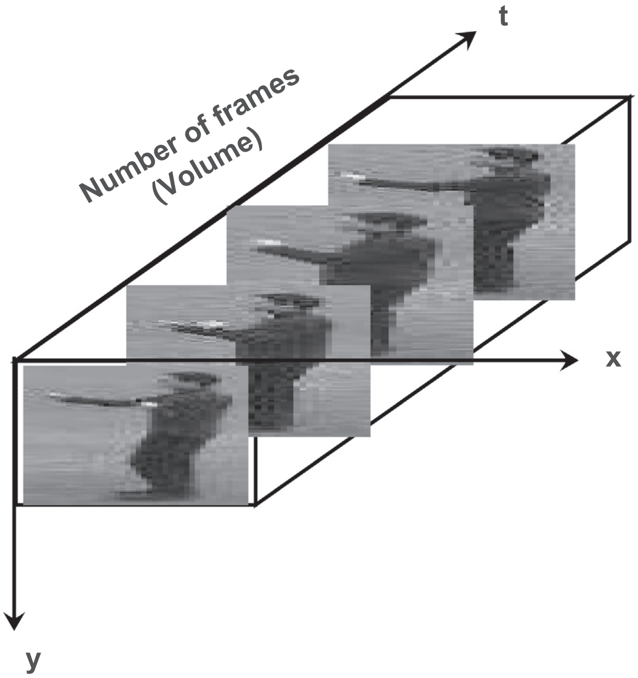


Fig. 2. Volume data model (xyt- 3 Dimensional Data).

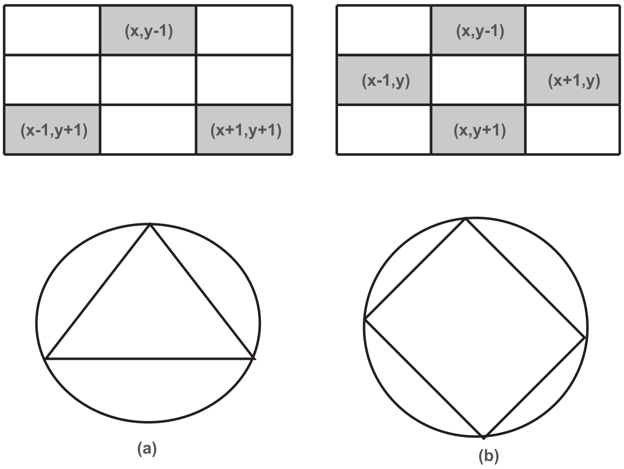


Fig. 3. Neighbour selection topologies. (a) Triangle (b) Quadrilateral.

where H and W is height and width of the frames that exist in x and y axes of spatial domain and D is the length of the frames that exist in t-axis of the temporal domain. The rectangular prism has been constructed using v. The rectangular prism is constructed using x, y and t axis. Fig. 2 shows the volume data model.

3.2.2. Introduction and motivation behind the proposed geometric shape-based neighbourhood topologies

Surrounding pixels of the centre are named as neighbour points. Ojala et al. (2000) introduced five general neighbourhood topologies, namely, circle, ellipse, parabola, hyperbola and spiral. Prewitt hexagonal mask (Vidya, Veni, & Narayanankutty, 2009) and Sobel operators on the hexagonal structure (He, Wu, Jia, & Hintz, 2008) show that hexagonal structure pixel values lead to fast computation and accurate localization. The inspiration is obtained from these neighbourhood topology and hexagonal mask structure. Geometric shape based Neighbourhood topologies have been developed instead of using 4-point or 8-point circular symmetric neighbourhood selection. Triangle and Quadrilateral neighbour selection topologies are shown in Fig. 3. Additionally, some of the geometric neighbourhood topologies such as pentagon with five number of neighbours, heptagon with seven number of neighbours and octagon with eight number of neighbours are introduced and shown in Fig. 4. In 3 × 3 × 3 voxel, pentagon and heptagon structures are compact. From the structure, neighbours are selected.

The selection of neighbours may have meaningful information in LBP. Most of the researchers have discussed circular symmetric neighbours using 4- points or 8- points. Some of the researchers have used elliptical topology (Nguyen & Caplier, 2012), parabola, hyperbola and spiral techniques are addressed in some literature. In this paper, geometric neighbourhood topologies such as hexagon, pentagon, heptagon and octagon have been experimented and results are shown in Table 10. Experimental result shows that hexagonal neighbourhood selection performs well. Hexagonal neighbourhood selection is used in this paper which is discussed in Section 3.2.3.

3.2.3. H-VLBP

The main advantage of LBP technique is invariance to monotonic changes in gray scale values (Ojala et al., 2000). H-VLBP technique is applied over constructed rectangular volumetric data model. The gray values within the rectangular prism are considered. The procedure for extraction of H-VLBP Descriptor is summarized in Algorithm 1. The following theoretical view gives the details of H-VLBP in HAR.

Let F_i represent gray images where $i = 1, 2, 3, \ldots, q$, q is number of frames. Each frame is divided into 3 × 3 × 3 non-overlapping voxels. Let $n_i(x, y)$ denote the centre pixel C of the voxel. The pixels around C are neighbours or neighbourhood. In 3 × 3 × 3 voxel, n_i $(x - 1, y - 1, t)$, $n_i(x, y - 1, t)$, $n_i(x + 1, y{-}1, t)$, $n_i(x + 1, y, t)$, $n_i(x + 1, y + 1, t)$, $n_i(x, y + 1, t)$, $n_i(x - 1, y + 1, t)$ and $n_i(x - 1, y, t)$ are neighbours at t^{th} frame. Let N be the size of the neighbourhood around C and FI represent the frame interval between the frames. FI is set to 1. Every frame has meaningful information to recognize the activity because minimal discriminative frames alone are taken as input. R is the radius of the circle and is set to 1. Within the circle hexagonal neighbourhood is formed and six neighbourhood points are chosen with respect to R from the centre pixel C_{F_i} in frame F_i.

Side of the regular hexagon = Radius of the circle (Morton & Waud, 1830).

Algorithm 1. H-VLBP Descriptor

```
Input: $F_i$
Output:$FV_i$
S: number of consecutive frames considered
PreProcessing:
$M_i$ = Resize(ROI($M_{F_i}$));
$D_i$ = Discriminative_frames ($M_i$) where $3 \leqslant |D_i| \leqslant 10$
Volume Model:
[H W D] = Size(grayscale ($D_i$))
H-VLBP:
Initialize FI, R = 1, S = 3;
$H_{bs} \leftarrow 2^{(2(N+1)+N)}$;
F ← $D_i$
for m = 1; m<|F|; m++ do
    $P_r\_F_i$ ← F[m];
    $C_r\_F_i$ ← F[m + 1];
    $N_t\_F_i$ ← F[m + 2];
    FS[m] ← [$P_r\_F_i$; $C_r\_F_i$; $N_t\_F_i$]
    for i = 1: h − 1 do
        for j = 1: w − 1 do
            for k = 1: d − 1 do
                Divide FS[m] into 3 × 3 × 3 voxels
                for all 3 × 3 × 3 voxels do
                    for $n_i$ = 0: N do
                        $THLBP_{pr} \leftarrow$ T[$d(c_{pr}, C_{cr}) + \sum_{N=0}^{n_i}$ d($P_r n_i, C_{cr}$)];
                        $THLBP_{cr} \leftarrow$ T[$d(C_{cr}, C_{cr}) + \sum_{N=0}^{n_i}$ d($C_r n_i, C_{cr}$)];
                        $THLBP_{nt} \leftarrow$ T[$d(C_{nt}, C_{cr}) + \sum_{N=0}^{n_i}$ d($n_t n_i, C_{cr}$)];
                        H-VLBP$_{F_i}$ ← V [$THLBP_{pr}, THLBP_{cr}, THLBP_{pt}$];
                        H-VLBP$_{F_i}$ multiplied by weights $2^a$, where a = 0, 1, 2, ..., 3 N + 2
                        H-VLBP$^{Rt}_{N,R}$ ← min{{H-VLBP$_{P,R}$ and $2^{3N+1}$} + {H-VLBP$_{N,R}$ and 1} + ROL{ROR {$HLBP_{pr,N}$, 2 N + 1}} + ROL{ROR{$HLBP_{cr,N}$, N + 1}} + ROL{ROR{$HLBP_{t,N}$, 1}}};
                    end for
                    $H_b$ ← H-VLBP$^{Rt}_{N,R}$
                end for
            end for
        end for
    end for
    $X_i \leftarrow \sum_{b=0}^{H_{bs}} |H_b|$;
end for
```

A hexagonal structure showing the vertices is given in Fig. 4 and the points on the circle at the angles corresponding to hexagonal neighbourhood are shown in Fig. 5(b) Hexagon Neighbour points are equally spaced and connected with respect to circle centre. The spatial coordinates of the hexagonal neighbours of a pixel (x, y) at a radius R = 1 are shown in Fig. 5(c) and (d). Fig. 7 shows the H-VLBP with hexagonal-prism structure in 3D view. The H-VLBP extracted procedure with R = 1, N = 6 and FI = 1 is shown in Fig. 6.

Zhao and Pietikainen (2007) have provided the coordinates of circularly symmetric neighbour set. The equations are reproduced in Eqs. (4)–(6).

The coordinates of C_{F_i,n_i} are (x, y, t). The coordinates of current frame $C_{cr,F_i,N}$ are given by

$$C_{cr,F_i,N} = ((x + Rcos2\pi n_i/N), (y - Rsin2\pi n_i/N), t) \tag{4}$$

The coordinates of previous frame $C_{pr,F_i,N}$ are given by

$$C_{pr,F_i,N} = ((x + Rcos2\pi n_i/N), (y - Rsin2\pi n_i/N), t - 1) \tag{5}$$

The coordinates of next frame $C_{pt,F_i,N}$ are given by

$$C_{pt,F_i,N} = ((x + Rcos2\pi n_i/N), (y - Rsin2\pi n_i/N), t + 1) \tag{6}$$

N is set to 6. The angles of the hexagonal neighbours $\{0$ or $2\pi, \frac{\pi}{3}, 2\frac{\pi}{3}, \pi, 4\frac{\pi}{3}, 5\frac{\pi}{3}\}$ are obtained using Eqs. (4)–(6) by assigning the values of $n_i = 0, 1, 2, 3, 4, 5$.

The neighbouring pixel values does not always fall exactly on the location of the pixel. Those pixel values are estimated using bilinear interpolation. Bilinear interpolation performs the computation of four corner pixel values which is near to the pixel. In such cases, input is translated by 0.5 pixels to the right in positive horizontal direction. Hence, the coordinate values are translated by 0.5 pixels using Eq. (7). The coordinates of C_{F_i,n_i} are

$$\begin{aligned} C_{F_i,n_i} = {} & ((x + Rcos((2\pi n_i)/N + 0.5)), \\ & (y - Rsin((2\pi n_i)/N + 0.5)), t) \end{aligned} \tag{7}$$

To get illumination invariance, joint distribution V of the gray levels of 3 N + 3 image pixels is obtained. Table 1 summarizes the notations used in the following equations. General LBP operator (Ojala et al., 2000) is reproduced here with hexagon structure in Eqs. (9)–(11). General HLBP operator is given in (8),

$$HLBP(c) = \sum_{i=0}^{5} T(n_i - c)2^{n_i} \tag{8}$$

where $n_i = 0, 1, 2, 3, 4, 5$. The framewise H-VLBP operations are given below with respect to previous, current and next frames. In Current Frame,

$$HLBP_{cr}(c) = \sum_{i=0}^{5} T(C_r n_i - c)2^{n_i} \tag{9}$$

In Previous Frame,

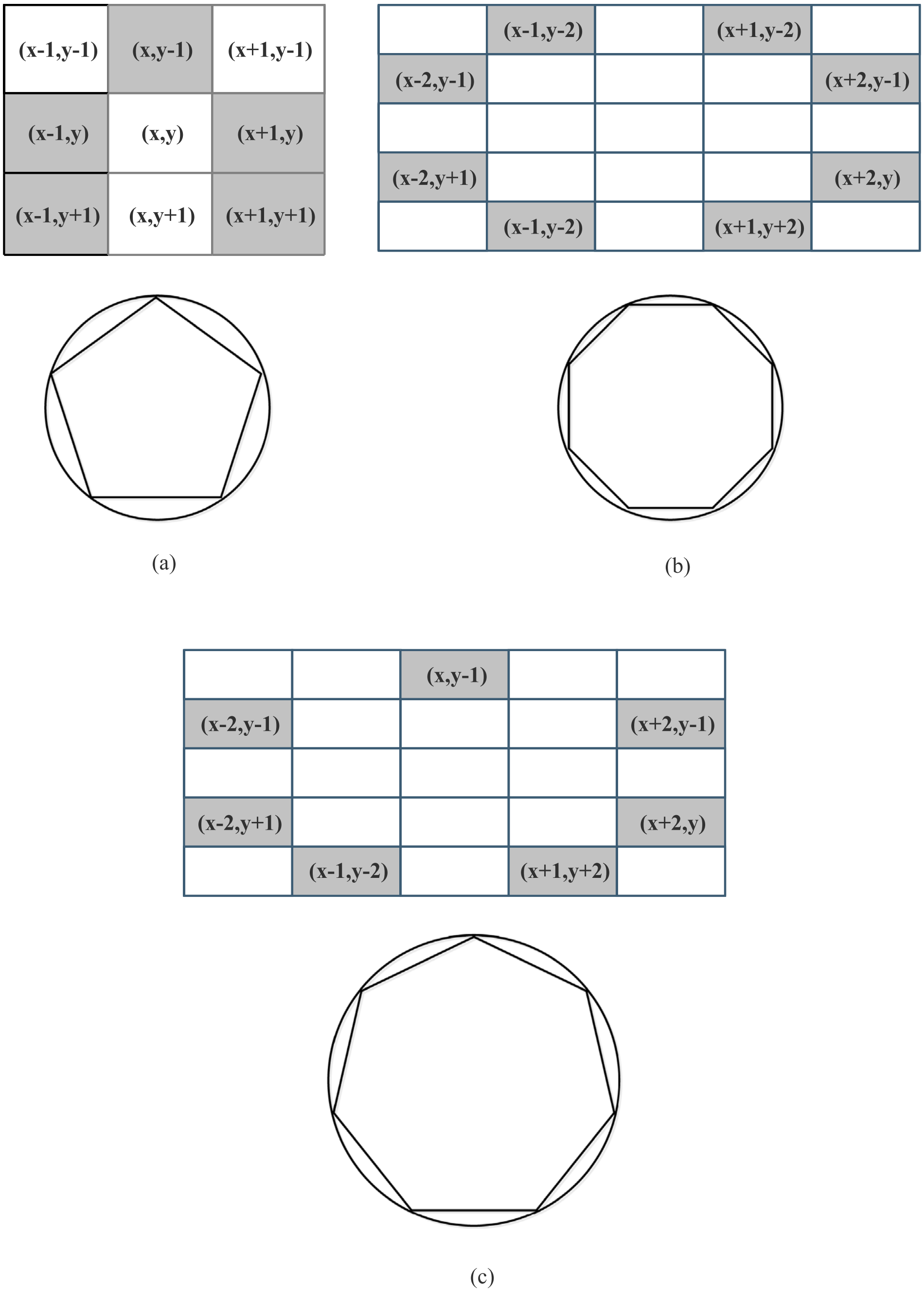
$$HLBP_{pr}(c) = \sum_{i=0}^{5} T(P_r n_i - c)2^{n_i} \tag{10}$$

In next Frame,

$$HLBP_{nt}(c) = \sum_{i=0}^{5} T(n_t n_i - c)2^{n_i} \tag{11}$$

Thresholding the difference between centre pixel and its neighbours are given in Eqs. (12)–(14).

$$THLBP_{pr} = T[d(c_{pr}, C_{cr}) + \sum_{i=0}^{5} d(P_r n_i, C_{cr})] \tag{12}$$



(a)

(b)

(c)

Fig. 4. Neighbour selection topologies. (a) Pentagon (b) Octagon (c) Heptagon.

$$THLBP_{cr} = T\left[d(C_{cr}, C_{cr}) + \sum_{i=0}^{5} d(C_r n_i, C_{cr})\right] \tag{13}$$

$$THLBP_{nt} = T\left[d(C_{nt}, C_{cr}) + \sum_{i=0}^{5} d(n_t n_i, C_{cr})\right] \tag{14}$$

where

$$T(x) = \begin{cases} 1, & \text{if } x \geqslant 0 \\ 0, & \text{else} \end{cases}$$

where x represents the difference between the neighbour pixels and centre pixel. d represents difference between two pixels. T represent for threshold. Finally,

$$\text{H-VLBP}_{F_i} = V[THLBP_{pr}, THLBP_{cr}, THLBP_{nt}] \tag{15}$$

Eq. (15) represents the joint distribution of THLBP_{pr}, THLBP_{cr} and THLBP_{nt} binary patterns. H-VLBP_{F_i} is multiplied by weights 2^i, where $\text{i} = 0, 1, 2, \ldots + 3\text{ N} + 2$.

$\text{H-VLBP}_{N,R}$ represent the hexagonal VLBP with N neighbours and radius R.

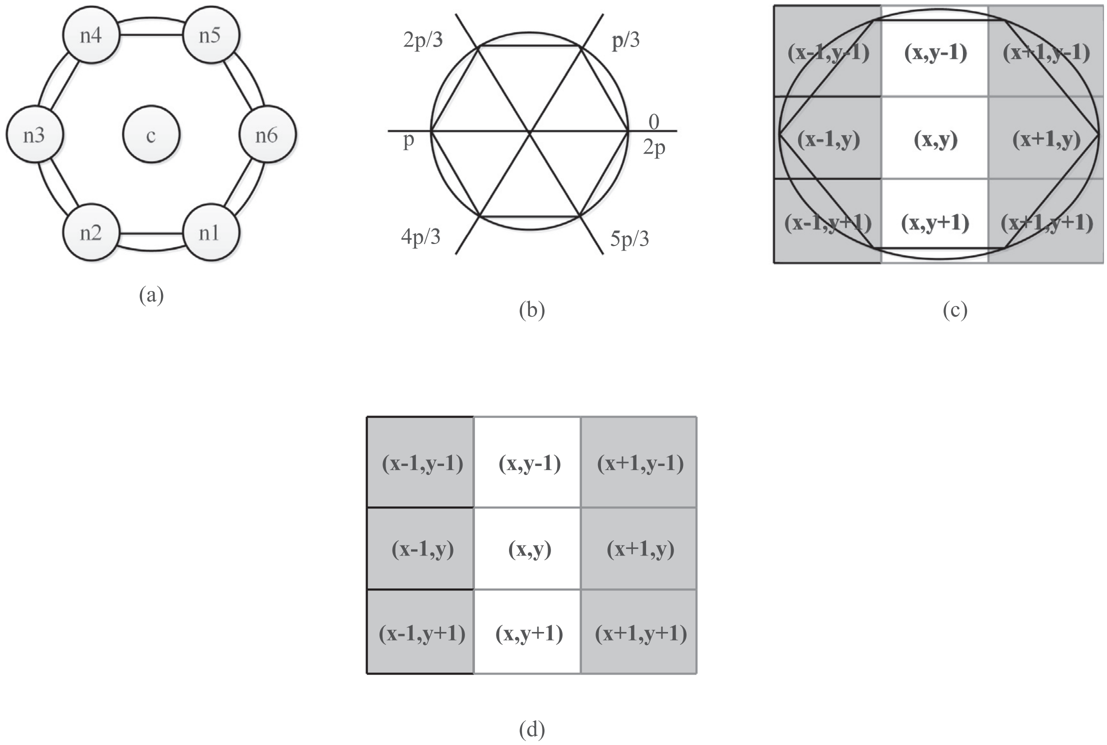


Fig. 5. Hexagonal neighbourhood. (a) Vertices in the hexagonal structure, (b) hexagonal neighbourhood structure with angle, (c) and (d) represents spatial coordinates of the hexagonal neighbours of a pixel (x, y) at a radius R = 1.

Feature count of six point neighbour selection is calculated using Eq. (16).

$$\text{Dimensionality of H-VLBP}(H_{bs}) = 2^{(2(N+1)+N)} \qquad (16)$$

where N = 6. Final estimated histogram is normalized using Eq. (17). X_i refers to feature vectors. Normalized histogram converted to single row feature vectors $[1 \times X_i]$.

$$X_i = \sum_{b=1}^{H_{bs}} |H_b| \qquad (17)$$

H_b is histograms resultant with H-VLBP. X_i is normalized to dimensionality of H-VLBP.

3.2.4. Rotation invariant H-VLBP

VLBP is not rotation invariant. Bitwise clockwise and anticlockwise shift on the 6-bit binary pattern has been used to overcome the limitation of variance to rotation. Eq. (18) is used to achieve the rotation invariance property (Ojala et al., 2000, 2002; Zhao & Pietikainen, 2007). The rotation invariance H-VLBP,

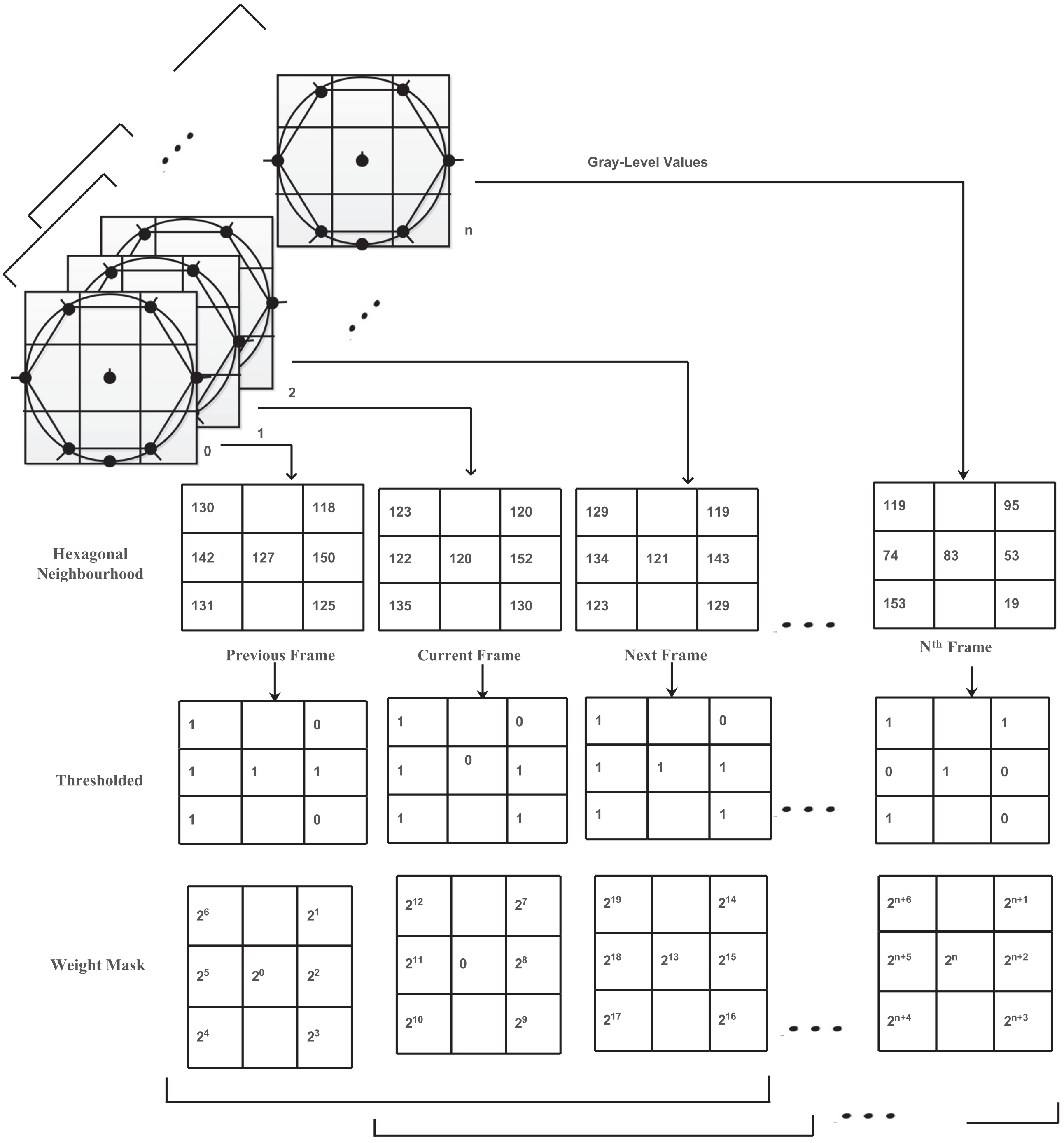
$$\begin{aligned} \text{H-VLBP}^{Rt}_{N,R} = \; & min\{\{\text{H-VLBP}_{N,R} and 2^{3N+1}\} \\ & +\{\text{H-VLBP}_{N,R} and 1\} \\ & +ROL\{ROR\{HLBP_{pr,N}, 2N+1\}\} \\ & +ROL\{ROR\{HLBP_{cr,N}, N+1\}\} \\ & +ROL\{ROR\{HLBP_{nt,N}, 1\}\}\} \end{aligned} \qquad (18)$$

where ROR represent the bit-wise clockwise shift rotation and ROL represent the bit-wise anticlockwise shift rotation. 2^{3N+1}, 1, 2 N + 1 and N + 1 represent the number of shifts. N is the number of neighbours and is set to 6. R is radius and is set to 1.

The six point neighbourhood selection can result in various unique rotation patterns. From these pattern, the circularly symmetric hexagon neighbour set, $\text{H-VLBP}^{rt}_{6,1}$ is considered. The rotation is done until a match is found in the set of unique patterns. The resultant matched pattern is taken as the feature vector in H-VLBP.

3.3. Multi class human action recognition using the deep stacked autoencoder

The dimensionality of H-VLBP $(2^{(2(N+1)+N)})$ is large. Hence, storing and managing the feature vectors are difficult. In order to overcome this issue, the dimensionality reduction technique is used. The popular dimensionality reduction techniques that are used in literature are Principle Component Analysis (PCA) (Sorzano, Vargas, & Montano, 2014), Independent Component analysis (ICA) (Van Der Maaten, Postma, & Van den Herik, 2009), Local linear embeddings (Wang & Sun, 2015), Isomap (Van Der Maaten et al., 2009), and so on. Most of the developed methods work effectively in reducing the dimensionality of feature vectors. Recently, autoencoders are being used in solving dimensionality reduction problems. Autoencoders or auto associative neural network has been introduced by Sisodiya. In this paper, the deep stacked autoencoder is used for dimensionality reduction and the output layer is activated by softmax function for multi class human action recognition.



$$\text{H\_VLBP Code} = 2^0+2^2+2^4+2^5+2^6+2^8+2^9+2^{10}+2^{11}+2^{12}+2^{15}+2^{16}+2^{17}+2^{18}+2^{19}+\,.....+2^{n+n}$$

Fig. 6. H-VLBP Procedure with R = 1, N = 6 and FI = 1.

A stacked autoencoder is a neural network consisting of multiple layers of sparse autoencoders. The input data is fed to the hidden layers and trained in an unsupervised manner. The outputs of each layer is wired to the inputs of the successive layer. Any autoencoder with more than three layers (one input layer + one hidden layer + one output layer) is called the deep stacked autoencoder. The total number of input sequences taken in this experiment is partitioned into three sets in the ratio of 70:15:15 for training, validation and testing set. Initially, training dataset is used to train the deep stacked autoencoder. Determining the number of neurons in the hidden layers should be two-thirds the size of the input neurons. There is no theoretical reason to use more than two hidden layers. In this work,

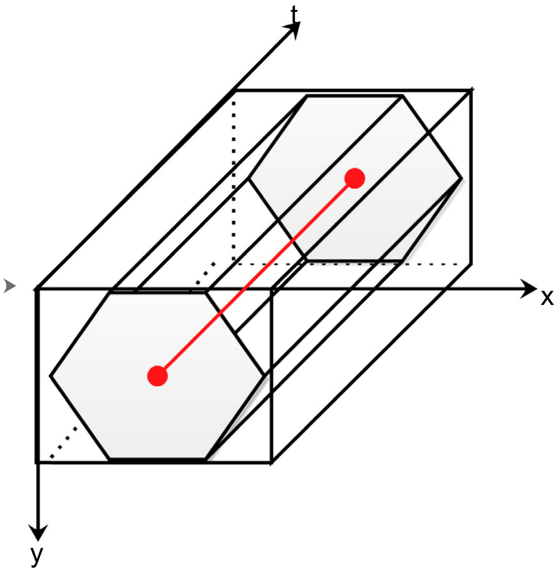


Fig. 7. H-VLBP with hexagon in 3D view: Hexagonal Prism.

Table 1
Notations and their explanations.

Notations	Explanations
n_i	Neighbours, where i = 0, 1, 2, 3, 4, 5
N	Total number of neighbours
C	Centre pixel
T	Threshold
2^n	Weight updation where n = 0, 1, 2, 3, 4, 5
C_rn_i	Neighbours of current frame
P_rn_i	Neighbours of previous frame
n_tn_i	Neighbours of next frame
C_{pr}	Center pixel in the previous frame
C_{cr}	Center pixel in the current frame
C_{nt}	Center pixel in the next frame

the objective of using the deep stack autoencoder is to reduce the dimensionality of the input feature vector. In this work, the number of hidden layers varies from 2 to 7. The number of hidden layers increases that resultant with reduced feature vector size. The output layer is trained in the n-dimensional feature vectors by softmax function in a supervised manner. The softmax layer size is given by the user and it is of the same size as that of the targets.

Training the deep stacked autoencoder:

Let I representing the feature vector $X = (X_i)^T$ where $i = 0, 1, 2, \ldots, z$ extracted by H-VLBP be wired to the encoder. z represents the number of examples in the training set. The sigmoid activation function is defined by the Eq. (19)

$$Sigmoid\ activationf(x) = \frac{1}{1 + \exp^{-x}} \tag{19}$$

The encoder consists of several hidden layers. The number of hidden layer and their neuron size is specified by the user. The neuron size should be 70–100 % of input size. In the encoder part, each hidden layer output is given to the successive layers. In the deep stacked autoencoder, single complete presentation of the dataset to be learned is named as an epoch or iteration. Encoder transforms the input feature vector into new feature representation. The first, second and third hidden layers have different number of hidden neurons based on input neurons. The user-defined parameter values are as follows: maximum epochs varied between 100 and 1000, L2 weight regularization varied between 0.001 and 0.01, sparsity regularization varied between 1 and 4, and sparsity proportion varied between 0.15 and 0.65. All values are obtained experimentally.

Softmax layer:

The last layer is a softmax layer trained to generate multi class classification of human actions. It is fully connected and the maximum epochs is set to 1000. Then deep stacked autoencoder are trained by back propagation technique to minimize the recognition error.

The softmax activation function is defined by Eq. (20)

$$Softmax\ activationf(x_j) = \frac{e^{x_j}}{\sum_{k=1}^{k} e^{x_k}} \tag{20}$$

4. Experiments and results

This paper reports the outcome of the proposed approach on three datasets. The proposed approach is compared with related works to show the effect of the H-VLBP descriptor. All the experiments are carried out using the Windows 8 environment over Intel(R) Core(TM) i7-4790 CPU processor with the speed of 3.60 GHz and 14 GB RAM.

4.1. KTH dataset

4.1.1. Dataset

KTH dataset (Guo et al., 2017; Selmi, El-Yacoubi, & Dorizzi, 2016) is one of the most popular benchmark datasets for action recognition as reported in most of the HAR

Table 2
Characteristics of the datasets used in this work.

S. No.	Characteristics	KTH dataset	Weizmann dataset	UCF11 dataset	IXMAS	Synthetic dataset
1	Activities	6	10	11	13	6
2	Total Clips	600	90	1588	2340	88
3	Video format	Mpeg-4	Mpeg-4	Mpeg-4	Mpeg-4	Mpeg-4
4	Frame rate	25 fps	50 fps	30 fps	23 fps	25 fps
5	Resolution	160 × 120	180 × 144	320 × 240	390 × 291	352 × 288
6	Number of clips per activity	25 clips	10 clips	25 sets	13 sets	24 clips

literatures. It contains 6 actions. Table 4 shows the action names and indexes of the KTH dataset. Each action is performed 3–5 times by 8 different actors in different scenarios. The scenarios of KTH datasets are shown in Table 3. Examples of actions in KTH dataset are shown in Fig. 8. Each action contains 300 to 400 frames per video approximately. In this large number of frames, the same activity has been performed by an actor 3 to 4 times. Instead of applying the recognition method directly to the complete video frame, the discriminative frames per action are selected. The total number of sequences taken to process the proposed approach is 504. The number of sequences taken in each action is 104 sequences for boxing action, 66 sequences for hand clapping, 68 sequences for hand waving, 80 sequences for jogging, 74 sequences for running and 110 sequences for walking. ROI has been selected and cropped for effective analysis. Each cropped frame is resized for spatial normalization and converted to grayscale frames for further work. Each sequence contains a maximum 10 frames.

4.1.2. Experimental setup

In this experiment, gray values of the D_{F_i} are modelled as the volumetric data model or 3D model. The dimensions of the volume are H × W × D. Spatially normalized video

Table 4
Action class names and indexes in KTH dataset.

Index	Action name	Index	Action name
1	Boxing	4	Jogging
2	Hand Clapping	5	Running
3	Hand Waving	6	Walking

Table 3
Datasets and their scenarios.

Dataset	Scenarios	Dataset	Scenarios
KTH	Static homogeneous background Outdoor Outdoor with scale variations Different clothes	Synthetic	Static homogenous background Outdoor Outdoor with Scale variations Different viewing angle Different resolutions Variations in Action capturing Different Clothes
Weizmann	Static homogenous background Outdoor Different Clothes	UCF11	Camera motion Object appearance and pose variant cluttered background Illumination variations Object scale and viewpoint invariant
IXMAS	Static homogenous background Indoor Multi-view point Illumination variations		

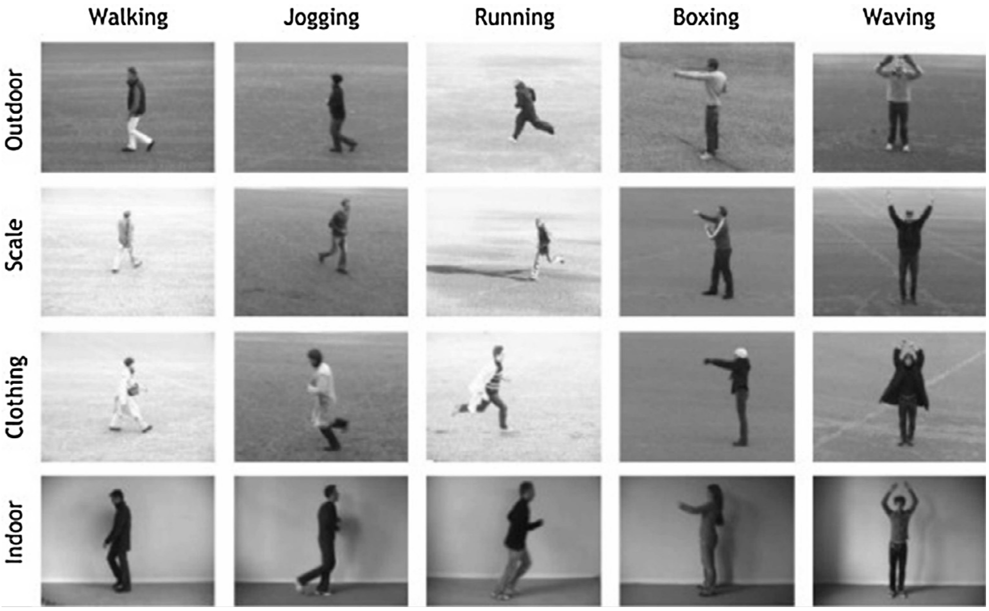


Fig. 8. Examples of actions in KTH dataset.

sequences contain $100 \times 100 \times D_i$ of Volume dimensions. In the H-VLBP approach, the gray value of voxel which is connected within the rectangular prism are taken for processing. As mentioned in Section 3.2.4, by default FI and R are set to 1. The circle and drawn using Eqs. (4)–(6) and the hexagon angles $\{0 \text{ or } 2\pi, \frac{\pi}{3}, \frac{2\pi}{3}, \pi, \frac{4\pi}{3}, \frac{5\pi}{3}\}$ are taken as neighbourhood of 3×3 cell's centre pixel. D_{F_i} is taken as overlapping continuous three frames up to the end of the sequence. Each frame is divided into non-overlapping $3 \times 3 \times 3$ voxels. Each $3 \times 3 \times 3$ voxels are represented as previous, current and next frame. The process will continue until the end of the frame and end of the D_{F_i}. Finally, 2^i different unique patterns are converted to a binary code using Eqs. (12)–(14). The threshold of difference values of neighbours and centre is multiplied with 2^i weighted values. The conversion from H-VLBP to rotation invariant H-VLBP is performed using the bit wise shift transformation expressed in Eq. (18).

The extracted feature size of H-VLBP is $2^{(2(N+1)+N)}$. The deep stacked autoencoder is used to reduce the dimensionality of the feature vector. In KTH dataset, the total number of video sequences taken in this experiment is 504. It is partitioned into three sets in the ratio of 70:15:15 for training, validation and testing sets. 352 videos are used as the training dataset, 76 sequence are taken as the validation set and 76 sequence are taken as the testing set for the deep stacked autoencoder. The deep stacked autoencoder consist of more than one hidden layer. The training samples with the size of 352 samples are wired to the first hidden layer (HL_1) with size of 245 neurons. (HL_1) is wired to second hidden layer (HL_2) with size of 170 neurons. 120 and 84 neurons are used in the third hidden layer (HL_3) and fourth hidden layer (HL_4) respectively. The model is trained for 500 epochs using the conjugate gradient descent algorithm and evaluated on test dataset. User defined parameter values are initialized as follows: L2 weight regularization is set to 0.001, Sparsity regularization is set to 1, and sparsity proportion is set to 0.15. Finally, a softmax layer is used for multi-class classification.

4.1.3. Comparison with related works

In most of the literatures, all the extracted frames have been used for feature extraction in human action recognition (Ahsan et al., 2014; Selmi et al., 2016). Some researchers have manually cropped the frames (Chun & Lee, 2016). Key frame extraction (Sheena & Narayanan, 2015) have been used in action recognition to reduce the time and space complexity. Key frame extraction has obtained repetitive frames and the reduced number of frame count is just half the total count. Hence, in this work, the minimum number of discriminative frames is used to reduce the time and space complexity. The comparison is done with methods which uses all the frames as input. The computational time of preprocessing in H-VLBP approach using KTH dataset is shown in Table 9.

The confusion matrix of the proposed approach on the KTH dataset is shown in Table 12. The vertical axis represent the actual class label and the horizontal axis represents the distribution of the predicted label. Higher diagonal values are related to correct classification. The diagonal values are shown in bold in Table 12. From the confusion matrix, it can be seen that there are some mistaken recognitions, because of some similarity between running, jogging and walking. The average recognition accuracy of the proposed approach is 97.6 % which is higher than the other state-of-art methods. The experimental results show significant improvement as compared with existing methods as indicated in Table 18.

4.2. Weizmann dataset

4.2.1. Dataset

The Weizmann dataset (Al-Berry et al., 2016; Guo et al., 2017) is a widely used benchmark dataset for HAR. It contains 10 actions. Table 5 shows the action names and indexes of Weizmann dataset. Each action is performed 3 to 5 times by 9 different actors with different scenarios. The scenarios of Weizmann are shown in Table 2. Examples of actions in Weizmann dataset are shown in Fig. 9. The total number of sequences used to process the proposed approach is 180. 18 sequences were taken for bend action, 18 sequences for Jack, 16 sequences for wave1, 16 for wave2, 20 for Pjump, 18 for skip1, 18 for run, 20 for side, 18 for jump and 18 sequences for Walk. ROI has been selected and cropped for effective analysis with gray level frames and normalized spatial information. The maximum number of frame in a sequence is 10.

4.2.2. Experimental setup

All of the parameter values were kept the same as those used in KTH experiment. Default values are initialized and H-VLBP feature extraction has been performed. The representation described in Section 3.2 is given as input to a stacked autoencoder to reduce the dimensionality of the

Table 5
Action class names and indexes in Weizmann dataset.

Index	Action name	Index	Action name
1	Bend	6	Gallop-side-ways(side)
2	Jumping-Jack(or jack)	7	Skip
3	Jump-forward-on-two-legs(or Jump)	8	Walk
4	Jump-in-place-on-two-legs(pjump)	9	Wave-one-hand (Wave1)
5	Run	10	Wave-two-hands (or wave2)

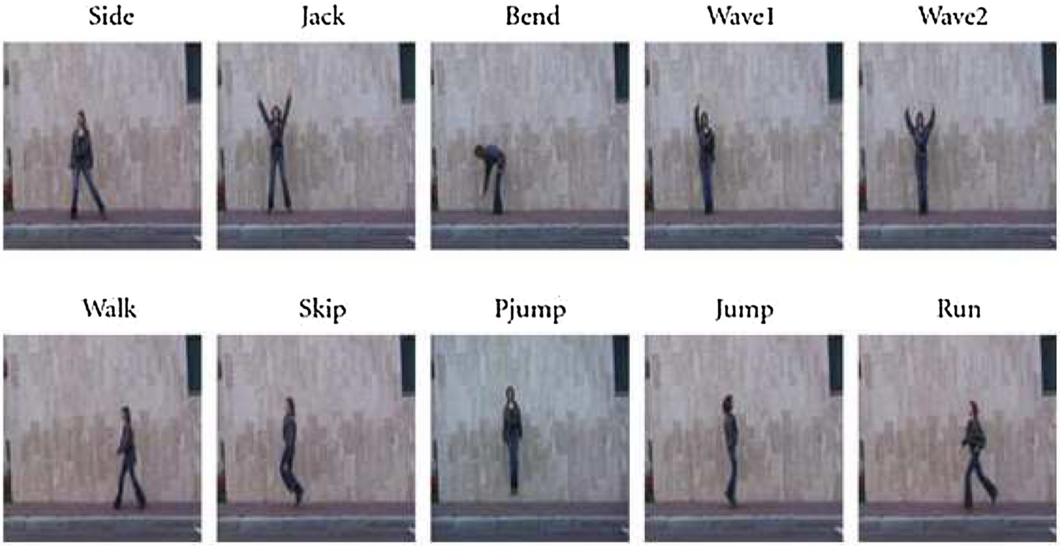


Fig. 9. Examples of actions in Weizmann dataset.

feature vector. In Weizmann dataset, the total number of video sequences taken in this experiment is 180. It is partitioned into three sets in the ratio 70:15:15. 126 input sequences are used as the training dataset, 27 sequences are taken as the validation set and 27 sequences taken as the testing set for the deep stacked autoencoder. The 126 training samples are wired to the first hidden layer of the stacked autoencoder which is trained to learn the features necessary to reconstruct the input into latent code with reduced feature vector. (HL_1) is trained with 88 neurons. A second hidden layer is trained to reconstruct the features learned by the first hidden layer. Hence, (HL_1) is wired to (HL_2) with 62 neurons. The model is trained for 300 epochs using conjugate gradient descent algorithm and evaluated on the test dataset. Finally, a softmax layer is used for multi-class classification.

4.2.3. Comparison with related works

The confusion matrix of the proposed approach on Weizmann dataset is shown in Table 13. The predicted recognition accuracy values are highlighted in bold font. From the confusion matrix, it can be seen that there are some mistaken recognitions, because of some similarity actions with locomotion changes. The average recognition accuracy of the proposed approach is 98.6% which is higher than other state-of-art methods. The comparison among the proposed approach and other state-of-art action recognition approach is shown in Table 19. It can be inferred that the proposed approach outperforms the approaches used by Guo et al. (2017) and Ahsan et al. (2014) in Weizmann dataset.

4.3. UCF11 dataset

4.3.1. Dataset

UCF11 dataset is a widely used benchmark dataset for HAR. It contains 11 actions. Table 6 shows the action names and indexes of the UCF11 dataset. The scenarios of UCF11 are shown in Table 2. Examples of actions in UCF11 dataset are shown in Fig. 10. The total number of sequences used to process the proposed approach is 1588. 143 sequences were taken for basketball shooting action, 135 sequences for biking or cycling, 156 sequences for driving, 143 for golf swinging, 198 for horseback riding, 156 for soccer juggling, 137 for swinging, 167 for tennis swinging, 115 for trampoline jumping, 115 sequences volleyball spiking and 123 sequences for walking with a dog. ROI has been selected and cropped for effective analysis with gray level frames and normalized spatial information. The maximum number of frame in a sequence is 10.

Table 6
Action class names and indexes in UCF11 dataset.

Index	Action name	Index	Action name
1	Basketball shooting	7	Swinging
2	Biking/Cycling	8	Tennis swinging
3	Driving	9	Trampoline jumping
4	Golf swinging	10	Volleyball spiking
5	Horse back riding	11	Walking with a dog
6	Soccer juggling		

4.3.2. Experimental setup

The H-VLBP feature vectors have been computed. In order to reduce the dimensionality of the feature vector, the deep stacked autoencoder has been used. The deep stacked autoencoder consists of more than one hidden layer. In UCF dataset, the total number of video sequences taken in this experiment is 1588. It is partitioned into the training set, validation set and testing set in the ratio 70:15:15. 1112 input sequences are used as the training dataset, 238 sequences are taken as the validation set and 238 sequences are taken as the testing dataset for the deep stacked autoencoder. The input samples with a size of 1112 samples are wired to (HL_1) with size of 778 neurons. (HL_1)

Fig. 10. Examples of actions in UCF11 dataset.

is wired to (HL_2) with the size of 545 neurons. In this study, 382, 267, 187, 131 and 92 neurons are used in (HL_3), (HL_4), fifth hidden layer (HL_5), sixth hidden layer (HL_6) and seventh hidden layer (HL_7) respectively. The model is trained for 500 epochs using conjugate gradient descent algorithm and evaluated on test dataset.

4.3.3. Comparison with related works

The confusion matrix of the experiment carried out on UCF11 dataset using the proposed approach is shown in Table 14. The average recognition accuracy of UCF11 dataset is 91.3%. Comparison with other related works is shown in Table 20.

4.4. IXMAS dataset

4.4.1. Dataset

INRIA Xmas Motion Acquisition Sequences (IXMAS) dataset contains 13 daily life actions performed by 11 persons. Each action is performed 3 times. The 13 actions and their indexes are shown in Table 7. Examples of IXMAS dataset actions are shown in Fig. 11. The total number of sequences used to experiment the proposed approach is 1514. 120 sequences were taken for check watch action, 128 sequences for cross arms, 134 sequences for scratch head, 118 sequences for sit down, 98 sequences for get up, 110 sequences for turn around, 112 sequences for walk, 104 sequences for wave, 121 sequences for punch, 128 sequences kick, 102 sequences for point, 135 sequences for pickup and 104 sequences for throw. ROI has been selected and cropped for effective analysis with gray level frames and normalized spatial information. The maximum number of frames in a sequence is 10.

Table 7
Action class names and indexes in IXMAS dataset.

Index	Action name	Index	Action name
1	Check watch	8	Wave
2	Cross arms	9	Punch
3	Scratch head	10	Kick
4	Sit down	11	Point
5	Get up	12	Pickup
6	Turn around	13	Throw
7	Walk		

4.4.2. Experimental setup

In the IXMAS dataset, the total number of video sequences taken in this experiment is 1514. It is partitioned into the training set, validation set and testing set in ratio 70:15:15. 1135 input sequences are used as the training dataset, 227 sequences are taken as the validation set and 227 sequences are taken as the testing dataset for the deep stacked autoencoder. The input samples with a size of 1135 samples are wired to (HL_1) with size of 801 neurons. First hidden layer (HL_1) is wired to second hidden layer (HL_2) with the size of 568 neurons. In this study, 405, 290, 210, 154 and 115 neurons are used in third hidden layer (HL_3), fourth hidden layer (HL_4), fifth hidden layer (HL_5), sixth hidden layer (HL_6) and seventh hidden layer (HL_7) respectively. The model is trained for 500 epochs using conjugate gradient descent algorithm and evaluated on test dataset.

4.4.3. Comparison with related works

The confusion matrix shown in Table 15 is obtained using H-VLBP feature vectors to recognize the IXMAS dataset action sequences. The average recognition accuracy of IXMAS dataset is 88.76%. Comparison with other related works is shown in Table 21.

4.5. Synthetic dataset

4.5.1. Dataset

Synthetic dataset is acquired using the Nikon D3400 DSLR Camera with a single lens for different types of jumping actions. It contains five actions. Table 8 shows the action names and indexes of the Synthetic dataset. Example actions are shown in Fig. 12. Each action is performed single time per video by 6 different actors in differ-

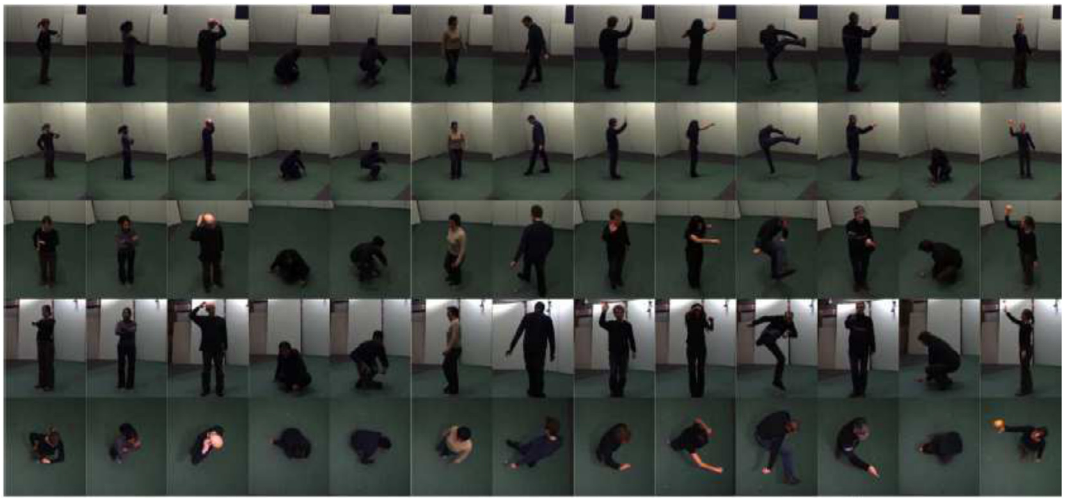


Fig. 11. Examples of actions in IXMAS dataset.

Table 8
Action class names and indexes in synthetic dataset.

Index	Action name	Index	Action name
1	Wall Jumping	4	Building to building entry
2	Walking from Building to Building	5	Building to Building Jumping
3	Entering via roof		

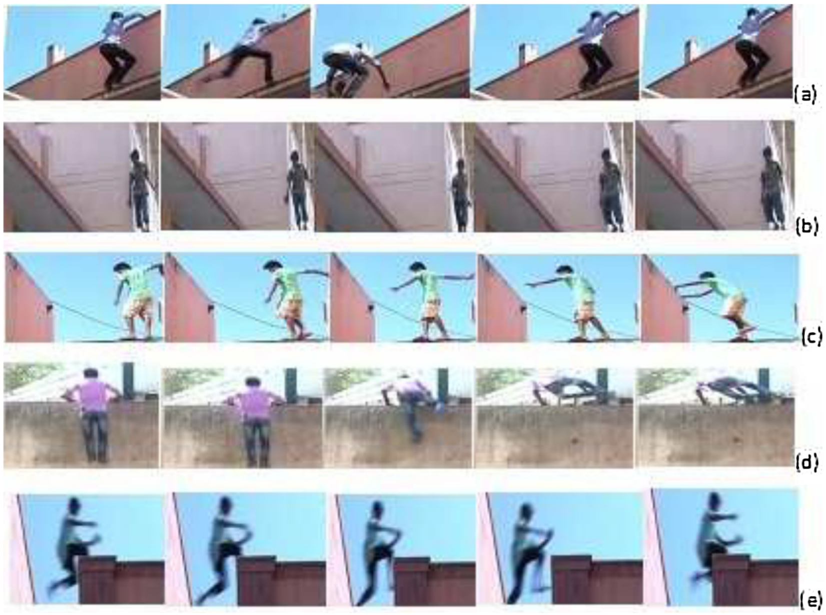


Fig. 12. Examples of actions in synthetic dataset. (a) Wall Jumping (b) Walking from Building to Building (c) Entering via roof (d) Building to building entry (e) Building to Building Jumping.

ent environment. The different scenarios are shown in Table 2. 154 sequences of action are considered in this work. 50 sequences were used for wall Jumping, 24 sequences for walking from building to building, 28 sequences for entering via roof, 18 for entering from building to building and 34 for building to building jumping. The maximum number of frames in a sequence is 10. Gray-level resized ROI frames are extracted and discriminative frames for each action have been selected. These discriminative frames have been used for H-VLBP.

4.5.2. Experimental setup

The H-VLBP feature vectors have been computed as was done with the benchmark dataset. In synthetic dataset, the total number of video sequences taken in this experiment is 154. It is partitioned into three sets in the ratio 70:15:15. 108 input sequences are used as the training dataset, 23 sequences are taken as the validation set and 23 sequences are taken as the testing dataset for the deep stacked autoencoder. The 108 input samples are wired to (HL_1) with 76 neurons. (HL_1) is wired to (HL_2) with the size of 53 neurons. (HL_2) is wired to the 37 neurons of (HL_3). The model is trained for 300 epochs using the conjugate gradient descent algorithm and evaluated on the test dataset. Finally, a softmax layer is used for multi-class classification.

4.5.3. Results

The confusion matrix shown in Table 16 is obtained using H-VLBP feature vectors to represent the synthetic action sequences. B2B represent the building to building. Enter-roof represent the action of entering through the roof. The greatest confusion is between building to building entry and building to building jumping. For some actors, building to building jumping is very similar to building to building entry and vice versa. The recognition accuracy of synthetic dataset is 90.9%.

5. Discussion

The proposed approach is an extended version of VLBP with the circularly symmetric neighbourhood. The drawback of the circularly symmetric neighbourhood and proposed H-VLBP approach is the length of the feature vector and the time complexity involved in feature extraction. In this paper, to overcome the dimensionality problem, the deep stacked autoencoder is used to reduce the dimension of the feature vector. The research additionally focuses on minimizing the time to extract the feature vectors and improving the accuracy of action recognition. Hence, in this work discriminative frames per action are used, thereby reducing the time complexity. The performance of VLBP features is very sensitive to neighbourhood topology and the number of connected neighbours. The feature vector length of VLBP method using different geometric neighbourhood topologies are shown in Table 10. Deep stacked autoencoder has been already discussed in Section 3.3. In this discussion, the effect of the number of hidden layer on recognition accuracy has been analysed.

Effects of H-VLBP approach and its Recognition accuracy in KTH, Weizmann, UCF11, IXMAS and synthetic dataset using different classifier are shown in Table 17. From the experiments two layer FFNN performs well and provides better results. The choice of user specified parameters, namely, the number of hidden neurons and the number of epochs for the two layer Feed Forward Neural Network (FFNN) are discussed below.

Two layer Feed Forward Neural Network:

Two-layer FFNN consist of a series of two layers, namely, the hidden layer and the output layer. Initially, the input data are partitioned into training data, validation data and testing data. The training data will adjust the weights on the neural network. The control of learning process and overfitting minimization is handled by validation data. Finally, the quality evaluation of the learning is done using testing data. The testing quality is measured in terms of cross-Entropy (CE) and Percent Error (%E). CE is the error rate of misclassified data. The %E is the fraction of misclassified samples. Small values of CE and %E indicate good classification performance.

In KTH dataset, the total number of video sequences taken in this experiment is 508. It is partitioned into three sets in the ratio 70:15:15. 352 videos are used as the training dataset, 76 sequences are taken as the validation set and 76 for the testing the multi-class actions. In Weizmann dataset, the total number of Video sequences taken in this experiment is 180. It is partitioned into 126 sequences were used as the training dataset, 27 sequences as the validation set and 27 for testing the proposed approach. In UCF11 dataset, the total number of video sequences taken in this experiment is 1588. It is partitioned into 1112 sequences and used as the training dataset, 238 sequences as the validation set and 238 for testing the proposed approach. In IXMAS dataset, the total number of video sequences taken in this experiment is 1135. It is partitioned into training dataset comprising of 851 sequences, validation set comprising of 170 sequences and testing set comprising of 170 sequences. In synthetic dataset, the total number of video sequences taken in this experiment is 154. It is partitioned into 108 sequences were used as the training dataset, 23 sequences as the validation set and 23 for testing the proposed approach.

The number of hidden neurons versus cross-entropy (CE) of training, validation and testing are shown in

Table 9
Computation time of preprocessing in H-VLBP approach for a sample video sequence from KTH dataset.

Elapsed time	Number of frames	Method
98.25 s	360	Sample video sequence
36.46 s	132	Key frames
42.63 s	12	D_{F_i}
2.49 s	12	$ROI(D_{F_i})$
2.07 s	12	$Resize(D_{F_i})$

Table 10
Feature Length of VLBP method.

Neighbourhood topology	Neighbourhood points	Feature length
Triangle	3	256
Quadrilateral	4	16384
Pentagon	5	131072
Hexagon	6	1048576
Heptagon	7	8388608
Octagon	8	67108864

Figs. 13–15. The line and dashed line represent the CE values at epoch j and epoch k respectively. The CE values epoch j and epoch k go down as more neurons are added to the model. It starts to go up sharply after 10 possibly indicating over fitting. From the analysis between the various number of hidden neurons and cross-entropy, the number of hidden neurons used in this work is 10 since it minimizes the cross-entropy values. In this work, the maximum training epoch is set between 100 and 500. Fig. 16 shows the recognition accuracy against the number of epochs for the KTH dataset, Weizmann dataset, UCF11 dataset, IXMAS dataset and the synthetic dataset. The highest accuracy on the test frame sequences was found to be 97.3 % in KTH dataset at epoch 97, 96.3 % in Weizmann dataset at epoch 93, 90.2% in UCF11 dataset at epoch 98, 84.52% in IXMAS dataset at epoch 93 and 90.9% in synthetic dataset at epoch 97.

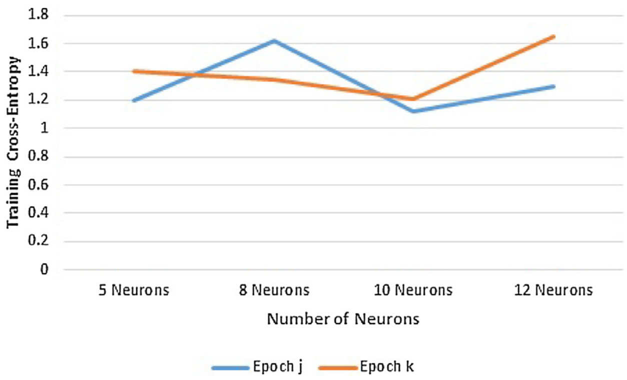


Fig. 13. The Number of hidden neurons versus training Cross-Entropy (CE) error.

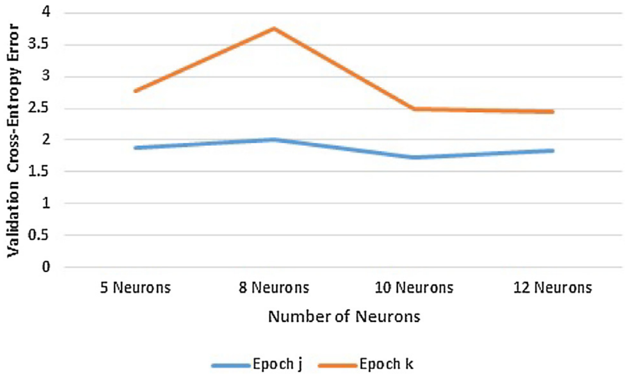


Fig. 14. The Number of hidden neurons versus validation Cross-Entropy (CE) error.

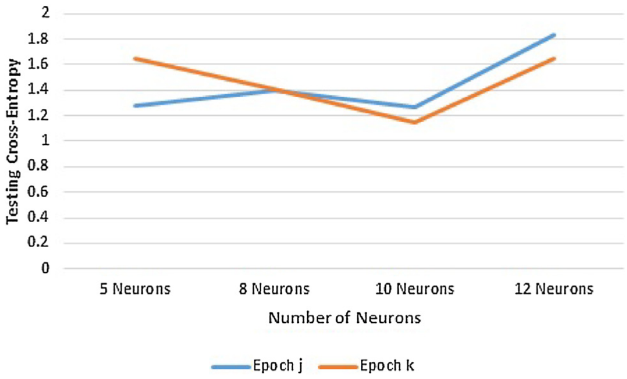


Fig. 15. The Number of hidden neurons versus testing Cross-Entropy (CE) error.

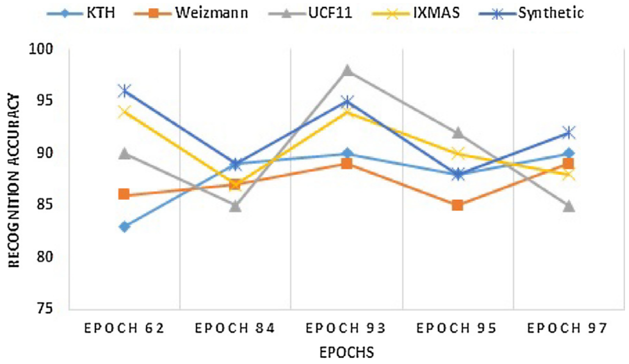


Fig. 16. Epoch versus Recognition accuracy for KTH, Weizmann, UCF11, IXMAS and synthetic dataset.

The effects of various geometric shapes and their recognition accuracy using deep stacked autoencoder vs two layer feed forward neural network are shown in Table 11. The experiments have been performed on four different types of human action dataset. From the results, it can be concluded that:

1. Discriminative frame selection improves the effectiveness of human action recognition and is used to minimize the time and space complexity.
2. H-VLBP performs better than VLBP for feature extraction for HAR. This is because of the hexagonal neighbourhood selection.
3. Dimensionality reduction has been achieved using the deep stacked autoencoder. Additionally, the proposed work has been evaluated in two layer FFNN without dimensionality reduction.

6. Conclusion and future work

In this paper, the H-VLBP descriptor has been proposed to involve an overlapping single step size in the spatio-temporal domain with the hexagonal neighbourhood selection to accumulate the motion and temporal information from the video frames. The computation of motion and temporal information helps to distinguish similar actions. Moreover, instead of using the circular neighbourhood selection in VLBP, the hexagonal neighbourhood selection is used in this work which results in characterising the rich dynamic information such as edge, corner and so on. The resultant H-VLBP code has been computed with weights of 2^N. The resultant histogram is converted to the feature

Table 11
Effects of various geometric shapes and its Recognition accuracy using deep stacked autoencoder vs two layer feed forward neural network. a represent deep stacked autoencoder and b represents Two Layer feed forward neural network. Bold value represent accuracy of proposed method, H-VLBP.

N	a					b				
	KTH	Weizmann	UCF11	IXMAS	Synthetic	KTH	Weizmann	UCF11	IXMAS	Synthetic
3	71.4	79.5	72.4	69.0	80.1	71.9	72.3	79.6	70.1	81.5
4	80.2	80.8	86.6	72.0	81.2	79.5	80.5	75.9	79.1	80.2
5	88.1	89.3	83.6	90.5	89.2	85.7	89.3	86.9	82.1	88.1
6	**97.6**	**98.6**	**91.3**	**84.52**	**93.0**	**97.3**	**96.3**	**90.2**	**88.76**	**90.9**
7	94.0	96.4	89.3	82.3	88.1	94.0	88.1	90.5	84.1	92.1
8	89.3	90.7	88.1	81.4	86.1	86.9	88.1	90.5	84.7	91.7

Table 12
Confusion Matrix obtained by H-VLBP over the KTH dataset. Each column corresponds to the predicted category and each row corresponds to the ground truth category.

	Boxing	Handc	Handw	Jogging	Running	Walking
Boxing	**1.00**	0.00	0.00	0.00	0.00	0.00
Handc	0.00	**1.00**	0.00	0.00	0.00	0.00
Handw	0.00	0.00	**1.00**	0.00	0.00	0.00
Jogging	0.00	0.00	0.00	**0.97**	0.02	0.01
Running	0.00	0.00	0.00	0.00	**0.98**	0.02
Walking	0.00	0.00	0.00	0.03	0.00	**0.97**

Table 13
Confusion Matrix obtained by H-VLBP over the Weizmann dataset. Each column corresponds to the predicted category and each row corresponds to the ground truth category.

	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wave1	Wave2
Bend	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Jack	0.00	**0.97**	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.01
Jump	0.00	0.00	**0.95**	0.02	0.00	0.00	0.02	0.01	0.00	0.00
Pjump	0.01	0.00	0.00	**0.99**	0.00	0.00	0.00	0.00	0.00	0.00
Run	0.00	0.02	0.00	0.00	**0.95**	0.02	0.00	0.01	0.00	0.00
Side	0.00	0.01	0.00	0.00	0.02	**0.97**	0.00	0.00	0.00	0.00
Skip	0.00	0.00	0.01	0.00	0.00	0.02	**0.95**	0.02	0.00	0.00
Walk	0.00	0.00	0.00	0.00	0.00	0.00	0.01	**0.99**	0.00	0.00
Wave1	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	**0.99**	0.00
Wave2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**1.00**

Table 14
Confusion Matrix obtained by H-VLBP over the UCF11 dataset. Each column corresponds to the predicted category and each row corresponds to the ground truth category.

	shoot	spike	jump	juggle	ride	cycle	dive	swing	g_swing	t_swing	walk
shoot	**0.96**	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00
Spike	0.10	**0.84**	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.04	0.00
Jump	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Juggle	0.00	0.00	0.04	**0.89**	0.00	0.07	0.00	0.00	0.00	0.00	0.00
Ride	0.00	0.00	0.00	0.00	**0.90**	0.00	0.00	0.02	0.04	0.04	0.00
Cycle	0.00	0.00	0.00	0.02	0.00	**0.92**	0.02	0.00	0.00	0.00	0.04
Dive	0.00	0.00	0.00	0.00	0.00	0.00	**0.89**	0.01	0.03	0.07	0.00
Swing	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.93**	0.02	0.01	0.04
g_swing	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.94**	0.04	0.02
t_swing	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.04	0.04	**0.85**	0.00
walk	0.00	0.01	0.00	0.05	0.01	0.00	0.00	0.00	0.00	0.00	**0.93**

vector as a compact and discriminative representation. The deep stacked autoencoder is used to reduce the dimensionality of the feature vector and the last softmax layer is used to classify the multi-class human actions. The proposed H-VLBP is extensively evaluated on two benchmark datasets and one synthetic dataset. Experimental results show that

Table 15
Confusion Matrix obtained by H-VLBP over the IXMAS dataset. Each column corresponds to the predicted category and each row corresponds to the ground truth category.

	Check	Cross	Scratch	Sit	Get	Turn	Walk	Wave	Punch	Kick	Point	Pick up	Throw
Check	**0.80**	0.05	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.08	0.00	0.00
Cross	0.15	**0.75**	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Scratch	0.00	0.00	**0.70**	0.03	0.07	0.00	0.00	0.00	0.00	0.05	0.15	0.00	0.00
Sit	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Get	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00
Turn	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Walk	0.00	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
Wave	0.00	0.00	0.00	0.00	0.00	0.08	0.05	**0.70**	0.12	0.01	0.04	0.00	0.00
Punch	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.94**	0.04	0.02	0.00	0.00
Kick	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.04	0.04	**0.85**	0.00	0.00	0.00
Point	0.00	0.01	0.00	0.04	0.01	0.00	0.00	0.00	0.00	0.00	**0.94**	0.00	0.00
Pickup	0.00	0.01	0.00	0.05	0.01	0.00	0.00	0.00	0.00	0.00	0.00	**0.93**	0.00
Throw	0.00	0.01	0.00	0.05	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.93**

Table 16
Confusion Matrix obtained by H-VLBP over the synthetic dataset. Each column corresponds to the predicted category and each row corresponds to the ground truth category.

	Wall Jump	B2B Walk	Enter-roof	B2B entry	B2B Jump
Wall Jump	**1.00**	0.00	0.00	0.00	0.00
B2B Walk	0.00	**1.00**	0.00	0.00	0.00
Enter-roof	0.00	0.00	**1.00**	0.00	0.00
B2B entry	0.00	0.00	0.00	**0.80**	0.20
B2B Jump	0.00	0.00	0.00	0.15	**0.85**

Table 17
Effects of H-VLBP approach and its Recognition accuracy in KTH, Weizmann, UCF11, IXMAS and synthetic dataset using different classifier.

Classifier	KTH	Weizmann	UCF11	IXMAS	Synthetic
SVM	90.3	94.7	84.1	83.3	81.4
Ensemble (Random subspace K-Nearest Neighbor)	93.5	96.8	88.7	81.9	80.6
FFNN	97.3	96.3	90.2	84.76	90.9

Table 18
Comparison of average recognition accuracy over the KTH dataset. Bold text and value represent method and accuracy of proposed H-VLBP approach respectively.

Method	Accuracy(%)
LBP-Top (Abdolahi et al., 2012)	77.3
DW-LBP with moments (Al-Berry et al., 2016)	96.0
LBP_H (Li et al., 2016)	73.0
3D Gradient LBP Descriptor (Guo et al., 2017)	92.25
3D CNN (Ji et al., 2013)	90.2
SDTD (Shi et al., 2016)	96.8
Online deep learning method + KNN (Charalampous & Gasteratos, 2016)	91.99
Online deep learning method + SVM (Charalampous & Gasteratos, 2016)	89.86
H-VLBP+ Two layer FFNN	**97.3**
H-VLBP+ Deep stacked autoencoder	**97.6**

the proposed approach outperforms the state-of-art methods in two benchmark datasets. Additional tests on collected synthetic dataset confirm that the H-VLBP descriptor is able to handle real-time environment which contains more challenging frame rate differences and complicated jump and walk actions in different scenarios.

In this paper, the proposed approach depends on the use of representative frames per action which overcome the limitation of number of frames, space and time complexity. For instance, it is not easy to mining a set of representative frames from videos having a large number of similar frames. The proposed approach requires an enormous amount of time and attention during the discriminative frame selection, which may be a burden when dealing with much more larger videos or frames per seconds. Hence, in the future, the work aims to explore the automatic extrac-

Table 19
Comparison of average recognition accuracy over the Weizmann dataset. Bold text and value represent method and accuracy of proposed H-VLBP approach respectively.

Method	Accuracy(%)
LBP-TOP (Kellokumpu et al., 2008)	95.6
MHI_LBP_H (Ahsan et al., 2014)	90.56
MHI_LBP_H+SF (Ahsan et al., 2014)	90.56
DMHI_LBP_H (Ahsan et al., 2014)	93.15
DMHI_LBP_H+SF (Ahsan et al., 2014)	94.26
3D Gradient LBP Descriptor (Guo et al., 2017)	92.88
Online deep learning method + KNN (Charalampous & Gasteratos, 2016)	94.7
Online deep learning method + SVM (Charalampous & Gasteratos, 2016)	100
H-VLBP+ Two layer FFNN	**96.3**
H-VLBP+ Deep stacked autoencoder	**98.6**

Table 20
Comparison of average recognition accuracy over the UCF11 dataset. Bold text and value mean method and accuracy of proposed H-VLBP approach respectively.

Method	Accuracy(%)
Online deep learning method + KNN (Charalampous & Gasteratos, 2016)	84.64
Online deep learning method + SVM (Charalampous & Gasteratos, 2016)	88.65
Average pooled + LSTM (Sharma et al., 2015)	82.56
Max pooled + LSTM (Sharma et al., 2015)	81.6
H-VLBP+ Two layer FFNN	**90.2**
H-VLBP+ Deep stacked autoencoder	**91.3**

Table 21
Comparison of average recognition accuracy over the IXMAS dataset. Bold text and value mean method and accuracy of proposed H-VLBP approach respectively.

Method	Accuracy(%)
ST-tSNE (Cheng et al., 2015)	73.64
HOMID (Chun & Lee, 2016)	83.03
MMHI (Su et al., 2016)	84.0
H-VLBP+ Two layer FFNN	**84.52**
H-VLBP+ Deep stacked autoencoder	**88.76**

tion of representative frames per action. Ongoing work focuses on further improvement by reducing the length of the feature vector using uniform multi-view orthogonal planes with geometric-shape based neighbourhood.

Declarations of interest

None.

Acknowledgment

The author expresses their sincere and reverential gratitude to University Grant Commission (UGC), India, for the obtained financial support during this research.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.cogsys.2019.03.001.

References

Abdolahi, B., Ghasemi, S., & Gheissari, N. (2012). Human motion analysis using dynamic textures. In *16th CSI international symposium on artificial intelligence and signal processing (AISP)* (pp. 151–156). IEEE.

Ahad, M. A. R., Tan, J. K., Kim, H., & Ishikawa, S. (2012). Motion history image: Its variants and applications. *Machine Vision and Applications, 23*(2), 255–281.

Ahsan, S. M. M., Tan, J. K., Kim, H., & Ishikawa, S. (2014). Histogram of spatio temporal local binary patterns for human action recognition. In *Joint 15th international symposium on soft computing and intelligent systems (SCIS) and 7th international conference on advanced intelligent systems (ISIS)* (pp. 1007–1011).

Akula, A., Shah, A. K., & Ghosh, R. (2018). Deep learning approach for human action recognition in infrared images. *Cognitive Systems Research, 50*, 146–154.

Al-Berry, M. N., Salem, M. A.-M., Ebeid, H. M., Hussein, A. S., & Tolba, M. F. (2016). Fusing directional wavelet local binary pattern and moments for human action recognition. *IET Computer Vision, 10*(2), 153–162.

Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., & Baskurt, A. (2011). Sequential deep learning for human action recognition. In *International workshop on human behavior understanding* (pp. 29–39). Springer.

Baumann, F., Lao, J., Ehlers, A., & Rosenhahn, B. (2014). Motion binary patterns for action recognition. In *International conference on pattern recognition applications and methods* (pp. 385–392).

Buonamente, M., Dindo, H., & Johnsson, M. (2016). Hierarchies of self-organizing maps for action recognition. *Cognitive Systems Research, 39*, 33–41.

Charalampous, K., & Gasteratos, A. (2016). On-line deep learning method for action recognition. *Pattern Analysis and Applications, 19*(2), 337–354.

Chaudhry, R., Ravichandran, A., Hager, G., & Vidal, R. (2009). Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In

IEEE conference on computer vision and pattern recognition (pp. 1932–1939).

Cheng, J., Liu, H., Wang, F., Li, H., & Zhu, C. (2015). Silhouette analysis for human action recognition based on supervised temporal t-sne and incremental learning. *IEEE Transactions on Image Processing, 24*(10), 3203–3217.

Chun, S., & Lee, C.-S. (2016). Human action recognition using histogram of motion intensity and direction from multiple views. *IET Computer vision, 10*(4), 250–256.

Chun, C.-S., & Lee, S. (2016). Human action recognition using histogram of motion intensity and direction from multiple views. *IET Computer vision, 10*, 250–256.

Guo, Z., Wang, B., & Xie, Z. (2017). A novel 3d gradient lbp descriptor for action recognition. *IEICE Transactions on Information and Systems, 100*(6), 1388–1392.

He, X., Wu, Q., Jia, W., & Hintz, T. (2008). Edge detection on hexagonal structure. *Journal of Algorithms & Computational Technology, 2*(1), 61–78.

Ijjina, E. P. (2016). Classification of human actions using pose-based features and stacked auto encoder. *Pattern Recognition Letters, 83*, 268–277.

Ji, S., Xu, W., Yang, M., & Yu, K. (2013). 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(1), 221–231.

Katircioglu, I., Tekin, B., Salzmann, M., Lepetit, V., & Fua, P. (2018). Learning latent representations of 3d human pose with deep neural networks. *International Journal of Computer Vision*, 1–16.

Kazak, N., & Koc, M. (2016). Performance analysis of spiral neighbourhood topology based local binary patterns in texture recognition. *International Journal of Applied Mathematics, Electronics and Computers*(4), 338–341.

Kellokumpu, V., Zhao, G., & Pietikäinen, M. (2008). Human activity recognition using a dynamic texture based method. *British machine vision conference* (Vol. 1, pp. 2). .

Krig, S. (2016). Interest point detector and feature descriptor survey. In *Computer vision metrics* (pp. 187–246). Springer.

Li, D., Yu, L., He, J., Sun, B., & Ge, F. (2016). Action recognition based on multiple key motion history images. In *13th international conference on signal processing (ICSP)* (pp. 993–996). IEEE.

Morton, P., & Waud, S. W. (1830). *Geometry, plane, solid, and spherical, in six books* (pp. 239). Angell Press.

Nguyen, H.-T., & Caplier, A. (2012). Elliptical local binary patterns for face recognition. In *Asian conference on computer vision* (pp. 85–96). Springer.

Nguyen, D. T., Li, W., & Ogunbona, P. O. (2016). Human detection from images and videos: A survey. *Pattern Recognition, 51*, 148–175.

Ojala, T., Pietikäinen, M., & Mäenpää, T. (2000). Gray scale and rotation invariant texture classification with local binary patterns. In *European conference on computer vision* (pp. 404–420). Springer.

Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(7), 971–987.

Popoola, O. P., & Wang, K. (2012). Video-based abnormal human behavior recognition-a review. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42*(6), 865–878.

Qu, S., & Li, T. (2017). Human action recognition based on improved cohog-lqc. In *IEEE conference on control and decision conference (CCDC)* (pp. 1928–1933).

Selmi, M., El-Yacoubi, M. A., & Dorizzi, B. (2016). Two-layer discriminative model for human activity recognition. *IET Computer Vision, 10*(4), 273–278.

Sharma, S, Kiros, R., & Salakhutdinov, R. (2015). Action recognition using visual attention, arXiv preprint arXiv: 1511.04119.

Sheena, C., & Narayanan, N. (2015). Key-frame extraction by analysis of histograms of video frames using statistical methods. *Procedia Computer Science, 70*, 36–40.

Shi, Y., Tian, Y., Wang, Y., & Huang, T. (2016). Sequential deep trajectory descriptor for action recognition with three-stream cnn. arXiv preprint arXiv: 1609.03056.

Sisodiya, A.S., Reducing dimensionality of data using neural networks.

Sorzano, C. O. S., Vargas, J., & Montano, A. P. (2014). A survey of dimensionality reduction techniques. arXiv preprint arXiv: 1403.2877.

Su, T.-F., Chiang, C.-K., & Lai, S.-H. (2016). A multiattribute sparse coding approach for action recognition from a single unknown viewpoint. *IEEE Transactions on Circuits and Systems for Video Technology, 26*(8), 1476–1489.

Van Der Maaten, L., Postma, E., & Van den Herik, J. (2009). Dimensionality reduction: a comparative review. *J Mach Learn Res, 10*, 66–71.

Veeriah, V., Zhuang, N., & Qi, G.-J. (2015). Differential recurrent neural networks for action recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 4041–4049).

Vidya, P., Veni, S., & Narayanankutty, K. (2009). Performance analysis of edge detection methods on hexagonal sampling grid. *International Journal of Electronic Engineering Research, 1*(4), 313–328.

Wang, F., & Sun, J. (2015). Survey on distance metric learning and dimensionality reduction in data mining. *Data Mining and Knowledge Discovery, 29*(2), 534–564.

Xu, K., Jiang, X., & Sun, T. (2017). Two-stream dictionary learning architecture for action recognition. *IEEE Transactions on Circuits and Systems for Video Technology, 27*(3), 567–576.

Yeffet, L., & Wolf, L. (2009). Local trinary patterns for human action recognition. In *IEEE 12th international conference on computer vision* (pp. 492–497).

Zhao, G., & Pietikainen, M. (2007). Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(6), 915–928.