Proceedings of the 2nd International Conference on Trends in Electronics and Informatics (ICOEI 2018)
IEEE Conference Record: # 42666; IEEE Xplore ISBN:978-1-5386-3570-4

Semi-Supervised Learning in Random Forest Classifier for Human Action Recognition

Ulli Srinivasu
Department of EC
NIT Rourkela, India
Email: ullisrinivasu57@gmail.com

Suraj Prakash Sahoo
Department of EC
NIT Rourkela, India
Email: 515ec1003@nitrkl.ac.in

Samit Ari
Department of EC
NIT Rourkela, India
Email: samit.ari@gmail.com

***Abstract*—The objective of human action recognition is the interpretation of ongoing events and context from video data for automated systems. In this paper, motion history image (MHI) is used as the region of interest (ROI) of action during the training phase to recognize human actions effectively. Therefore, the extracted spatio-temporal interest points (STIPs), that are used to train the classifier model, are free from noisy interest points due to the clutter background and illumination changes. After extracting the STIPs, the histogram of oriented gradient (HOG) and histogram of optical flow (HOF) features are calculated for the video patches extracted around the STIPs. Action recognition is performed by calculating mutual information of each STIP with respect to all the action classes provided in the training dataset. Mutual information is calculated by using random forest voting. The trees of random forest are built through proposed semi-supervised learning. The tree nodes are split by using unsupervised learning upto a certain predefined depth of the tree by taking the maximum variance of feature differences of the hypothesis. Next, the splitting process of the nodes is carried out by binary error minimization technique based on supervised learning. The experiments are performed on the standard KTH dataset. The performance of proposed technique is 95% which is better compared to earlier reported methods. Further, similar action classes from Weizmann dataset are tested on the same KTH trained forest model and the results are relevantly comparable with the state-of-the-art methods.**

Keywords: Motion History Image, Mutual information, Random Forest, Region of Interest, Semi-Supervised learning, Spatio-Temporal Interest Points

I. Introduction

Human action recognition (HAR) is one of the core research areas in the field of computer vision and pattern recognition. HAR finds many applications like video surveillance in security systems, sports video analysis to take critical decisions for umpire or referee and human computer interaction for automated systems. HAR is a very challenging task because of background clutter, illumination changes, intra class variations and also processing of large video data. Human action comprises of moments from many body parts. Different actions contain similar body part moments which increases the action recognition complexity.

Many algorithms in literature are reported for human action recognition (HAR). Initially, HAR was developed based on object detection in the video data followed by object tracking and template matching [1, 2]. However, tracking of object is not an easy task in crowded scenes. Also template matching is computationally not efficient. Spatio-temporal patterns [3, 4] are used to represent the action in a video sequences. Ivan Laptev *et al.* [3] extended the Harris spatial interest point detection for the image data to spatio-temporal interest point detection for 3D video data. A silhouette based view independent motion history image (MHI) [5] is proposed for human action recognition. MHI is robust in representing action moments. 3D shape invariant feature transform (SIFT) is proposed by Scovanner *et al.* [4] which is an extended work to 2D SIFT descriptor. BLOB (Binary Large OBject) [2, 6] is used to represent motion object in a video for human action recognition. Histogram of oriented gradient (HOG) [7, 8], Histogram of optical flow (HOF) [9, 10] are computed as features. Nearest neighbor [11] algorithm, support vector machine (SVM) [12, 13] are used for solving the classification problem in earlier methods. In this paper, a random forest [14, 15] voting based classifier technique is employed for multi class action recognition.

The extracted spatio temporal interest points contains noisy points. The noisy STIPs are due to the clutter background and illumination changes which affect the performance of the recognition paradigm. In order to eliminate the noisy points from the extracted STIPs, region of interest (ROI) of motion history image (MHI) [5] is adopted in proposed method. During the training phase, only the STIPs which are in the ROI of MHI are considered. Hence, the STIPs due to noise are not present in the training dataset points. During testing no ROI of MHI is employed and the noisy STIPs can be handled by discriminative voting [1]. It also discriminate intraclass variations. The defined STIPs are described by HOG and HOF features. Now, the feature vector can be used for classification through random forest technique. The classification can be unsupervised [16] or supervised [1] learning. In unsupervised learning, feature vector splits at any node of trees according to the variance of feature difference. After reaching certain depth, when variance is not distinguishable to split, the method fails. Upto the predefined depth of tree unsupervised learning is used to train the tree. After reaching predefined depth of tree the training changes to supervised learning process. Hence, the proposed algorithm got good properties of both supervised and unsupervised learning.

The remaining paper is organized as follows: Section II describes about the details of proposed methodology. Section III discusses the experimental results and comparison of performance with earlier reported methods. Finally section IV concludes the discussion.

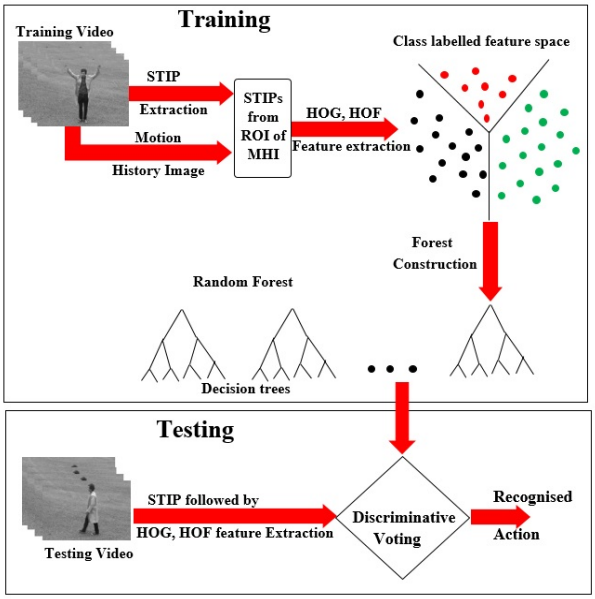

Fig. 1. Overview of random forest classifier based action recognition

II. Methodology

The overview of human action recognition algorithm is shown in Fig. 1. During training, the STIPs are extracted from the training video data. Next, the extracted STIPs are passed through the region of interest of MHI. Hence, the noisy points are eliminated from the extracted STIPs. Further HOG, HOF features are computed for the resulted STIPs video patches. A random forest is built by using proposed semi-supervised learning technique from the feature data space. At the time of testing, STIPs are extracted from test video but the region of interest of MHI is not used. Hence, the test data contains noisy points which are handled by discriminative voting score [16] when passed through the decision trees. Posterior probability is estimated by integrating scores of all trees in the random forest. Mutual information can be estimated using posterior and prior probabilities. Mutual information is estimated for all the action classes. At least one class mutual information of STIP crosses the threshold value, then the corresponding STIP is taken into consideration for action recognition. STIP can be classified to a particular class based on the maximum mutual information of the classes. The final action classification of a test video depends on the maximum number of STIPs classified to a particular class.

A. Interest point detection

The interest points are detected by using STIP detection technique. The extracted STIPs contain noisy points. In order to eliminate the noisy points region of interest of MHI is used. Noisy points are removed when the STIPs are passed through the region of interest of MHI.

1) Spatio-Temporal Interest Points: Interest points provide an abstract or concise representation of patterns in an image. Ivan Laptev extended Harris interest point detection into the spatio-temporal domain as spatio-temporal interest points (STIPs) [3]. STIPs describe the compact representation of video data. The basic idea of interest point is where the video frame values have valid local variations in both space and time. To detect STIPs Laplacian of video data is computed over spatial and temporal scales. Maximization of the normalized Laplacian operator of video results the STIPs. The video sequence is represented with a function v and the convolution of video data with isotropic Gaussian kernel is represented with Q. Gaussian kernel parameters σ_l^2, σ_l^2 are spatial variance, temporal variance respectively.

$$Q(x,y,t;\sigma_l^2,\tau_l^2) = gau(x,y,t;\sigma_l^2,\tau_l^2) * v(x,y,t) \quad (1)$$

Where Gaussian kernel is defined as

$$gau(:;\sigma_l^2,\tau_l^2) = \frac{1}{\sqrt{(2\pi)^3\sigma_l^4\tau_l^2}} \exp(-(x^2+y^2)/2\sigma^2 - t^2/2\tau^2) \quad (2)$$

In general the spatial and temporal events are independent in a video, hence separate scale parameters σ_l^2, τ_l^2 are needed to use in Gaussian kernel. Interest points in the video sequence are the points where v has significant variations in both spatial and temporal dimensions. Interest points are extracted by forming the second order moment matrix, which is a 3×3 matrix formed by averaging the derivatives of a Gaussian function $gau(x,y,t;\sigma_i^2,\tau_i^2)$ convolved with video data v.

$$\mu^+ = gau(x,y,t;\sigma_i^2,\tau_i^2) * \begin{pmatrix} Q_x^2 & Q_xQ_y & Q_xQ_t \\ Q_xQ_y & Q_y^2 & Q_yQ_t \\ Q_xQ_t & Q_yQ_t & Q_t^2 \end{pmatrix} \quad (3)$$

Here, σ_i^2, τ_i^2 are integration scales and related with local scales σ_l^2, τ_l^2 as $\sigma_i^2 = s\sigma_l^2$, $\tau_i^2 = s\tau_l^2$.

The first order derivatives of Q are as follows

$$\begin{aligned} Q_x(x,y,t;\sigma_l^2,\tau_l^2) &= \partial x(gau * v) \\ Q_y(x,y,t;\sigma_l^2,\tau_l^2) &= \partial y(gau * v) \\ Q_t(x,y,t;\sigma_l^2,\tau_l^2) &= \partial t(gau * v) \end{aligned} \quad (4)$$

Search for regions in v containing significant eigenvalues e_1, e_2, e_3 of second order matrix μ^+ in order to detect the interest points. A function H' is formed to find the significant eigenvalues and H' is given by

$$\begin{aligned} H' &= \det(\mu^+) - K\,trace(\mu^+)^3 \\ &= (e_1e_2e_3) - K(e_1+e_2+e_3)^3 \end{aligned} \quad (5)$$

Where K is a constant and choose the K value such that for higher eigenvalues positive local maxima of H' to be found, and that point is an interest point.

2) Motion history image: Motion history image (MHI) [5] is a view based silhouette representation of the history of an object motion in a video. The MHI preserves a history of temporal variations at each pixel position and then decays as the time passes. The MHI representation is concise into grayscale images where dominant motion of object information is preserved. MHI $\mathbb{M}_\tau(x,y,t)$ is computed as follows

$$\mathbb{M}_\tau(x,y,t) = \begin{cases} \tau & if\ D(x,y,t)=1 \\ MAX(0, \mathbb{M}_\tau(x,y,t-1)-1) & otherwise \end{cases} \quad (6)$$

Where $D(x,y,t)$ is binarized image with threshold ξ of consecutive frame difference and represents object's motion in the video, τ represents the time integration for MHI, (x,y) are spatial dimensions and t is temporal dimension of video frame.

B. feature extraction

HOG and HOF features are calculated for extracted video patches around the STIPs. The dimensions of the extracted video patch are determined by the local scales of Gaussian scale parameters σ_l^2 and τ_l^2. The spatial extension $x'=K'\sigma_l^2$ and temporal extension $t'=K'\tau_l^2$ are used to extract the video patch and the value of *K'* is 18 for spatial, 9 for temporal dimensions. HOG descriptor represents shape and appearance of an object. HOG is used in many object recognition algorithms. HOG descriptor is robust to scale and rotation variations. To find the HOG, first calculate gradients of the extracted video. Next, compute the magnitude and orientation angle at each pixel position. split the patch into sub cells and for each cell estimate 4 bin histogram and concatenate all the cells histograms to form HOG descriptor. Divide the orientation angle full range 180^0 into 4 sectors. At each pixel position check in which sector the orientation angle is falling and add the corresponding histogram bin value is angle multiplied with magnitude. Split the each patch such that the feature size of HOG is 72. The gradient magnitude and orientation angle are computed by using following equations

$$mag = \sqrt{g_x^2 + g_y^2} \qquad \theta = \tan^{-1}(\frac{g_y}{g_x}) \quad (7)$$

where, g_x, g_y are gradients in both the spatial dimensions.

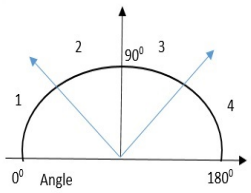

Fig. 2. Dividing full range of orientation angle into 4 sectors

The orientation angle is divided into 4 sectors shown in Fig. 2. Since 4 bin histogram is computing for each cell.

$$b(x) = b(x) + \theta * mag \quad (8)$$

Here, $b(x)$ is bin value in histogram estimation.

HOF descriptor describes the local motion of an object. Optical flow velocities *u,v* are estimated by using Lucas Kanade method along both spatial dimensions *x, y*. Next, similar to the HOG, estimate the magnitude and orientation angles of optical flow. Divide the orientation angle into five sectors to form 5 bin histogram and concatenate all the sub cell histograms to form HOF descriptor of feature size 90.

C. Mutual information based action recognition

The video clip v can be represented with set of STIPs, each STIP $d \in \mathbb{R}^N$ represents the N dimensional feature vector. Action recognition is done by computing mutual information [1] of STIPs with respect to all the action classes *L*(1,2,...,*l*). Mutual information of video clip v is given by

$$M_I(L=l,v) = \log(\frac{\mathbb{P}(v/L=l)}{\mathbb{P}(v)}) \quad (9)$$

$$M_I(L=l,v) = \log(\frac{\prod_{d\in v}\mathbb{P}(d/L=l)}{\prod_{d\in v}\mathbb{P}(d)}) \quad (10)$$

Here, L represents the set of class labels, l represents a particular class label and d is the STIP point of video v and assume that d is independent from other STIP points. Therefore for each STIP, a point wise mutual information computed with a specific class l as follows

$$S^l(d) = \log(\frac{\mathbb{P}(d/L=l)}{\mathbb{P}(d)}) \quad (11)$$

The above equation can be rewritten by using posteriori probability is

$$S^l(d) = \log(\mathbb{P}(L=l/d)) - \log(\mathbb{P}(L=l)) \quad (12)$$

The prior probability $\mathbb{P}(L=l)$ is depends on the number of STIPs present in the training feature dataset for particular class. The posteriori probability $\mathbb{P}(L=l/d)$ is estimated by using random forest.

D. Semi-supervised learning in random forest

Random forest [14, 15, 17] is built by *M* number of independent random decision trees. Random forest is found applications in both regression and classification problems. As the number of trees increases in the forest accuracy increases accordingly. Random forest achieves good accuracy among the currently available algorithms and also handles the large datasets, more number of input variables. The time taken to vote while testing is less. The final voting score is the average of all M decision trees. In this paper semi-supervised learning is proposed to build the tree i.e. tree is build by using both unsupervised as well as supervised learning methods.

Suppose the dataset have N number of STIPs, denoted as $\mho_i = (\mho_i^1, \mho_i^2), i = 1,2,...,N$; $\mho_i^1 \in \mathbb{R}^{72}$ and $\mho_i^2 \in \mathbb{R}^{90}$ are the HOG and HOF features respectively. Decision tree is built by splitting the feature data at current node into left and right childs. In order to split the data, generate a random number $\tau \in \{1,2\}$. τ is 1 for HOG feature and 2 for HOF feature selection. Based on the random number τ either HOG or

HOF feature is selected for splitting. Further, generate two more random numbers (r_1,r_2) as feature dimension indices. After that a feature difference $\mathbb{D}_i$ [1] is evaluated. $\mathbb{D}_i$=$\mho_i^T(r_1)$-$\mho_i^T(r_2)$, *i*=1,2, ..., N. Once the differences are obtained, a threshold is computed to split the data by using two different learning procedures at two different depth levels of the tree as explained in the following sections.

1) Unsupervised learning: At Initial nodes of the tree feature size is very large. Therefore the feature data is split by using unsupervised learning which provides better result [16] upto the predefined depth. Without using the class information of features, simply the variance of the feature difference of hypothesis is computed and the threshold is taken as mean at which maximum variance occurs. The mean and variance are computed by using following formula:

$$\begin{aligned} m(k) &= \sum_{i=1}^{N} \mathbb{D}_i \quad k = 1, 2, ..., 200 \\ \text{var}(k) &= \sum_{i=1}^{N} (\mathbb{D}_i - m(k))^2 \quad k = 1, 2, ..., 200 \end{aligned} \tag{13}$$

The threshold θ is given as

$$\theta = \arg\max_{\text{var}}(m) \tag{14}$$

If the $\mathbb{D}_i \geq \theta$ the corresponding feature data goes to a left child otherwise moves to a right child to current node of the tree. The inference dividing the feature dataset is same for both unsupervised and supervised learning process, but estimation of a threshold is different.

2) Supervised learning: When the tree depth reaches a predefined value, learning procedure of tree is changed to supervised from unsupervised learning. Hence, the name is semi-supervised learning. During supervised learning, binary minimization of error is estimated for the feature differences by using class labels [1] and the threshold value θ is estimated by using following equations

$$\theta^* = \arg\min_{\theta}(\min\{\varepsilon(l)_{left}+\varepsilon(l')_{right}\,,\ \varepsilon(l)_{right}+\varepsilon(l')_{left}\}) \tag{15}$$

Where, $\varepsilon(l)_{left}$ is the misclassification error due to the left node and $\varepsilon(l)_{right}$ is the misclassification error due to right node when the labels of both the tree nodes are *l*. $\varepsilon(l')_{left}$ is the misclassification error due to the left node and $\varepsilon(l')_{right}$ is the misclassification error due to right node when the labels of both the tree nodes are not *l*. The error function $\varepsilon(l)_{left}$ is computed as

$$\varepsilon(l)_{left} = \sum_{i=1}^{N} I(y_i \neq l) \quad if \ \ \mathbb{D}_i > \theta \tag{16}$$

Here the function I($y_i \neq l$) is equal to 1 when ($y_i \neq l$) is true otherwise zero and y_i is the class label. Similarly other error terms are also estimated. Tree splitting procedure is stopped when the maximum depth of the tree is reached or the minimum number of features are reached at the current node.

To compute posterior probability $\mathbb{P}(L = l/d)$, add the information from all the tree leaves which contain d. Suppose for tree T_k the STIP d matched to a leaf with Z_k^+ positive query STIP points and Z_k^- negative points [16] and Z_T is the total number of trees, then $\mathbb{P}(L = l/d)$ is computed as

$$\mathbb{P}(L = l/d) = \frac{1}{Z_T} \sum_{k=1}^{Z_T} \frac{Z_k^+}{Z_k^+ + Z_k^-} \tag{17}$$

III. Experimental results and discussions

Experiments are conducted on the standard KTH dataset to evaluate the proposed algorithm. The KTH dataset having 6 different classes of actions, 25 different persons are performed under 4 scenarios. The experimental setup is similar to [1, 3, 11, 16]. Out of 25 videos clips, 16 video clips are used to train the random forest model and 9 video clips for testing.

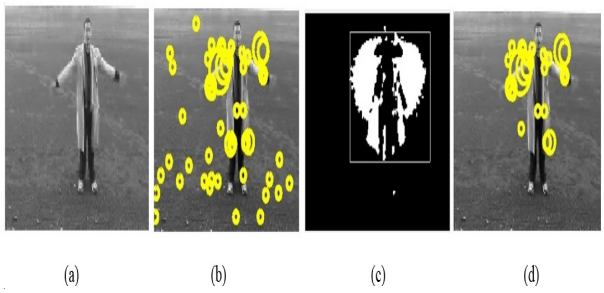

Fig. 3. Extracted STIPs for handwaving class. (a) video frame (b) Extracted STIPs without MHI, contain noisy points (c)Region of interest of MHI of corresponding video frame (d) Extracted STIPs after passing through region of interest of MHI, noisy points are not present.

The extracted STIPs are shown in Fig. 3 (b) and Fig. 3 (a) is corresponding video frame. The extracted STIPs contains noisy points along with action points in Fig. 3 (b). The region of interest of MHI is shown in Fig. 3 (c). Only the points which are fallen in the region of interest of MHI are taken into account. The noise free extracted STIPs are shown in Fig. 3 (d). Next, HOG and HOF features are computed for the resulted STIPs to train the random forest. The mutual information due to prior probability is computed by using the equation $\log(\mathbb{P}(L = l))$. Since the testing video clips are independent but the value $\log(\mathbb{P}(L = l))$ is varies with number of STIPs present in each action class of the training dataset. The values are obtained by experimentally for boxing, handwaving, handclapping, running, jogging and walking as: -1.8123, -1.792, -1.8412, -1.81, -1.75 and -1.57 respectively. In order to discriminate intra class actions and noisy points, while giving voting score to STIPs, interest points from negative class actions are considered as shown in eqaution (17). In the experiments for boxing, handwaving and handclapping action classes, walking class is considered as negative class and for running, jogging and walking action classes, boxing class is considered as negative class.

TABLE I
CONFUSION MATRIX ON KTH DATASET

Each 36 testing data	Boxing	Handwaving	Handclapping	Running	Jogging	Walking
Boxing	36	0	0	0	0	0
Handwaving	1	35	0	0	0	0
Handclapping	0	0	36	0	0	0
Running	0	0	0	34	2	0
Jogging	1	0	0	7	29	0
Walking	0	0	0	0	0	36

TABLE II
CLASSIFICATION ACCURACY RESULTS FOR KTH DATASET

S.NO	Methods	Accuracy
1	A. M. Nickfarjam [18]	90.15%
2	Laptev [3]	91.8%
3	Gang Yu [1]	91.8%
4	Miao Wang [19]	94.5%
5	Proposed Method	95%

The confusion matrix on KTH dataset is shown in Table I. The jogging class is confusing with running class. Since running and jogging are similar kind of activities, otherwise the performance of the proposed algorithm is more accurate. We got 95% accuracy for action recognition with standard KTH dataset. The comparison of results are shown in Table II

The performance of the proposed algorithm is cross validated with Weizmann dataset are shown in table III. From Weizmann dataset similar action classes like walking, running and handwaving video clips are tested with the tree model, which is trained by the KTH dataset and got an accuracy of 96.5%.

TABLE III
CONFUSION MATRIX ON WEIZMANN DATASET WITH KTH DATASET TRAINING

Each 9 testing data	Boxing	Handwaving	Handclapping	Running	Jogging	Walking
Handwaving	9	0	0	0	0	0
Running	0	0	0	8	1	0
Walking	0	0	0	0	0	9

IV. CONCLUSION

In this work, MHI is proposed to combine with STIP detection technique to remove the noisy interest points. The technique handles the noisy points and so handles the clutter background, illumination changes and intra class variation more efficiently. Semi-supervised learning technique is proposed for training in random forest. The unsupervised training is followed by supervised training used to split feature data at nodes by considering class labels. The defined interest point detection technique along with semi-supervised learning has improved the performance of HAR. The proposed method provides an accuracy of 95%. Which is better than the state-of-the-art methods. The algorithm is cross verified from similar actions of Weizmann dataset and the accuracy of 96.5% for three such classes proves the efficiency of the proposed algorithm.

REFERENCES

[1] G. Yu, N. A. Goussies, J. Yuan, and Z. Liu, "Fast action detection via discriminative random forest voting and top-k subvolume search," *IEEE Transactions on Multimedia*, vol. 13, no. 3, pp. 507–517, 2011.
[2] P. V. K. Borges, N. Conci, and A. Cavallaro, "Video-based human behavior understanding: A survey," *IEEE transactions on circuits and systems for video technology*, vol. 23, no. 11, pp. 1993–2008, 2013.
[3] I. Laptev, "On space-time interest points," *International journal of computer vision*, vol. 64, no. 2-3, pp. 107–123, 2005.
[4] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007, pp. 357–360.
[5] M. A. R. Ahad, J. K. Tan, H. Kim, and S. Ishikawa, "Motion history image: its variants and applications," *Machine Vision and Applications*, vol. 23, no. 2, pp. 255–281, 2012.
[6] J. Candamo, M. Shreve, D. B. Goldgof, D. B. Sapper, and R. Kasturi, "Understanding transit scenes: A survey on human behavior-recognition algorithms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 1, pp. 206–224, 2010.
[7] C. P. Huang, C. H. Hsieh, K. T. Lai, and W. Y. Huang, "Human action recognition using histogram of oriented gradient of motion history image," in *2011 First International Conference on Instrumentation, Measurement, Computer, Communication and Control*, Oct 2011, pp. 353–356.
[8] F. Murtaza, M. H. Yousaf, and S. A. Velastin, "Multi-view human action recognition using histograms of oriented gradients (hog) description of motion history images (mhis)," in *2015 13th International Conference on Frontiers of Information Technology (FIT)*, Dec 2015, pp. 297–302.
[9] K. Lertniphonphan, S. Aramvith, and T. H. Chalidabhongse, "Human action recognition using direction histograms of optical flow," in *2011 11th International Symposium on Communications Information Technologies (ISCIT)*, Oct 2011, pp. 574–579.
[10] B. Jagadeesh and C. M. Patil, "Video based action detection and recognition human using optical flow and svm classifier," in *2016 IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, May 2016, pp. 1761–1765.
[11] J. Yuan, Z. Liu, and Y. Wu, "Discriminative subvolume search for efficient action detection," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 2442–2449.
[12] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
[13] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3. IEEE, 2004, pp. 32–36.
[14] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
[15] F. Schroff, A. Criminisi, and A. Zisserman, "Object class segmentation using random forests." in *BMVC*, 2008, pp. 1–10.
[16] G. Yu, J. Yuan, and Z. Liu, "Unsupervised random forest indexing for fast action search," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 865–872.
[17] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.
[18] M. Wang, J. Sun, and J. Yu, "Human action recognition based on feature level fusion and random projection," in *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*, Dec 2016, pp. 767–770.
[19] A. M. Nickfarjam and H. Ebrahimpour-Komleh, "Shape-based human action recognition using multi-input topology of deep belief networks," in *2017 9th International Conference on Information and Knowledge Technology (IKT)*, Oct 2017, pp. 1–4.