Expert Systems With Applications 115 (2019) 524–534

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

On an algorithm for human action recognition

Suraj Prakash Sahoo*, Samit Ari

Department of Electronics and Communication Engineering, National Institute of Technology Rourkela, Odisha 769008, India

ARTICLE INFO

Article history:
Received 5 December 2017
Revised 21 June 2018
Accepted 10 August 2018
Available online 10 August 2018

Keywords:
Activity recognition
Interest point
Random projection tree
Local feature
Hough voting

ABSTRACT

Human action recognition which needs video processing in real time, requires large memory size and execution time. This work proposes a local maxima of difference image (LMDI) based interest point detection technique, random projection tree with overlapping split and modified voting score for human action recognition. In LMDI based interest point detection method, difference images are obtained using consecutive frame differencing technique and next, 3D peak detection is applied on the bunch of calculated difference images. Histogram of oriented gradients and histogram of optical flow as local features are extracted by defining a block of size 16×16 around each of the interest point. These local features are then indexed by random projection trees. Overlapping split is used during tree structuring to reduce failure probability. Hough voting technique is applied on testing video to compute highest similarity matching score with individual training classes. In addition to Hough voting score, the number of matched interest points of a single query video with each training class, is considered for recognition. The proposed method is evaluated on segmented UT-interaction dataset, J-HMDB dataset and UCF101 dataset. The experimental results indicate that the proposed technique provides better performance compared to earlier reported techniques.

1. Introduction

The increase in theft and vandalism have increased the need of 24×7 visual surveillance in the living premises, business areas and dense traffic areas. Current surveillance systems are comprised of surveillance cameras and it requires huge man power to handle the camera output. As the risk detection and scene understanding need careful observation over long period, the task cannot be left totally on human resource. Therefore, visual surveillance has attracted much more attention for the research by many researchers. Human action recognition (HAR) which is a part of visual surveillance, is a recent trend and a major attraction in computer vision. The efficient recognition of human actions is very much required for understanding a scene. As HAR is used to detect the behavioral changes of human being thus, it can also be used to detect the abnormality of a scene for surveillance applications. It includes motion characteristics of human body present in the temporal domain.

In real time surveillance, generally it is difficult to collect a larger training dataset for rarely occurring actions. As the abnormal actions do not occur frequently, a HAR must have the ability to recognize abnormal actions from insufficient training data. The present research work finds the application of HAR in real time scenario where training data is insufficient (Brendel & Todorovic, 2011; Gaur, Zhu, Song, & Chowdhury, 2011; Ryoo & Aggarwal, 2009; Ryoo & Yu, 2011; Seo & Milanfar, 2009; Yu, Yuan, & Liu, 2015). (Seo & Milanfar, 2009) have employed space time local steering kernels through dense computation for human action detection from a single video. (Ryoo & Yu, 2011) have created composed videos which are semi artificial to handle insufficient training. String of feature graphs (Gaur et al., 2011) is a technique which comprises of collection of graphs made up off spatio-temporal representations of low level features extracted from a video. (Yu et al., 2015) have worked towards insufficient training through propagative hough voting. The technique models the low level spatio-temporal features through tree structuring and casts votes to decide the action class. Ryoo and Aggarwal (2009) have developed the UT-interaction dataset and used it to test their algorithm with 20% training and 80% testing setup. Brendel and Todorovic (2011) have represented videos by spatio-temporal graphical structures and tested the algorithm on 20% training set up of UT-interaction dataset. From literature it is clear that the human action recognition paradigm is a challenging task when the training data is very less. The topic needs more attention to develop the algorithms to work for rarely occurring abnormal actions in real time scenarios.

* Coresponding author.
E-mail addresses: 515ec1003@nitrkl.ac.in (S.P. Sahoo), samit@nitrkl.ac.in (S. Ari).

https://doi.org/10.1016/j.eswa.2018.08.014
0957-4174/

For insufficient training, a HAR is developed in this work with the following propositions: (1) An efficient and simple Local maxima of difference images (LMDI) based interest point detector is proposed. It detects the motion due to body parts and then calculate the interest points by extracting local maxima. The advantage of this algorithm is that it extracts the regions having larger motion due to body as those regions play vital role to distinguish the actions. (2) Overlapping split instead of median split is used during tree formation. By doing this, the points lying to cell boundaries can be shared by both the split sub cells. This relatively reduces the failure probability due to boundary points. (3) Vote count is added to voting procedure to make it more efficient. When vote count and voting score is different during matching, the recognition can not be concluded. It searches the next best vote score and decides the matching class. The algorithm is evaluated on UT-interaction (Ryoo & Aggarwal, 2009), J-HMDB (Jhuang, Gall, Zuffi, Schmid, & Black, 2013) dataset and UCF101 (Soomro, Zamir, & Shah, 2012) dataset which are the benchmarked dataset for HAR. The proposed work is also tested on real time video data acquired for five action types similar to UT-interaction dataset (*'punch', 'kick', 'hug', 'handshake', and 'push'*). The experimental results show that the proposed technique performs better compared to earlier reported techniques for HAR.

The remainder of this paper is organized as follows. Related works are described in Section 2. Section 3 describes the proposed framework with explanation of proposed LMDI based interest point detection followed by feature extraction and RP tree formation. Advanced voting technique is explained for query videos to decide the recognition technique. Section 4 presents and widely discusses the experimental results with UT-interaction, J-HMDB and UCF101 datasets. The work of this paper is concluded in Section 5.

2. Related works

In the literature, many approaches are reported for human action recognition (HAR). Some important related techniques are discussed here. Motion descriptors (Dollár, Rabaud, Cottrell, & Belongie, 2005; Efros, Berg, Mori, & Malik, 2003; Laptev, 2005; Scovanner, Ali, & Shah, 2007) are useful to represent a video more efficiently. Spatio-temporal patterns are important to define a particular event and at the same time it differentiates one event from other. Laptev (2005) have proposed STIP by extending the Harris corner detection from spatial to spatio-temporal domain. STIP interest points are detected by detecting 3D Harris corners in video. Cuboids technique was proposed by Dollár et al. (2005) to represent the video using Gabor filtering on the spatial and temporal dimensions individually. A 3D SIFT descriptor is proposed by Scovanner et al. (2007) which is an extended work of 2D SIFT descriptor. Bag-of-Words (BoW) (Csurka, Dance, Fan, Willamowski, & Bray, 2004) model is adopted by many researchers (Dollár et al., 2005; Junejo, Dexter, Laptev, & Perez, 2011; Niebles, Wang, & Fei-Fei, 2008) to model the motion features. In this BoW method, the motion features are used to make a codebook having certain codewords. 3D shape models have been used to model the human actions (Gall, Yao, Razavi, Van Gool, & Lempitsky, 2011; Leibe, Leonardis, & Schiele, 2008; Yu et al., 2015). Yu et al. (2015) proposed an implicit spatial temporal shape model using the spatio-temporal configuration information among interest points. Recently tree structuring (Dasgupta & Sinha, 2013; Yu, Yuan, & Liu, 2012; 2015) is used efficiently to index the feature space. In recognition procedure, Hough transform (Gall et al., 2011; Leibe et al., 2008) is used by many researchers which is basically a process of the addition of votes from local patches to recognize the action. Hough forest techniques are reported by Gall et al. (2011). In this technique, random forest and hough transform are adapted combiningly for human action recognition. Peng, Zou, Qiao, and Peng (2014) have proposed stacked fisher vectors and fusion of multiple coding to improvise the dense trajectory based action recognition techniques. Deep learning techniques are also presently applied for HAR (Chen, Chen, Hu, Chen, & Wang, 2017; Ji, Xu, Yang, & Yu, 2013; Shi, Tian, Wang, & Huang, 2017; Yu, Cheng, Xie, & Li, 2017). Ji et al. (2013) have extracted features in both spatial and temporal dimension by proposing a 3D Convolutional neural network (CNN) model for human action recognition. Chen et al. (2017) have developed a temporal scale invariant deep learning framework which handles the action speed or duration of the action. Though these deep learning techniques are applied for HAR, but these techniques need a large training data to train the model.

3. Proposed framework

The block diagram of the proposed human action recognition technique is shown in Fig. 1. Processing of a video requires large memory and time complexity. To reduce the computational load, the video is described by extracting important spatio-temporal interest points. Then, these interest points are described by extracting features around its neighborhood. Tree structuring with overlapping split is used to subgroup similar points by putting them in a single leaf at a certain depth. This helps to reduce the searching time during testing. When a query video arrives, its interest points are extracted and described by features. The extracted interest points are then passed through the training trees. When a query interest point is passed through a tree, it reaches certain leaf node at certain depth. The leaf node contains the trained interest points which are decided as matched interest point to that query point. These matched interest points are then cast votes to the query point. Accumulation of all votes from all query interest points is used to recognize the action. The details are explained in subsequent subsections.

3.1. Local maxima of difference image (LMDI) based interest point detection

Interest points in a video are generally edges, corners etc. Movement of 3D corners is one of the best technique to describe actions (Laptev, 2005) as it describes the motion of corners in spatial as well as in temporal direction. Therefore, 3-Dimensional Harris corner based interest point detection (STIP) (Laptev, 2005) technique is applied in this work. This is an extension of the Harris corner detection from the spatial domain to spatio-temporal domain. The method is all about finding local structures in space-time where the data values have significant local variations in both space and time. During experiments, it is observed that the STIP method generally detects the movement of 3D corners by Harris corner detection. However, in some videos as 3D corners are not prominent, important motion areas remain undetected (Dollár et al., 2005). Detection of those less prominent corners can add more accuracy to the overall detection as they are important sections to represent an action. To overcome these limitations, a local maxima of difference images (LMDI) based interest point detector is proposed and evaluated for human action recognition.

The LMDI based interest point detection technique is explained in Algorithm 1 . Assume that the query video is V with N_f number of frames. In this method, the difference images (D_i) are calculated by consecutive frame differencing. Median filtering is applied on difference images to suppress unwanted spurs or unwanted local maxima. 3D local maxima calculation is then imposed on bunch of difference images. In this method, each and every pixel is compared with 26 neighbor pixels around it in a $3 \times 3 \times 3$ pixel block to decide it as maxima or non maxima. The pixel which is to be compared, is denoted as center pixel C. The pixel block structure is

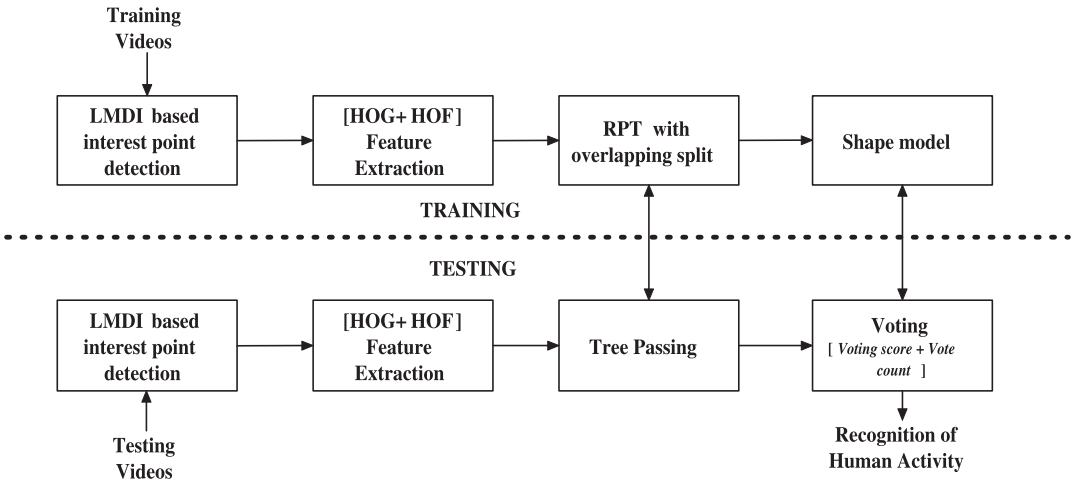


Fig. 1. Block diagram of the proposed human action recognition technique.

Algorithm 1: Proposed LMDI based interest point detection.

Initialize query video = V with N_f number of frames, interest points IP = 0, iteration i=1;

Procedure $DIFF\_BUNCH(V, N_f)$

while $i < N_f$ **do**
  $D(i) = Medianfilter(V(:, :, i+1) - V(:, :, i))$;$i = i$+1;
end

Procedure $LOCAL\_MAXIMA(D)$

Calculate 26 no. of spatio-temporal neighbor pixels around each pixel and decide whether a maxima or not

while *each pixel of D is not processed* **do**
  CenterPixel=Current pixel
  NeighborPixel=26~pixels around CenterPixel in a 3 × 3×3~pixel block size
  for *l=1:26* **do**
    if *intensity(NeighborPixel(l)) > intensity(CenterPixel)* **then**
      Not an interest point, **Break** and exit the loop
    else
      Continue
    end
  end
  if *no neighbour pixel intensity > center pixel intensity* **then**
    Interest point=*CenterPixel*
    IP← Save location information of *CenterPixel*
    Process next pixel of *D*
  end
end

Procedure $DISCARD(IP)$

Discard closely positioned interest points

for *i=2 : size(IP)* **do**
  E=Euclidean_distance(IP(i) , IP(i-1))
  if *E < Threshold* **then**
    Discard the point
  end
end

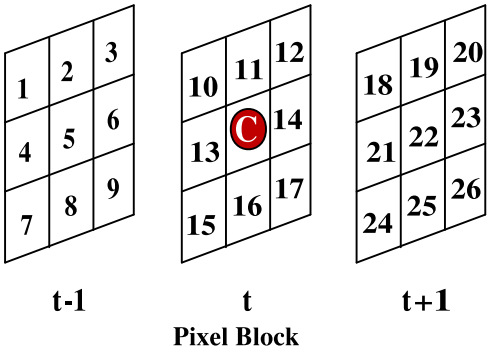


Fig. 2. $3 \times 3 \times 3$ pixel block structure with 26 spatio-temporal neighbor pixels and center pixel C of current time frame t with previous frame t-1 and next frame $t + 1$.

shown in Fig. 2 for the center pixel C. If all of the neighbor pixels are having less intensity than center pixel then, the pixel is recognized as an interest point. After extracting all the maxima, euclidean distance between two consecutive maxima points are compared with an empirically chosen threshold value to discard the closely positioned interest points. The threshold value is chosen by repetitive observations from experiments. The value is 20 pixel distance in x-direction and 20 pixel distance in y-direction.

Subjective description about proposed interest point detection technique is shown in Fig. 3. An example frame of UT-interaction dataset (punching action) is represented in Fig. 3(a). The difference frame obtained by consecutive frame differencing is shown in Fig. 3(b). The contour representation in Fig. 3(c) is shown to compare the maxima positions with the detected interest points. Closely positioned interest points and the points having less magnitude in contour plot are discarded as they are treated as false peaks. The yellow circles in Fig. 3(d) represents the interest points for the current frame.

3.2. Feature extraction

Each interest point is described with histogram of oriented gradient (HOG) (Dalal & Triggs, 2005) and histogram of optical flow (HOF) (Chaudhry, Ravichandran, Hager, & Vidal, 2009; Munoz-Salinas, Medina-Carnicer, Madrid-Cuevas, & Carmona Potayo, 2008) by taking a 16×16 patch around that point. The dimension of HOG based feature set is 72 and the dimension of HOF based feature set is 90. The overall feature set dimension is 162 (Laptev, Marszalek, Schmid, & Rozenfeld, 2008).

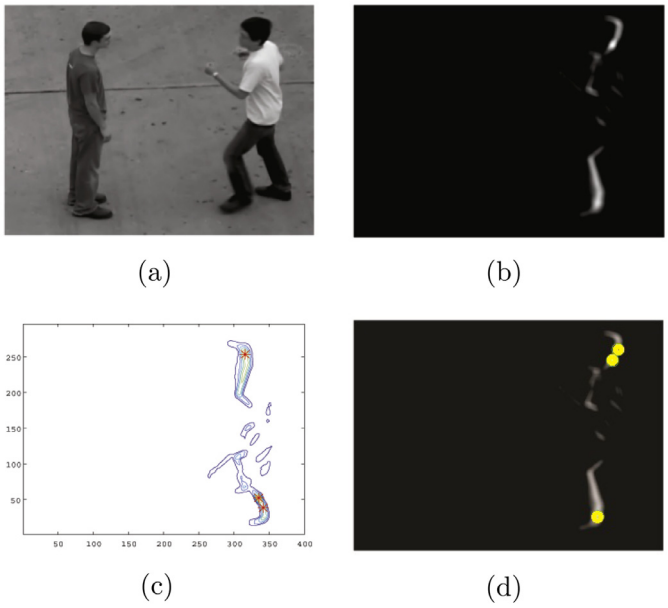


Fig. 3. LMDI based Interest point detection applied on punching action of UT-interaction dataset. (a) Example frame of punching action, (b) Difference frame by consecutive frame differencing, (c) Contour representation, (d) Detected interest points.

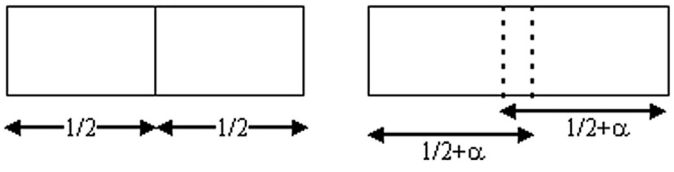


Fig. 4. Median split vs Overlapping split to be used in RP Tree during cell splitting.

3.3. Random projection tree (RP Tree) with overlapping split

The feature space is indexed by tree structures, so as to find the matched interest points efficiently by passing the query interest points from the root to the leaf nodes (Yu et al., 2015). For this work random projection trees (Freund, Dasgupta, Kabra, & Verma, 2007) are used to index the feature space. The RP tree technique is followed because it is better than nearest neighbor as it can adapt to the low dimensional manifold (Freund et al., 2007). It provides a fast search for query interest points by indexing the training data. The algorithm for tree structuring is taken from Freund et al. (2007) and reimplemented. By analyzing the tree structures, the following problem is found. When the interest points are concentrated near the cell boundaries, the similar interest points may well lie in a different cell. As a result, the failure probability of this RP Tree method will be high.

As shown in Fig. 4, the overlapping between cells is allowed as reported by (Dasgupta & Sinha, 2013) to reduce the failure probability. Each cell C is split along a direction U chosen at random from the unit sphere. The overlapping spread is controlled by the variable α which is a very small quantity i.e. 0.05 or 0.1. The data will be split along the $(1/2-\alpha)$ fractile value $l(C)$, and the $(1/2+\alpha)$ fractile value $r(C)$.

RP Tree technique is shown in Algorithm 2 . Assume that a set of interest points extracted from an acquired video is denoted by $P=\{p_i; i=1,\ldots,N_P\}$, where $p_i=[f_i, l_i]$, where f_i and l_i are the descriptor and spatio-temporal location of interest point p_i, respectively. N_P is the total number of interest points. The parameter t_d is the maximum tree depth during the construction of tree. The nature of the split is defined in a subroutine SPLITDATA and the core tree-building algorithm is called MAKETREE.

Algorithm 2: Random projection tree (overlapping split).

Initialize P=Feature set, t_d=maximum tree depth, *MinSize*=Minimum number of interest points in a leaf node.

Trees = Construct RPTree (P)
MAKETREE (P, 0)

Procedure MAKETREE (P,*depth*)
if $depth < t_d$ **then**
  if $|P| <$ *MinSize* **then**
    return *(Leaf)*
  else
    $Rule \leftarrow SPLITDATA(P)$
    $LeftTree \leftarrow MAKETREE(\{x \in P : Rule(x) = true\} \cup \{x \in P : x \in P_{common}\})$
    $RightTree \leftarrow MAKETREE(\{x \in P : Rule(x) = false\} \cup \{x \in P : x \in P_{common}\})$
    return $([Rule, LeftTree, RightTree])$
  end
end

Procedure SPLITDATA (P)
Choose a random unit direction v.
Sort projection values: $p(x) = v.x \ \forall x \in P$, generating the list $p_1 \le p_2 \le \ldots \le p_n$
for i=1,... ,n-1 **do**
  $\mu_1 = \frac{1}{i}\sum_{j=1}^{i} p_j$, $\mu_2 = \frac{1}{n-i}\sum_{j=i+1}^{n} p_j$
  $c_i = \sum_{j=1}^{i}(p_j-\mu_1)^2 + \sum_{j=i+1}^{n}(p_j-\mu_2)^2$
  find i that minimizes c_i and set $\theta = (p_i + p_{i+1})/2$
  $Rule(x) := v.x \le \theta$
end

3.4. Shape model with modified voting

Implicit spatial temporal shape model (Yu et al., 2015) has been introduced to address the matching problem. This was introduced by G. Yu *et al.* while motivated from the work of 2D implicit shape model (Leibe et al., 2008). After extracting STIPs, each of them are described with HOG and HOF features combiningly. Then RP trees are constructed by taking all the interest points into consideration. Training data is referred as $R=\{d_r=[f_r, l_r]; r=1,2,\ldots,N_R\}$, where f_r and l_r are the descriptor and spatio-temporal location of interest point d_r, respectively. N_R is the total number of interest points. Let $V(x, t, \rho)$ refers to the test video with temporal center t and spatial center x, ρ refers to the scale size and duration of the test video. The extracted interest points for V is denoted by P as described in previous section. Then, similarity matching score (Yu et al., 2015) between V and R is calculated as:

$$S(V(x,t,\rho),R) \propto \sum_{d_r \in R}\sum_{p_i \in V}\sum_{j=1}^{N_T} I_j(f_i,f_r)\exp\left(-\frac{[\|x_v-x,t_v-t\|]^2}{\sigma^2}\right) \quad (1)$$

where $I_j(f_i, f_r)=1$ if f_i, f_r are in same leaf of Tree T_j, otherwise 0. $l_v=[x_v, t_v]$ is the voting position, where,

$$x_v = x_i - \eta_x(x_r - c_r^x)$$
$$t_v = t_i - \eta_t(t_r - c_r^t) \quad (2)$$

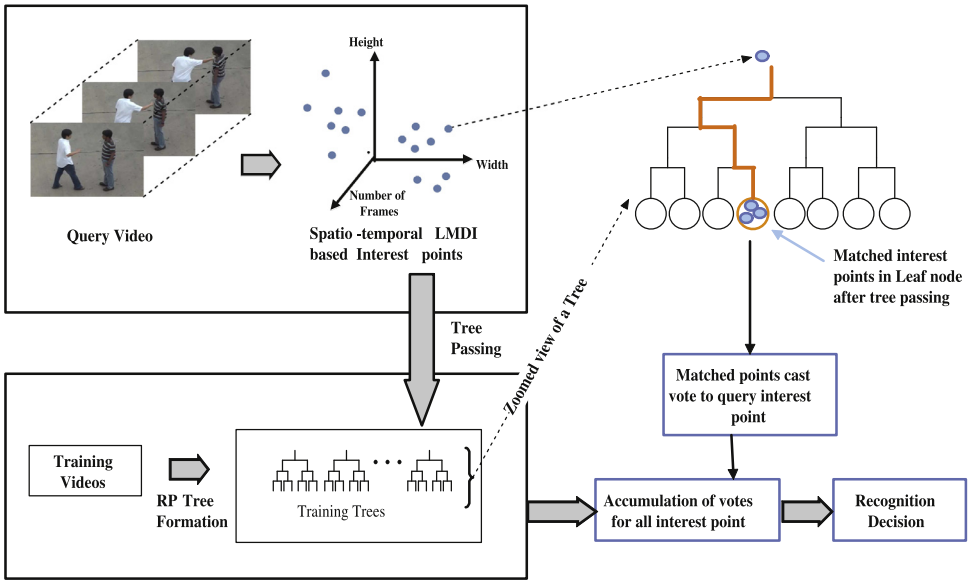


Fig. 5. Detailed Voting Procedure. Interest points are extracted from query video (Blue circles) and then passed through trained RP trees for interest point matching. One example interest point (blue) is shown in right hand side for tree passing (Orange colored path). The leaf containing training interest points are the matched interest points for a single query point to cast vote for it.

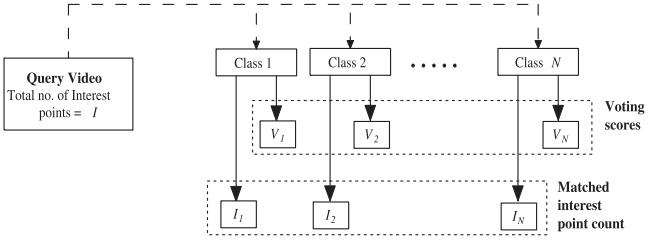


Fig. 6. Voting score and matched interest points of a single query video for each training class.

and $[x_i, t_i] = l_i$, $[x_r, t_r] = l_r$, $[c_r^x, c_r^t]$ is spatio-temporal center position of training action, $\eta = [\eta_x, \eta_t]$ is scale and duration of testing video. The detailed voting procedure is shown in Fig. 5.

The similarity matching score depends on the $[x_v, t_v]$, which is the difference of euclidean distances between query point to action center ($[c_r^x, c_r^t]$) and matched training interest points to action center ($[c_r^x, c_r^t]$). If training videos of different classes have unequal number of extracted interest points, then at a certain depth for trees, the leaf nodes will contain unequal number of interest points. If the original class has higher number of interest points in leaf node, then the accumulation of error distance for original class will be more in comparison to false positive classes. Thus, the query video can be misclassified to false positive class. This observation motivates to modify the voting technique by the help of matched interest point count. The technique is described in following sub section.

3.4.1. Advanced voting technique

The sum of difference of euclidean distances of matched points to query points are calculated. This is used to calculate the similarity matching score based on (1). In addition to this parameter, number of interest points which matches with each individual class, are counted. By taking combination of these two, the query videos are recognized.

Voting as well as matched interest point count is shown in Fig. 6. Here V_i, $i = 1, \ldots, N$ is the voting score to the i^{th} class. I_j is the number of interest points matched to j^{th} class and $\sum_{j=1}^{N} I_j = I$ (Total number of interest points). Algorithm for advanced voting technique is shown in Algorithm 3 and described as follows:

- The initial recognized class is decided by voting score V_i. Now, to strengthen the recognition, the count of matched query interest points is taken into consideration. If the recognized class has maximum number of matched interest points to that class, then the query video is recognized as the same class as before.
- Otherwise, the classes with higher number of matched interest points are compared to initially recognized class. The class which has higher accumulate voting score, is decided as recognized class.

Algorithm 3: Proposed advanced voting technique.

Initialize the algorithm with pre-calculated Voting scores=V_i for i=1,... ,N
Matched interest point counts=I_j for j=1,... ,N
N=number of classes

Determine $(V_i)_{\max}$ and $(I_j)_{\max}$
if $i == j$ **then**
  Detected class label = i
else
  I_i = matched interest point count where $V_i = (V_i)_{\max}$
  Find class labels k where $I_k > I_i$
  Locate $(V_k)_{\max}$ for all k
  Detected class label = k
end

4. Results & discussion

For the evaluation of the proposed method, 'UT-Interaction' (Ryoo & Aggarwal, 2009), 'J-HMDB' (Jhuang et al., 2013) and 'UCF101' (Soomro et al., 2012) datasets are used in this work. The proposed method is implemented using MATLAB Version: 9.0.0.341360 Release 2016a of Mathworks Inc. The algorithms are executed on an Intel Core i5 personal computer, clock 3.20 GHz, RAM 4GB Windows10 platform.

Fig. 7. Detected interest points using Harris corner detection on UT-interaction Dataset for actions pushing, handshaking, hugging, kicking, pointing and punching (left → right).

Fig. 8. Detected interest points using proposed LMDI based interest point detection on UT-interaction Dataset for actions pushing, handshaking, hugging, kicking, pointing and punching (left → right).

4.1. Datasets

UT-Interaction dataset. The 'UT Interaction' dataset is a public video dataset containing high-level human actions of multiple actors. It contains segmented actions of *punching, handshaking, hugging, kicking, pushing, pointing*, each of 10 sets. Total number of video sequences are 60. Each sequence except *pointing* contains a single interaction between two human being. The *pointing* sequences contain action of a single person. To evaluate the proposed method, the dataset is divided randomly into two parts for training and testing set. Since all videos are from different subject, the set up is subject independent. Randomly 9 sequences out of 10 of each class are used for testing and 1 sequence is used to train the model. Therefore, 10% of data is used as training dataset (Gaur et al., 2011; Yu et al., 2015).

J-HMDB dataset. The J-HMDB dataset contains single person interaction. The dataset contains 21 categories involving a single person in action: *brush hair, catch, clap, climb stairs, golf, jump, kick ball, pick, pour, pull-up, push, run, shoot ball, shoot bow, shoot gun, sit, stand, swing baseball, throw, walk, wave*. Each action class is having 36–55 videos. During evaluation of the dataset, randomly six videos from each class are taken as training dataset. During testing, randomly 30 videos from each class are tested.

UCF101 dataset. The UCF101 dataset is a more realistic dataset collected from YouTube. The dataset contains 101 action classes with 13,320 video clips. The videos in 101 action categories are grouped into 25 groups, where each group can consist of 4–7 videos of an action. The action categories can be divided into five types: (1)Human-Object Interaction (2) Body-Motion Only (3) Human-Human Interaction (4) Playing Musical Instruments (5) Sports.

4.2. Experiments on UT-Interaction dataset

As the hands and legs are important body parts to represent an action, STIP method fails to do so at certain instances. This is overcome by the proposed technique. In Figs. 7 and 8, a comparative subjective evaluation of proposed interest point detection technique is shown. The yellow markers are shown for comparison. Fig. 7 is for STIP points detected for six classes (*pushing, handshaking, hugging, kicking, pointing, punching*) of UT-interaction dataset and Fig. 8 is for the proposed interest point detection. Here, for comparison purpose detected interest points are shown for similar frames with both the techniques.

The interest points are detected using STIP (Laptev, 2005), LMP feature detection (Guha & Ward, 2012) and our proposed LMDI based interest point detection method. The result for execution time of proposed technique with other existing techniques are given in Table 1. The execution time which is shown in the table, is calculated by averaging the time taken for all six classes during interest point detection. The proposed technique takes less time in comparison to other two methods. It is because the difference frame calculation is a spatial operation on which spatio-temporal maxima calculation are imposed. As spatial operations are less complex than spatio-temporal operations, they are faster. In the same context, the STIP method detects 3D Harris corners in spatio-temporal domain and thus slower. LMP feature extraction method uses 2D Harris corner detection technique. Therefore, its execution time is comparable to proposed technique. The execution time for proposed technique is a little less than LMP feature detection but at the same time the proposed method provides more number of interest points to represent the video. As the LMDI based method produces more number of interest points, this work is also studied for the performance of the proposed method with different number of interest points as given in Table 2. In post processing step, if interest points from alternate frames are discarded, the number of interest points is reduced to 759 on an average. At the same time, the accuracy of the technique is reduced to 72.22%. Similarly, when interest points are considered from one frame out of four consecutive frames, the interest points are dropped to 379 and the accuracy is reduced to 62.96%. Therefore, More number of interest points to motion region adds more confidence to describe an action. this is why, in our experiment, all the interest points are taken into consideration.

In Table 3, statistical indices (Sensitivity, Specificity, Positive predictivity) for each class are shown to compare the performance of all classes. Sensitivity defines the proportion of positives that are

Table 1
Comparison of execution time for different interest point detection methods applied on UT-interaction dataset.

Method	Time (in second)	No of interest points
STIP (Laptev, 2005)	7.09	840
LMP feature detection (Guha & Ward, 2012)	5.25	146
Proposed LMDI based Interest Points	5.22	1486

Table 2
Performance analysis based on interest points.

Sl No	Number of interest points	Discarded frames	Accuracy
1	1486	0	75.93%
2	759	1 out of each 2	72.22%
3	379	1 out of each 4	62.96%

Table 3
Comparative performances of six classes of UT-interaction dataset using proposed technique.

Class	Sensitivity	Positive predictivity	Specificity
Punching	0.78	0.63	0.91
Handshake	0.78	0.7	0.93
Hugging	0.67	1	1
Kicking	0.67	1	1
Pushing	0.89	0.57	0.87
Pointing	0.78	1	1

Table 4
Confusion matrix for the performance of action recognition using proposed method for UT-interaction dataset.

	Punch	Handshake	Hug	Kick	Push	Point
Punch	**7**	0	0	0	2	0
Handshake	0	**7**	0	0	2	0
Hugg	0	1	**6**	0	2	0
Kick	3	0	0	**6**	0	0
Push	1	0	0	0	**8**	0
Point	0	2	0	0	0	**7**

Table 5
Comparison of action classification on UT-interaction dataset using proposed technique with 10% training.

Technique	Accuracy
BoW (Csurka et al., 2004)	34.44%
SFGGaur et al. (2011)	65%
RPT+HV (Yu et al., 2015)	73%
Proposed Technique	75.93%

correctly identified and it can be calculated by taking the ratio of correctly classified action among all actions. Specificity defines the proportion of negatives that are correctly identified and it can be calculated by taking the ratio of the number of correctly rejected non class actions (True Negative) to the total number of non class actions. Positive predictivity is the ratio of the number of correctly detected actions to the total number of detected actions for a class. The pointing action class is having a sensitivity of 0.78, specificity of 1, and positive predictivity of 1 which is comparatively better than all other classes. Thus, this class is better classified in comparison to others. The reason behind the better performance lies with the action containing only single person. All other actions are described by multiple persons.

In Table 4, confusion matrix for detection of various actions for each class is shown. Each row contains total number of test videos for a particular class. The diagonal cells are having the count of successfully detected actions for each class. It is observed that pushing action is better classified in comparison to other classes with an accuracy of 89%. At the same time, this class is having higher false detection rate. The false detections are coming from punching, handshaking and hugging classes because in all these classes hand motion plays an important role to describe the actions.

Bag of Words (BoW) (Csurka et al., 2004) is a standard method to compare performance of any proposed technique. The technique clusters the features and makes histogram of clustered feature set. As mentioned in Gaur et al. (2011), the string of feature graph (SFG) performs better in comparison of BoW technique for HAR application. In this work, BoW technique is compared with proposed technique and it is shown in Table 5. The accuracy in BoW method is found to be 34.44%. SFG method (Gaur et al., 2011) extracts the spatio-temporal relationship between local features formed on different time windows. The method relies on strings or clusters of local features and in Yu et al. (2015), the local features are indexed through RP tree. The technique uses random projection and minimum variance to split the data through tree structures. When SFG works for a constant time window, RP tree technique groups similar feature points from any time frame, thus adds more flexibility. The average accuracy is found to be 73% for RPT (Yu et al., 2015) and 65% for SFG (Gaur et al., 2011). The proposed technique is better due to addition of vote counting and overlap split tree structuring to LMDI based interest point detection technique. The accuracy is compared in Table 5 with earlier reported techniques and it is found to be 75.93% that is significantly better than the average accuracy of Yu et al. (2015) and Gaur et al. (2011).

Table 6
Comparison of action classification on UT-interaction dataset using proposed technique with 20% training.

Technique	Accuracy
Ryoo and Aggarwal (2009)	70.8%
Brendel and Todorovic (2011)	78.9%
NN+HV (Yu et al., 2015)	75%
RPT+HV (Yu et al., 2015)	85.4%
Proposed technique	87.5%

Since the actions are having common motion parts (leg and hand motion), it needs more training data to train the shape models. If training data is very less such as 10% as used in this work, the recognition technique becomes more challenging to handle inter-class and intra-class variations. However, advantage of study on insufficient training is that, it is more effective and needful for human action recognition in real time environment. The algorithm is also tested with 20% training data as in Ryoo and Aggarwal (2009) and Brendel and Todorovic (2011). Here, randomly 2 videos from each class are taken for training and rest for testing. The performance is shown in Table 6. The proposed work gives an accuracy of 87.5% which is better than state of the art method. The class wise performance is shown as a bar diagram in Fig. 9. In bar diagram, the proposed technique is compared with Ryoo and Aggarwal (2009) and Brendel and Todorovic (2011). The work is unable to be compared with the work of Yu et al. (2015) as they have not mentioned the class wise accuracy. Except 'hug' action other action classes give better performance in comparison to earlier techniques.

4.3. Experiments on J-HMDB dataset

J-HMDB dataset contains 30 action classes. As our method heavily dependent on interest point detection, the classes which are not having prominent 3D motion corners or motional body parts, are excluded from evaluation. The excluded action classes are *pick, pour, shoot gun, sit, stand*. When interest points from these five classes are added to tree formation, the tree structures are more erroneous and as a result the overall accuracy of the system is going below 30%.

To evaluate the proposed algorithm in this dataset, two adjustments are made. First, to work for insufficient training, randomly six videos from each class (total 96 videos out of 703 videos of 16 selected classes=13.67% training) are used for RP tree formation. Second, during testing, randomly 30 videos from each class are tested. Therefore, during a single observation 480 test videos are considered.

In Table 7, confusion matrix for detection of various actions for 16 classes of J-HMDB dataset is shown. Each row contains total number of test videos for a particular class. The diagonal cells are having the count of successfully detected actions for each class. At the bottom of the confusion matrix, accuracy of each class is shown. From confusion matrix, it is found that the classes *'brush hair', 'shoot ball', 'shoot bow'* are classified better in comparison to

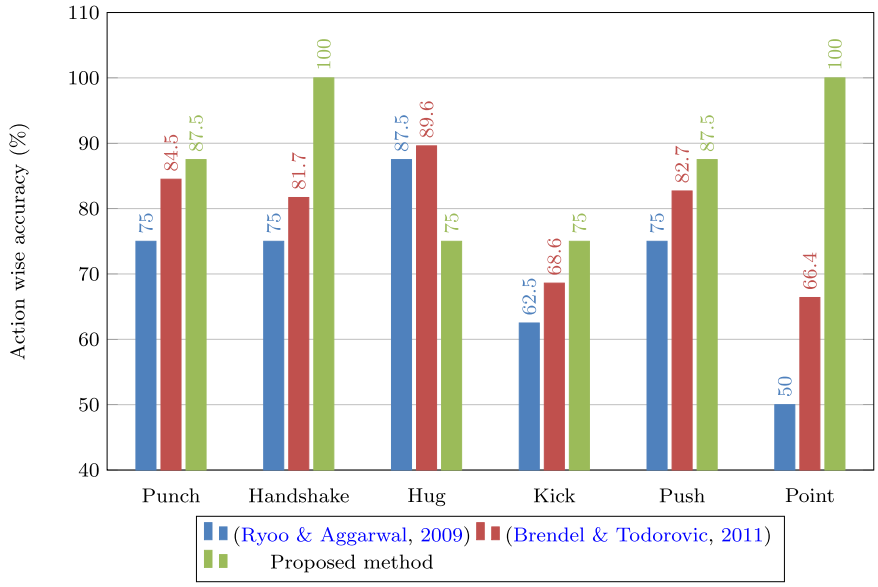


Fig. 9. Class wise performance of UT-interaction actions with 20% training.

Table 7
Confusion matrix for the performance of action recognition using proposed method for J-HMDB dataset for 16 classes.

	Brush hair	Catch	Clap	Climb stairs	Golf	Jump	Kick ball	Pull up	Push	Run	Shoot ball	Shoot bow	Swing baseball	Throw	Walk	Wave
Brush hair	**23**	0	0	0	2	0	0	0	3	0	0	1	0	1	0	0
catch	4	**12**	2	1	0	0	0	3	1	0	4	0	1	0	0	2
clap	2	0	**17**	1	0	0	2	4	0	0	1	0	2	0	0	1
climb stairs	1	0	3	**14**	1	0	0	5	3	0	0	0	0	0	1	2
golf	2	0	0	4	**21**	0	0	2	0	0	0	1	0	0	0	0
jump	0	1	1	0	0	**15**	3	0	2	0	3	2	0	2	0	1
kick ball	0	1	0	2	0	0	**18**	4	1	2	0	0	0	0	1	1
pull up	2	0	3	3	1	0	2	**18**	0	0	0	1	0	0	0	0
push	1	1	6	1	0	0	1	3	**8**	1	2	5	0	0	0	1
run	3	0	1	2	0	0	2	1	1	**13**	4	0	1	0	1	1
shoot ball	0	0	2	1	0	0	2	1	0	0	**23**	0	0	1	0	0
shoot bow	0	0	0	0	0	0	1	0	1	1	0	**25**	0	2	0	0
swing baseball	2	1	0	2	1	0	0	1	0	0	3	0	**18**	0	2	0
throw	3	1	0	0	1	0	0	2	1	1	0	2	0	**19**	0	0
walk	0	0	4	0	3	1	0	4	0	0	2	0	0	1	**14**	1
wave	3	1	1	1	1	0	3	2	0	0	2	1	1	2	0	**12**
Accuracy	76.67	40.00	56.67	46.67	70.00	50.00	60.00	60.00	26.67	43.33	76.67	83.33	60.00	63.33	46.67	40.00

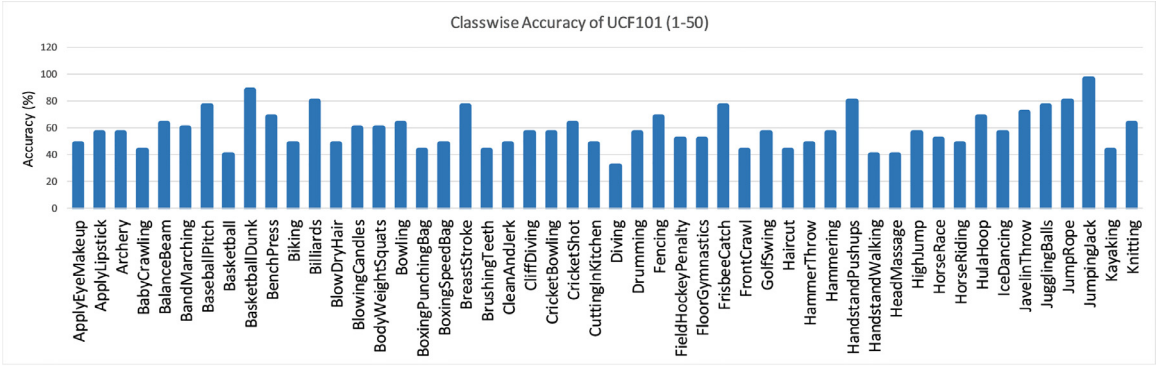
Table 8
Comparative performances of sixteen classes of J-HMDB dataset using proposed technique.

	Brush hair	Catch	Clap	Climb stairs	Golf	Jump	Kick ball	Pull up	Push	Run	Shoot ball	Shoot bow	Swing baseball	Throw	Walk	Wave
sensitivity	0.77	0.40	0.57	0.47	0.70	0.50	0.60	0.60	0.27	0.43	0.77	0.83	0.60	0.63	0.47	0.40
positive predictivity	0.50	0.67	0.43	0.44	0.68	0.94	0.53	0.36	0.38	0.72	0.52	0.66	0.78	0.68	0.74	0.55
specificity	0.95	0.99	0.95	0.96	0.98	1.00	0.96	0.93	0.97	0.99	0.95	0.97	0.99	0.98	0.99	0.98

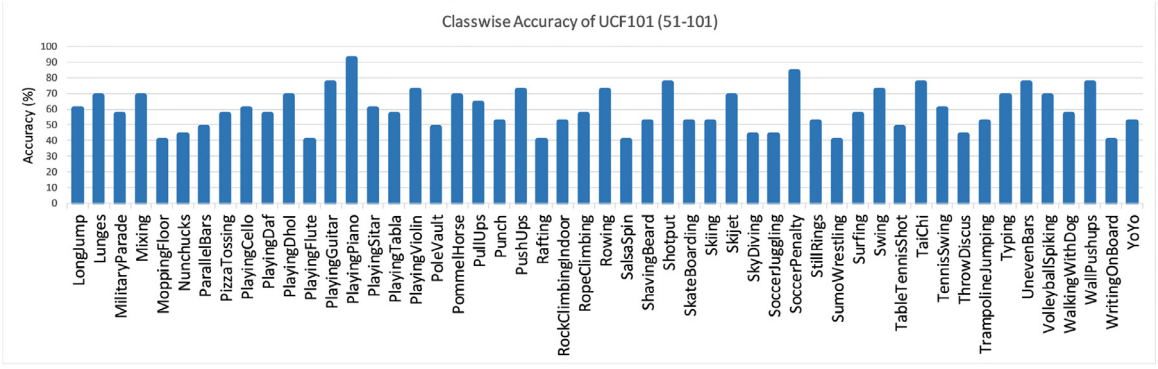
other classes. The better performance is due to the presence of more similar action videos in these classes. The action like *'push'* is more difficult to recognize as it is having more intra class variation. The overall accuracy of the proposed technique on J-HMDB dataset is found to be 56.25%. In Table 8, statistical indices (Sensitivity, Specificity, Positive predictivity) for each class are shown to compare the performance of all classes. According to accuracy the 'shoot bow' is the better recognized class. However, according to performance matrix, the 'jump' action is better classified. the reason is, other actions are not confused much with this class. The performance on J-HMDB dataset suggests that in real complex context, the algorithms are more challenged.

4.4. Experiments on UCF101 dataset

The UCF101 dataset is having 101 classes and roughly 100–170 videos per class. Total number of action videos present is 13,320. In our experiment, two different settings are adopted to validate the performance of the proposed technique. The first setting is to train with randomly 15 videos from each class ($15 \times 101 = 1{,}515$ out of 13,320 *i.e.* 11.37% training) and the second one is to train with 30 videos from each class ($30 \times 101 = 3{,}030$ out of 13,320 *i.e.* 22.75% training). During testing randomly 85 videos are tested in first set up and 70 videos are tested in second set up.



(a)



(b)

Fig. 10. Class wise accuracy of UCF101 dataset with insufficient training (setting 1: 11.37% training). (a) accuracy of action classes 1 to 50, (b) accuracy of action classes 51 to 101.

The class wise accuracy of the proposed technique with first setting is shown as bar diagram in Fig. 10. The overall accuracy is found to be 58.39%. As mentioned by Soomro et al. (2012), the actions of UCF101 dataset are divided into five types: (1) Human-Object Interaction (2) Body-Motion Only (3) Human-Human Interaction (4) Playing Musical Instruments (5) Sports. Therefore, an analysis on different action types are performed to evaluate the proposed technique. The accuracy by proposed technique for the predefined action types are: Human-Object Interaction (55.24%), Body-Motion Only (64.04%), Human–Human Interaction (48%), Playing Musical Instrument (63.76%), Sports (57.81%). Better accuracy of 'Body-Motion Only' and 'Playing Musical Instrument' actions are due to better interest point detection. In 'Body-Motion Only' actions, the interest points detected are more concentrated near the motional body parts. For 'Playing Musical Instrument', the interest points are more concentrated near the hand region where the instruments are being played.

Similarly the experiments are carried out for second setting (22.75% training). The increased training data has helped for better representation of actions and as a result the performance has increased. With second set up, the overall accuracy by proposed technique is found to be 65.11%. The accuracy for the predefined action types are: Human-Object Interaction (62.82%), Body-Motion Only (70.51%), Human-Human Interaction (56%), Playing Musical Instrument (68.47%), Sports (64.54%).

4.5. Experiments on real time dataset

The action recognition framework proposed in Section 3 is evaluated on action videos acquired in real time. The data is acquired through a Nikon5300 camera with a specification of 1080 × 1920 frame size and 50 frames per second. In preprocessing stage, the frames are down sampled to 120 × 160 and converted to gray scale images. The real time query videos are tested on RP trees made from training through UT-interaction dataset. One video from each class of UT-interaction dataset is used to form the training trees. Five different real time classes are tested: *'punch', 'kick', 'hug', 'handshake', and 'push'*. All actions contain different complex backgrounds. Action frames from each class is shown in Fig. 11. The recognized actions are marked with green markers and misclassified classes are with red markers. Out of 25 real time videos (5 per class), 16 videos are correctly classified. As the complexity is more for real time data, the recognition rate is lower than the benchmarked datasets. As the acquired real time data comprises of complex backgrounds and thus adds more false positive rate for real time recognition.

5. Conclusion

In this work, an LMDI based interest point detection technique is proposed to detect the interest points from the video where 3D corners are not prominent. This development is able to detect the motional body parts which describes an action more efficiently. Overlapping split is adopted into RP tree to handle the points which are residing at the boundary during splitting. In median split, similar interest points present at boundary may lie in different cells. Thus, sharing of points lying near boundary is allowed between two consecutive cells. Voting technique is well supported by vote counts. Along with voting sum, the number of points matching with that class, is taken into consideration in this work. The advantage of the proposed work is that, it is fast due to LMDI based interest point detection and it han-

Fig. 11. Real time implementation of proposed technique on insufficient training. Green markers are for recognized actions and red colour markers are for misclassified actions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

dles the recognition technique more efficiently by considering vote counting along with accumulated voting score. LMDI based interest point detection technique is faster than STIP and LMP feature detection. It provides more number of interest points to describe an action with more efficiency. As for less training dataset, the algorithm performance is better, therefore, the proposed technique can be applied for real time HAR. The proposed method is evaluated on three publicly available datasets: UT-interaction segmented dataset, J-HMDB dataset and UCF101 dataset. The algorithm is also tested on real time data acquired from outdoor environment. Experimental results show that the performance of the proposed method is better compared to earlier reported techniques.

Acknowledgment

This Publication is an outcome of the R&D work undertaken in the project under the Visvesvaraya PhD Scheme of Ministry of Electronics & Information Technology, Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia). [grant number PhD-MLA/4(13)/2015-16]

References

Brendel, W., & Todorovic, S. (2011). Learning spatiotemporal graphs of human activities. In *Computer vision (ICCV), 2011 IEEE international conference on* (pp. 778–785). IEEE.

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 *CVPR* (pp. 1932–1939).

Chen, H., Chen, J., Hu, R., Chen, C., & Wang, Z. (2017). Action recognition with temporal scale-invariant deep learning framework. *China Communications, 14*(2), 163–172.

Csurka, G., Dance, C., Fan, L., Willamowski, J., & Bray, C. (2004). Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV: 1* (pp. 1–2). Prague.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR: 1* (pp. 886–893).

Dasgupta, S., & Sinha, K. (2013). Randomized partition trees for exact nearest neighbor search.. In *COLT* (pp. 317–337).

Dollár, P., Rabaud, V., Cottrell, G., & Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. In *International workshop on visual surveillance and performance evaluation of tracking and surveillance* (pp. 65–72).

Efros, A. A., Berg, A. C., Mori, G., & Malik, J. (2003). Recognizing action at a distance. In *CVPR* (pp. 726–733).

Freund, Y., Dasgupta, S., Kabra, M., & Verma, N. (2007). Learning the structure of manifolds using random projections. In *Advances in neural information processing systems* (pp. 473–480).

Gall, J., Yao, A., Razavi, N., Van Gool, L., & Lempitsky, V. (2011). Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(11), 2188–2202.

Gaur, U., Zhu, Y., Song, B., & Chowdhury, A. R. (2011). A "string of feature graphs" model for recognition of complex activities in natural videos. In *ICCV* (pp. 2595–2602).

Guha, T., & Ward, R. K. (2012). Learning sparse representations for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 34*(8), 1576–1588.

Jhuang, H., Gall, J., Zuffi, S., Schmid, C., & Black, M. J. (2013). Towards understanding action recognition. In *Computer vision (ICCV), 2013 IEEE international conference on* (pp. 3192–3199). IEEE.

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.

Junejo, I. N., Dexter, E., Laptev, I., & Perez, P. (2011). View-independent action recognition from temporal self-similarities. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(1), 172–185.

Laptev, I. (2005). On space-time interest points. *International Journal of Computer Vision, 64*(2–3), 107–123.

Laptev, I., Marszalek, M., Schmid, C., & Rozenfeld, B. (2008). Learning realistic human actions from movies. In *CVPR* (pp. 1–8).
Leibe, B., Leonardis, A., & Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision, 77*(1–3), 259–289.
Munoz-Salinas, R., Medina-Carnicer, R., Madrid-Cuevas, F., & Carmona Potayo, A. (2008). Histograms of optical flow for efficient representation of body motion. *Pattern Recognition Letters, 29*, 319–329.
Niebles, J. C., Wang, H., & Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision, 79*(3), 299–318.
Peng, X., Zou, C., Qiao, Y., & Peng, Q. (2014). Action recognition with stacked fisher vectors. In *European conference on computer vision* (pp. 581–595). Springer.
Ryoo, M. S., & Aggarwal, J. K. (2009). Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *ICCV* (pp. 1593–1600).
Ryoo, M. S., & Yu, W. (2011). One video is sufficient? Human activity recognition using active video composition. In *Applications of computer vision (WACV), 2011 IEEE workshop on* (pp. 634–641). IEEE.
Scovanner, P., Ali, S., & Shah, M. (2007). A 3-dimensional sift descriptor and its application to action recognition. In *15th ACM international conference on multimedia* (pp. 357–360).
Seo, H. J., & Milanfar, P. (2009). Detection of human actions from a single example. In *Computer vision, 2009 IEEE 12th international conference on* (pp. 1965–1970). IEEE.
Shi, Y., Tian, Y., Wang, Y., & Huang, T. (2017). Sequential deep trajectory descriptor for action recognition with three-stream CNN. *IEEE Transactions on Multimedia, 19*(7), 1510–1520.
Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv:1212.0402v1.
Yu, G., Yuan, J., & Liu, Z. (2012). Propagative hough voting for human activity recognition. In *European conference on computer vision* (pp. 693–706). Springer.
Yu, G., Yuan, J., & Liu, Z. (2015). Propagative hough voting for human activity detection and recognition. *IEEE Transactions on Circuits and Systems for Video Technology, 25*(1), 87–98.
Yu, S., Cheng, Y., Xie, L., & Li, S.-Z. (2017). Fully convolutional networks for action recognition. *IET Computer Vision, 11*(8), 744–749.