How to cite this article:

Basavaiah, J., & Patil, C. M. (2020). Human activity detection and action recognition in videos using convolutional neural networks. *Journal of Information and Communication Technology, 19(2),* 157-183. https://doi.org/10.32890/jict2020.19.2.1

HUMAN ACTIVITY DETECTION AND ACTION RECOGNITION IN VIDEOS USING CONVOLUTIONAL NEURAL NETWORKS

Jagadeesh Basavaiah & Chandrashekar Mohan Patil
Department of Electronics and Communication Engineering,
Vidyavardhaka College of Engineering, India

jagadeesh.b, patilcm@vvce.ac.in

ABSTRACT

Human activity recognition from video scenes has become a significant area of research in the field of computer vision applications. Action recognition is one of the most challenging problems in the area of video analysis and it finds applications in human-computer interaction, anomalous activity detection, crowd monitoring and patient monitoring. Several approaches have been presented for human activity recognition using machine learning techniques. The main aim of this work is to detect and track human activity, and classify actions for two publicly available video databases. In this work, a novel approach of feature extraction from video sequence by combining Scale Invariant Feature Transform and optical flow computation are used where shape, gradient and orientation features are also incorporated for robust feature formulation. Tracking of human activity in the video is implemented using the Gaussian Mixture Model. Convolutional Neural Network based classification approach is used for database training and testing purposes. The activity recognition performance is evaluated for two public datasets namely Weizmann dataset and Kungliga Tekniska Hogskolan dataset with action recognition accuracy of 98.43% and 94.96%, respectively. Experimental and comparative studies have shown that the proposed approach outperformed state-of-the art techniques.

Received: 20/12/2018 **Revised:** 20/5/2019 **Accepted:** 26/5/2019 **Published:** 31/3/2020

Keywords: Action recognition, convolutional neural network, Gaussian Mixture Model, optical flow, SIFT feature extraction.

INTRODUCTION

A dramatic growth has been noticed in various technologies in which computer vision technique is one of the most powerful and widely used techniques for real-time applications such as traffic monitoring, facial recognition and surveillance systems. Due to the rapid increase in population, the development of effective surveillance systems are highly demanded to ensure security in crowded regions. In the field of computer vision, action recognition and classification are widely researched areas and utilized in various applications such as surveillance systems, human-to-computer interactions, video retrieval, etc. Recognition of various actions is an active research area in the field of computer vision. Due to increased demand for accurate action recognition, it has become a more complex task for researchers (Varol, Laptev, & Schmid, 2018).

Several techniques have been introduced for real-time action recognition. Transferable Belief model is introduced for action recognition in video scenes (Ramasso, Panagiotakis, Pellerin, & Rombaut, 2007). Conventional techniques of action recognition are focused on single feature extraction models which are portioned as global features (Shao, Zhen, Tao, & Li, 2014) and local features (Laptev, Marszalek, Schmid, & Rozenfeld, 2008). Spatio-temporal scheme of feature extraction is used for supervised learning scheme (Dollár, Rabaud, Cottrell, & Belongie, 2019). In this spatio-temporal approach features are extracted which lead to extraction of 3D oriented gradient (3D HOG), and 3D scale invariant feature transform and histogram of optical flow are extracted. Based on global feature extraction model, discriminative features are used for analysis by combining features along with temporal and spatial dimensions. Generally, action recognition techniques are categorized into two models which are known as (a) learning based models and (b) template based models of action recognition (Ji, Xu, Yang, & Yu, 2013). According to the learning based techniques, a huge reliable database is required to formulate the classifier model. Similarly, for template based matching approach, single template is used to find the similarity between query video and database. However, a huge amount of work has been carried out for action recognition but still there is a performance gap due to various issues such as occlusion, image scaling, cluttering and variations in object appearance, etc. Due to these issues, conventional feature or template matching algorithm fails to obtain the desired performance of action recognition. Hence, the development of an efficient approach becomes a challenging task for researchers.

During action recognition, action classification plays an important role. The classification scheme follow pattern learning scheme which can be categorized into two main categories: stochastic model and statistical model similar to previous work (Dollár, Rabaud, Cottrell, & Belongie, 2019). Spatio-temporal techniques are used for action recognition (Wong & Cipolla, 2007) and conventional techniques discard global information which may lead to improper results. To overcome these issues, a novel approach is presented here which includes global information during feature extraction such as blobs and moving pixels, etc. which help to identify moving objects resulting in better performance for action recognition. For robust performance in action recognition, adaptive computation approaches such as Hidden Markov Model is also used (Moghaddam & Piccardi, 2014). In this process, various techniques such as Support Vector Machine (Varol, Laptev, & Schmid, 2018; Li, Zhang & Liao, 2017), AdaBoost (Zhang et al., 2017) and Naïve Bayes (Zhen, Zheng, Shao, Cao, & Xu, 2017) have been proposed. According to these studies, human action recognition in videos has also attracted researchers due to its potential applications. However, these feature extraction techniques depend on visual pattern analysis.

These visual patterns are known as local features and deep-learned features (Wang, Qiao, & Tang, 2015). Local features include trajectory computation (Wang, Kläser, Schmid, & Liu, 2013), cuboids, and space time interest points (Liu, Chen, & Liu, 2017). These feature computation processes are decomposed into two main stages, viz. detector and descriptor. A detector module helps to compute the salient features of an input scene for action recognition whereas descriptors describe the visual feature pattern of an input scene. Similarly, for deep learned-features, deep learning computation approach is used which includes various convolutional networks for processing such as 3D ConvNets (Ji, Xu, Yang, & Yu, 2013), Deep ConvNets (Karpathy et al., 2014), Convolutional RBMs (Karpathy et al., 2014), and Two stream ConvNets (Taylor, Fergus, LeCun, & Bregler, 2010). However, image classification studies showed that the local features achieved improved performance when compared with the deep-learned features. Deep learning models of action recognition require more numbers of input label videos for the training process; however, available datasets are comparatively small. Moreover, current approaches of deep learning methods, discard information related to spatial and temporal domains resulting in degraded performance in action recognition (Krizhevsky, Sutskever, & Hinton, 2017).

There is a need to develop an efficient approach for action recognition which can provide better accuracy with low-complexities. A new improvised cascaded approach is introduced for feature extraction, which uses optical flow computation combined with SIFT features where spatial and temporal feature extraction are implemented and later Convolution Neural Network

(CNN) classification is used for activity classification. The main components of the work are as follows:

1. In order to estimate the trajectory, optical flow computation approach is used which is further refined by using streak line flow.
2. In the next phase, feature extraction is applied using SIFT features including shape, orientation and gradient computation techniques which provide a robust feature vector.
3. Finally, CNN based machine learning approach is implemented for activity classification and recognition using learned feature patterns.

RELATED WORK

Various researches have been carried out in this field of action recognition using pattern learning based computer vision approaches. In this section, some recent techniques of action recognition are discussed. Human action recognition is a vast area in the computer vision field which has several potential surveillance applications in real-time systems. Aggarwal and Ryoo (2011) conducted an extensive survey which examined all recent studies in surveillance systems which showed recent advancements in surveillance systems with the help of human action recognition systems. Various schemes have been discussed which are based on object trajectory.

Trajectory computation plays an important role for object detection in video applications. These models are based on the Kanade–Lucas–Tomasi (KLT) tracker between alternate frames. Due to insufficient feature extraction, the performance of these systems degrades which may lead to performance degradation. In order to deal with these issues, image classification techniques have been considered as promising. In this process, dense point features are extracted and sampled; tracking of the features is done using optical flow computation (Wang, Kläser, Schmid, & Liu, 2013). Likewise, particle trajectory based method is introduced for motion decomposition to recognize human activity. Moving videos are complex in nature and require a better pre-processing module with motion compensation, object detection and tracking. Lagrangian particle trajectories based technique is developed by computing optical flow computation to deal with moving camera object detection tasks; in addition moving camera issues could be addressed with the help of low rank optimization (Wu, Oreifej, & Shah, 2011).

Recently, machine learning and sensing hardware devices are used for activity recognition algorithm. Hardware devices cause implementation cost complexity, hence low cost and low power devices have been recommended for use. In order to present machine learning approach, semi-Markov Random

Field is used. These activity recognitions require an efficient modeling of visual features and their correlation. Generally, these features are in the form of 2D feature models which sometimes suffer due to inefficient modeling and context between individual features resulting in degraded performance. 3D feature representation is a significant technique which can provide a better modeling of individual features (Lee et al., 2011). To achieve this, an approach is introduced which uses fusion scheme for both 3D depth sensor image and grayscale image. Further, for better visual feature analysis, depth based filters are applied which helps to identify and remove false detections. Later, 3D modeling of spatial and temporal features is extracted for fused image. During this process, complete information about video is partitioned into various parts which are later accumulated using a structure level modeling (Burghouts & Schutte, 2013).

However, human action video sequence may contain various occlusions, cluttered environments which increase complexity for human action recognition. During video acquisition, if camera motion is fast then it also becomes a challenging task for the identification of action. To overcome this, (Jian, Drew, & Li, 2010) convex match based approach is developed. A highly non-convex video problem can be converted into smaller linear problems and later, can be resolved using successive approaches.

In the field of human activity recognition, a huge amount of work has been carried out. These techniques mainly depend on the feature extraction process i.e. if robust features are extracted, it can provide a better analysis of any given input video frame. However, conventional techniques suffer from complexity and important feature extraction schemes. Some significant features are also discarded during feature reduction modeling which may lead to degraded performance.

A new video representation called trajectory-pooled deep-convolutional descriptor (TDD) (Wang, Qiao, & Tang, 2015) shares the advantages of both hand-crafted features and deep-learned features. Specifically, deep architectures are used to learn discriminative convolutional feature maps, and conduct trajectory-constrained pooling to aggregate these convolutional features into effective descriptors. To improve the robustness of TDDs, two normalization methods to transform convolutional feature maps, namely spatio-temporal normalization and channel normalization are designed. The advantages of these features come from (i) TDDs which are automatically learned and contain highly discriminative capacity compared with those hand-crafted features; (ii) TDDs take account of the intrinsic characteristics of temporal dimension and introduce the strategies of trajectory-constrained sampling and pooling for aggregating deep-learned features.

A lightweight action recognition architecture based on deep neural networks is addressed by just using RGB data (Wang, et al, 2018). The architecture comprises CNN, long short-term memory (LSTM) units, and temporal-wise attention model. First, the CNN is used to extract spatial features to differentiate objects from the background with both local and semantic characteristics. Next, two kinds of LSTM networks are accomplished on the spatial feature maps of different CNN layers (pooling layer and fully-connected layer) to extract temporal motion features. Then, one temporal-wise attention model is designed after the LSTM to learn which parts in which frames are more important. Lastly, a joint optimization module is designed to explore intrinsic relations between two kinds of LSTM features.

A new deep learning network for action recognition that integrates quaternion spatial-temporal convolutional neural network (QST-CNN) and Long Short-Term Memory network (LSTM), called QST-CNN-LSTM (Meng, Liu, & Wang, 2018). Unlike a traditional CNN, the input for a QST-CNN utilizes a quaternion expression for an RGB image, and the values of the red, green, and blue channels are considered simultaneously as a whole in a spatial convolutional layer, avoiding the loss of spatial features. As the raw images in video datasets are large and have background redundancy, pre -extraction of key motion regions is conducted from RGB videos using an improved codebook algorithm. Furthermore, the QST-CNN is combined with LSTM for capturing dependencies between different video clips.

The low-level feature-based framework for human activity recognition includes feature extraction and descriptor computing, early multi-feature fusion, video representation, and classification. A spatio-temporal bigraph-based multi-feature fusion algorithm (Yao, Liu, & Huang, 2016) is proposed to capture useful visual information for recognition. Dense trajectory features are extracted from the videos and each feature is encoded to three different descriptors which are Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF), and Motion Boundary Histogram (MBH). The features are sampled and clustered into *k*-visual words. Then, a spatio-temporal biograph is constructed and an efficient *k*-way segmentation algorithm is employed to segment the graph. Visual words with strong spatio-temporal relationships are fused while visual words with weak spatio-temporal relationships are segmented and support vector machine (SVM) is used for action recognition.

PROPOSED MODEL

A new approach for human activity recognition from video scenes is proposed by developing an enhanced technique of feature extraction. Initially, optical flow computation scheme for video information extraction is applied and temporal information of input sequence is obtained. In optical flow, sequential frames are considered for analysis and initial optical flow vectors need to be calculated for these frames. This provides significant information about the flow in the image. Figure 1 shows a block diagram of the proposed model.

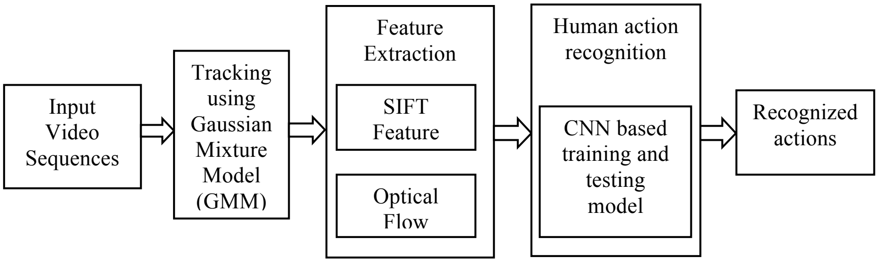


Figure 1. Block diagram of the proposed model.

Optical Flow Estimation

Optical flow refers to the pattern of deceptive movements of objects in a visual scene. In order to model and compute the optical flow, Lucas-Kanade approach is considered for a given video sequence. Video sequences have huge information which can be analyzed efficiently via trajectory information extraction using the optical flow computation approach. The algorithm for optical flow computation is as shown in Figure 2.

Optical flow methods attempt to calculate the motion between two image frames which are taken at times t and $t + \delta t$ at every voxel position. These methods are called differential since they are based on local Taylor series approximations of image signals; that is, they use partial derivatives with respect to spatial and temporal coordinates.

Video frame corners are also considered for analysis and hence this process helps to preserve information during trajectory computation and feature extraction. Here, corner points can be detected and information can be extracted by optimizing the problem and is given by Equation 1.

$$\in (\delta x, \delta y) = \mathbb{I}(x, y) - \mathbb{I}(x + \delta x, y + \delta y) \tag{1}$$

where $\in (\delta x, \delta y)$ is the displacement across two frames in time, $I(x, y)$ is the previous frame with some intensity and $I(x + \delta x, y + \delta y)$ is the current frame with shifted intensity.

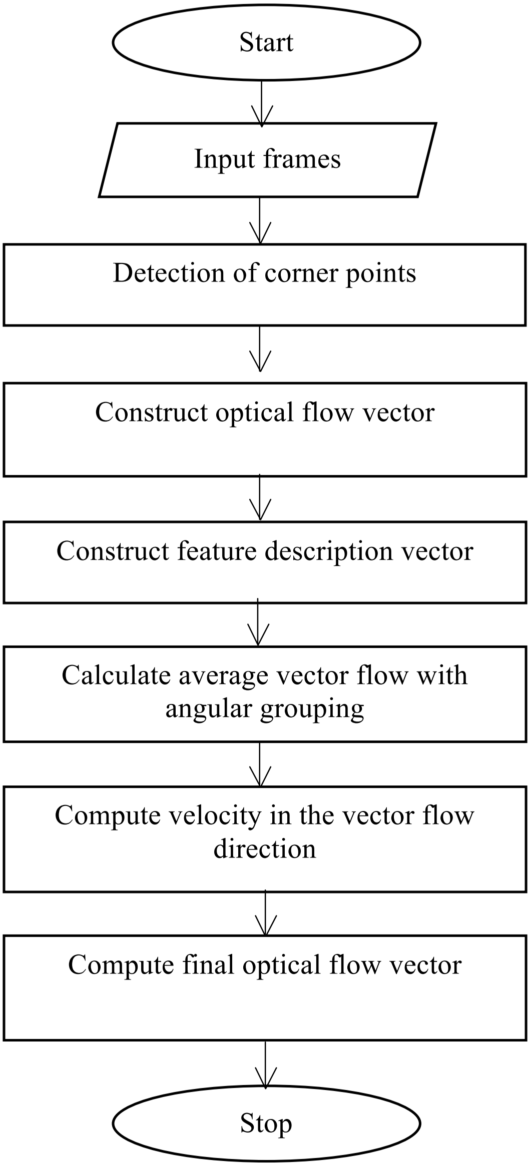

Start
Input frames
Detection of corner points
Construct optical flow vector
Construct feature description vector
Calculate average vector flow with angular grouping
Compute velocity in the vector flow direction
Compute final optical flow vector
Stop

Figure 2. Flowchart of optical flow computation algorithm.

Appropriate parameter selection of δx and δy are used to obtain a better optical vector which contains sufficient temporal information. Here, video aperture problem remains unaddressed which may lead to inappropriate temporal information extraction resulting in degraded feature extraction for video analysis. In order to deal with this issue, neighboring points of current pixel locations are also considered for computation process and this function is given by Equation 2

$$\in (\delta x, \delta y) = \sum_{u_y - w_y}^{u_y + w_y} \sum_{u_x - w_x}^{u_x + w_x} [\mathbb{I}(x, y) - \mathbb{I}(x + \delta x, y + \delta y)] \quad (2)$$

Summation of these vectors in $x - y$ direction can provide a solution for aperture problems. With the help of a window w , centered at point (x, y) the neighboring points are calculated and estimated. In this work, optical flow vector computation process is used for each frame and these flow vectors are combined to formulate the feature vector which contains trajectory information. Optical flow vector set can be constructed and is given by the Equation 3,

$$R = [S(V), \phi] \quad (3)$$

where $S(V)$ denotes set of optical flow vector and ϕ denotes descriptor vector. These vectors are helpful for identifying the relationship between optical flow feature vector and sequential frame. With the help of these relationships, temporal information of video can be extracted. This is a generic process which can be adapted easily for various other operations. This technique is widely used for optical flow representation of video sequences. In optical flow computation, histogram based optical flow computation process shows significant results by splitting the histograms. These histograms are used to formulate the feature vector. However, conventional methods of optical flow computation have impact on video analysis; however, performance can still be improved through refining the feature vectors by preserving the edge information.

Video scenes contain huge amounts of temporal information which may face dimensionality issues. hence a new model for optical flow feature representation is presented by considering feature magnitudes. To deal with this issue, average frame velocity concept is developed to interpret information of input video sequence. Conventional approach in histogram computation considers similarity as its segments, whereas it needs to be computed in the form of mean of histograms. According to the proposed approach, optical flow vectors for each frame are considered and computed at the first step. This can be obtained by using Equation 3 where generic representation of optical flow is achieved. In order to construct the feature representation model, a descriptor vector is used and is given by Equation 4

$$S(f_v, \alpha, \beta) = \{V(r, \angle\sigma) \in f_v | \mu < \sigma \le \omega\} \quad (4)$$

where $S(f_v)$ denotes total number of optical flow vector in current video frame f, $S(V_f, \mu, \omega)$ denotes set of optical flow vector which has the angle between μ and ω in current frame. If the input video sequence is denoted by V where total number of frames is given by F , total angle intervals are denoted by m and length of video V is l in the terms of seconds, then the average vector flow with the angular grouping can be given by Equation 5.

$$R = \left[\left\| \sum_{S(f_{\forall v}, \mu_1, \mu_2)} V(r, \angle\sigma) \right\|, \left\| \sum_{S(f_{\forall v}, \mu_2, \mu_3)} V(r, \angle\sigma) \right\| \ldots, \left\| \sum_{S(f_{\forall v}, \mu_m, \mu_{m+1})} V(r, \angle\sigma) \right\|\right] \tag{5}$$

where $V(r, \angle\sigma)$ denotes a vector of optical flow which contains a magnitude r and angle σ. The above given expression is used to denote the optical flow vector. However, this representation lacks in terms of velocity which is improved by including a velocity computation vector. Velocity is a main component which affects the analysis of various activities such as walking, running standing, etc. In order to deal with this, average weight frame velocity computation model is presented. In this model, the velocity of each video segment is computed in the given vector flow direction and is expressed by Equation 6.

$$A(\mu, \omega) = \frac{\sum_{i=0}^{n-1} \left\lfloor \left\| \sum_{S(f_v, \mu, \omega)} V(r, \angle\sigma) \right\| . \left| S(f_v, \mu, \omega) \right| \right\rfloor}{\sum_{i=0}^{n-1} \left| S(f_v, \mu, \omega) \right|} \tag{6}$$

where $A(\mu, \omega)$is the velocity of each video segment computed in the given vector flow direction. The average frame velocity is also considered as a component of feature representation, with the help of this analysis, a final optical flow vector can be represented which is given by Equation 7.

$$R_f = \left[A(\mu_1, \mu_2) \left\| \sum_{S(f_v, \alpha_1, \alpha_2)} V(r, \angle\varphi) \right\|, A(\mu_2, \mu_3), \left\| \sum_{S(V_f, \alpha_2, \alpha_3)} V(r, \angle\varphi) \right\|, \ldots, A(\mu_m, \mu_{m+1})\right] \tag{7}$$

where R_f is the final optical flow vector, $A(\mu_1, \mu_2)$ is the velocity in vector flow direction, $V(r, \angle\varphi)$ vector of optical flow consisting of magnitude and direction.

Feature Extraction

Feature extraction refers to the process of extracting informative characteristics called features from a video frame. There is a variety of feature extraction techniques such as Histogram of Oriented Gradients (HOG), Speeded-Up Robust Features (SURF), Local Binary Patterns (LBP), Haar Wavelets, Color histograms, Scale Invariant Feature Transform (SIFT) and Space-Time Interest Points (STIP). The SIFT technique of feature extraction is one of the classic techniques and is more accurate than other feature descriptors which are rotation and scale invariant. SIFT features are local and has an ability to find distinctive keypoints that are invariant to location, scale and rotation, and robust to affine transformations. In addition to these properties, they are highly distinctive, relatively easy to extract and allow for correct object identification with low probability of mismatch.

SIFT Feature Extraction

SIFT feature extraction approach obtains scale-invariant features with the help of a staged filtering technique. In this process, key locations are identified by computing the maxima and minima of a Difference-of-Gaussian (DoG) function. Each point helps to generate the feature vector which can describe the complete local region of image. This feature information contains local variations, image projections and image gradients and the resulting set of features is called SIFT keys. In this process, image is transformed into pyramid which can be used for feature extraction using re-sampling at each level of input image. The flowchart of the SIFT algorithm is as shown in Figure 3.

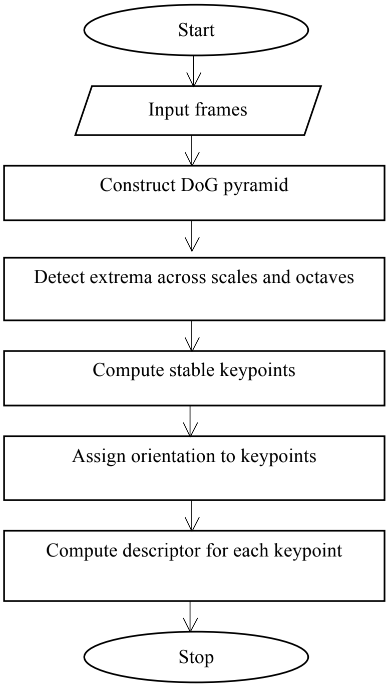

Start
Input frames
Construct DoG pyramid
Detect extrema across scales and octaves
Compute stable keypoints
Assign orientation to keypoints
Compute descriptor for each keypoint
Stop

Figure 3. Flowchart of SIFT algorithm.

Shape Feature Extraction

A brief description of shape feature extraction is presented here which is carried out by using the following stage computations: edge map computation, orientation estimation and curvature computation.

Edge Map Computation

First stage of this process is to compute the edges of any given input image. A robust Canny edge detector is used which provides the edge map of each image by traversing through each pixel of the input image.

Orientation Estimation

Once the edge map of the input image is computed, orientation computation for image map gradient estimation is carried out which can be calculated as given by Equation 8.

$$\Theta(p,q) = atan2\left(\frac{\partial I}{\partial q}(p,q), \frac{\partial I}{\partial p}(p,q)\right) \tag{8}$$

where $\Theta(p,q)$ denotes the range of orientation computation given as $(-\pi,\pi]$ and (p,q) denotes the position of pixel for which orientation and gradients are computed.

Curvature Computation

Curvature of any edge map is given by the Equation 9

$$\mathcal{C}(x,y) = D(\Theta_{min}(x,y), \Theta_{max}(x,y) \tag{9}$$

where $\mathcal{C}(x,y)$ is the curvature of edge map, $\Theta_{max}(x,y)$ is the minimum orientation, $\Theta_{min}(x,y)$ is the maximum orientation and $D(\Theta_{min}(x,y), \Theta_{max}(x,y))$ is the difference between the minimum and maximum orientation.

Overview of SIFT algorithm

Keypoints can be extracted since features are invariant to the image rotation, translation and scale. Main stages of this technique are: Scale space extrema detection, Keypoint localization, Orientation assignment, and Keypoint descriptor.

Scale space Extrema Detection

This is the first stage of SIFT feature extraction implementation where keypoints are extracted which are invariant to change of scale. In order to extract the stable features, Gaussian function is applied. Let $L(x,y,\sigma)$ be a scale-space function of an image, obtained by applying convolution on image $I(x,y)$convolution with a Gaussian function $G(x,y,\sigma)$. For keypoint location estimation, Difference-of-Gaussian function is used which is denoted by

$D(x, y, \sigma)$. Extrema is computed by taking the difference of nearby scales given by Equation 10.

$$\begin{aligned} D(x, y, \sigma) &= G(x, y, k\sigma) * I(x, y) - G(x, y, \sigma) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{10}$$

where kk is a constant factor.
This image is processed in octave manner where in each octave convolution is applied resulting in scale space images. Similarly, DoG is produced by subtracting the adjacent Gaussian images. After applying octave process, images are down sampled by factor of 2.

Keypoint Localization

Interpolation approach is conducted on detected candidate keypoints which helps to obtain the position of keypoints. Furthermore, keypoints which are prone to noise, and contains lower contrast results in performance degradation will be eliminated.

Orientation Assignment

Orientation assignment is used to obtain invariance image rotation relative to keypoint descriptors. Since image samples are Gaussian smoothed, keypoints are selected; orientation, magnitude mm and gradients are computed as given by Equation 11.

$$m(x, y) = \sqrt{\left(L(x+1, y) - L(x-1, y)\right)^2 + \left(L(x, y+1) - L(x, y-1)\right)^2} \tag{11}$$

Consider two images are denoted by I_1 and I_2 respectively. The keypoints detection of these two images uses a saliency based scheme to represent the matching of keypoints. The detected keypoint of image I_1 is denoted as P_{I_1} and for the image, P_{I_2} respectively. In order to find the match between key points of the first image P_{I_1} and the second image keypoint, P_{I_2},the nearest neighbor of P_2 and the Euclidean distance based criteria are used which is given by Equation 12.

$$\frac{d\left(P_{I_1}, P_{I_2}\right)}{d\left(P'_{I_1}, P_{I_2}\right)} < t \tag{12}$$

where t denotes the threshold for detection and P'_{I_1} denotes the second nearest neighbor distance to P_{I_2}.

Motion Descriptor Matching

In this work, along with action detection, human tracking is also performed using Gaussian Mixture Model technique. The Gaussian Mixture Modeling technique is used for motion modeling and its variations during tracking is used to compute motion descriptors. Initially, video stream is converted into multiple frames and optical flow computation is applied on the extracted frames. The extracted optical flow F is partitioned into two scalar fields:F_x and F_y where x and y denote the direction of optical flow. Further, these vectors are divided into four non-negative channels corresponding to each direction as $F_x = F_x^+ - F_x^-$ and $F_y = F_y^+ - F_y^-$. After obtaining these channels, Gaussian kernel scheme is applied and this provides the complete information of all the four channels. For further matching of the motion descriptor, these channels are also considered where each channel is divided into various patches such as p_{x1}, p_{x2}, p_{x3} and p_{x4} and these patches are concatenated for vector formulation. Similarly, patches are divided for y direction as p_{y1}, p_{y2}, p_{y3} and p_{y4}. For robust distance matching between patches of each video clip, the mean of each patch direction is computed as $\hat{a}_k = [p_{x1}^1 - \bar{p}_{x1}, p_{x2}^2 - \bar{p}_{x2}, \dots, p_{xn}^n - \bar{p}_{xn}]$ and $\hat{a}_k = [p_{y1}^1 - \bar{p}_{y1}, p_{y2}^2 - \bar{p}_{y2}, \dots, p_{yn}^n - \bar{p}_{yn}]$. Hence, the similarity between x and y direction can be computed using Equation 13.

$$d(i,j) = C - \sum_{k=1}^{4} \frac{\hat{a}_k^T \, \hat{b}_k + \varepsilon}{\sqrt{(\hat{a}_k^T \, a_k + \varepsilon)(\hat{b}_k^T \, \hat{b}_k + \epsilon}} \tag{13}$$

where C denotes the positive constant which is used to make distance as a non-negative value and ε is a small constant. This distance measure is used for identifying the most similar video clip which is further processed for feature computation and convolutional neural network classifier.

CNN Classification Model

Convolutional Neural Networks (CNN) is considered as a special type of neural network which is based on the feedforward neural network processing. According to the process of CNN (Farhadi & Tabrizi, 2008), prior knowledge of data attributes is incorporated into the CNN architecture. Video clips are considered as input signals where action recognition and classification are performed because of the robust nature of CNN for pause variations for 2D shape recognition. CNN models utilize spatial subsampling which helps to ensure scale shift, data deformation and combines local features with neural network weights. In this process, CNN models can extract local simple visual features such as end-points and corner edges. In the next phase, these features are passed to the succeeding layer to identify the more complex features.

Generally, convolutional neural networks contain set of the layer which contains various layers along with one or more planes for computation which is connected to the local neighborhood of the previous layer. These units are also considered as local feature detector whose activation functions are determined at the learning phase resulting in feature map formulation. These feature maps can be obtained by using input image scanning through a single weight unit by forming a local field with a combination of previous features and stores in the output. This process of feature generation is similar to data convolution with kernel. Later, this feature map generation can be considered as a plane which shares the weight of each unit. In the next phase of CNN, data subsampling is performed that follows local and convolutional feature maps for generating feature distortions; it reduces the spatial resolution of the data and increases its complexity. This spatial resolution reduction and data complexity increment are detected in each successive layer. CNN contains 6 layers where layer 1 performs convolution on the pre-processed data. This formulates a convolution mask where weights of each input data are shared on the same feature map.

For data scanning, the window selection is chosen as 20x20 pixels for each incoming data of mask size 5x5 and the feature map size is considered as 16x16. The next layer is denoted by S2 which is also known as averaging or sub-sampling layer where 4 planes are formulated, each plane is of size 16x16 pixels. Here each unit receives the feature set as input from the corresponding layer C1. Receptive fields do not overlap the weights in single unit. Hence, local averaging is performed and a subsampling of 2 to 1. After performing the feature extraction process, spatial and substantial feature relationships are obtained during the next phase of CNN which shows that layers S1 and C2 are connected to each other based on the different feature maps. Layer C2 contains SIFT and optical flow features. In this model, a total of 40 features are considered and operated in 3x3 size receptive field. The first 20 features are considered into a single receptive field and formulated into two groups of action where similar or non-similar actions are stored. Finally, the last layer of CNN model contains feature points which are connected to all previous layers in the network. In this approach, weight sharing helps to reduce the free parameters and improves the database training capacity in a supervised learning process using back-propagation approach adopted for convolutional neural network.

EXPERIMENTAL RESULTS AND DISCUSSION

In order to analyze the performance of the proposed algorithm, two open source action datasets: (a) Weizmann human action dataset (Blank, Gorelick, Shechtman, Irani, & Basri, 2005, October) and (b) KTH human action dataset (Schuldt, Laptev, & Caputo, 2004) were considered.

Dataset Description

Two public datasets, Weizmann and KTH datasets were used for experimental analysis of the work.

Weizmann Human Action Dataset

The dataset contained 90 video sequences with a resolution of 180x144 where nine users performed ten different actions such as: walking, running, skipping, one hand-waving, two-hand waving, jacking, jumping, sideways movement, etc. Each video sequence contained about 40–120 frames. Figure 4 shows examples of some frames of each action in the Weizmann dataset.

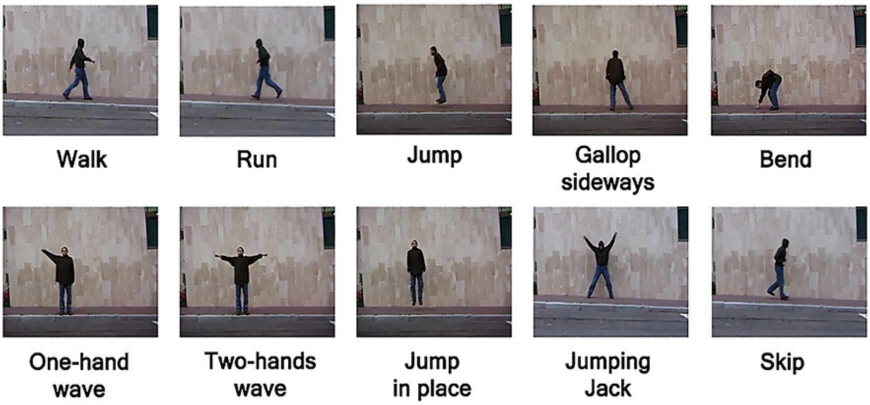


Figure 4. Weizmann dataset action sample frames.

KTH Activity Recognition Dataset

The dataset was created by KTH Royal Institute of Technology in 2004 for developing a new approach in computer vision application for human activity recognition. This database contained various video sequences of actions which were obtained for various scenarios. Similar to the Weizmann dataset, this dataset was also captured over homogenous backgrounds, using a static camera. This dataset consisted of six types of actions such as jogging, walking, running, boxing, hand waving and hand clapping which were performed by 25 people for several times in different scenarios. Sample frames corresponding to each action are depicted in Figure 5. The video sequences were divided with respect to the subjects into a training set (eight persons), a validation set (eight persons) and a test set (nine persons).

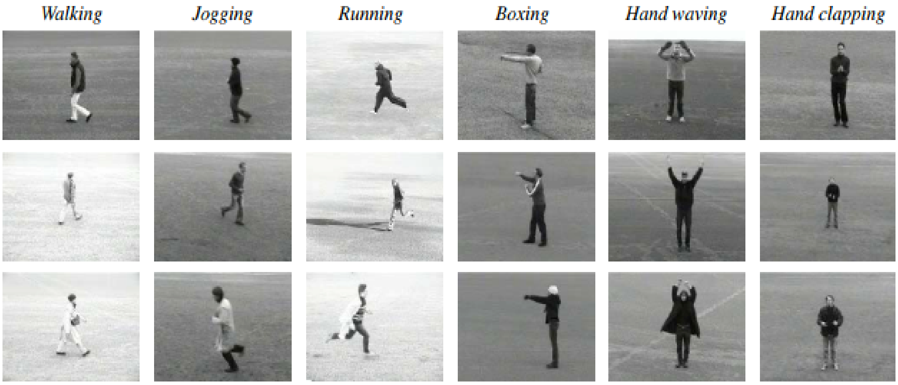
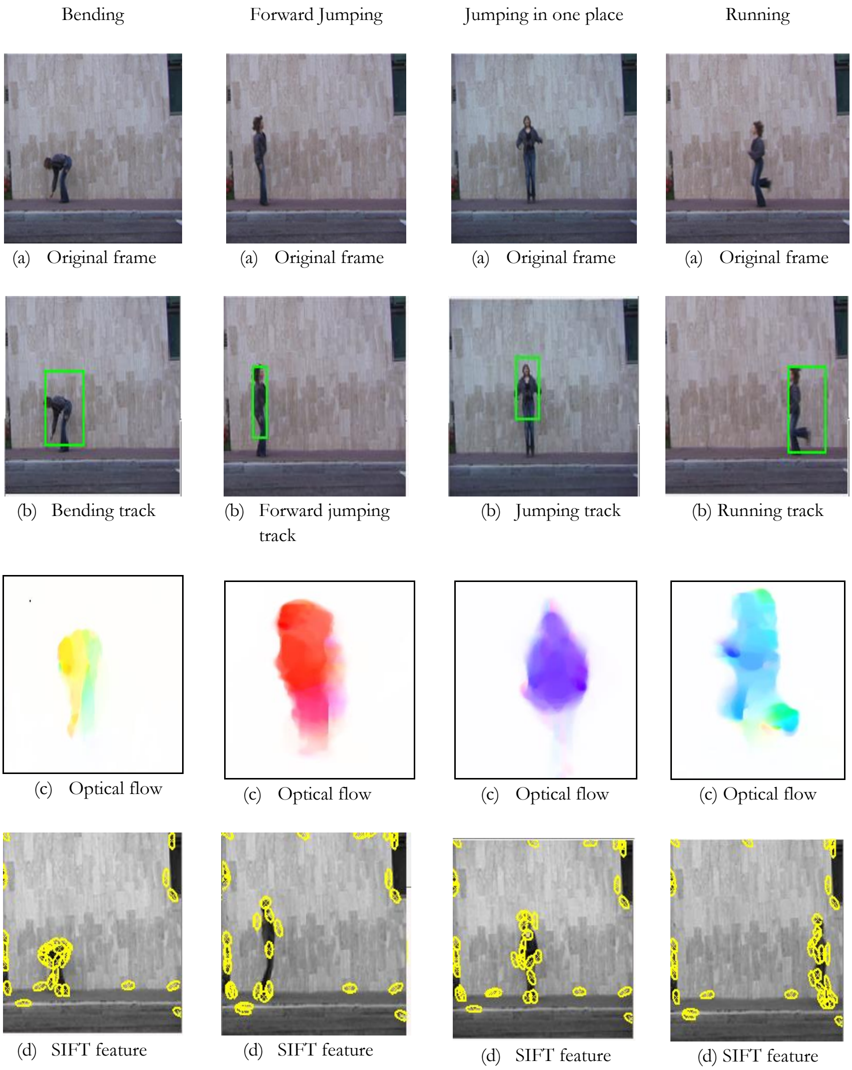


Figure 5. KTH sample frames on action recognition.

Experimental Study for Weizmann Dataset

Initially, the experiment was conducted on the Weizmann dataset. In the proposed approach, tracking, optical flow, SIFT feature extraction and classification were performed. This complete process for each activity is depicted in Figure 6 for the Weizmann dataset. The dataset of 226 videos (10 classifications) was divided into two parts: training (150 videos) and testing (76 videos).

Similarly, tracking, optical flow, SIFT feature extraction and classification were performed on the other action videos of Weizmann's dataset. A feature vector was formulated which was processed through CNN where multiple layers were connected to each other to formulate a trained network. Based on the performance of the Convolutional Neural Network approach, the obtained confusion matrix for Weizmann dataset is shown in Table 1 as follows.



Bending

Forward Jumping

Jumping in one place

Running

(a) Original frame

(a) Original frame

(a) Original frame

(a) Original frame

(b) Bending track

(b) Forward jumping track

(b) Jumping track

(b) Running track

(c) Optical flow

(c) Optical flow

(c) Optical flow

(c) Optical flow

(d) SIFT feature

(d) SIFT feature

(d) SIFT feature

(d) SIFT feature

Figure 6. Action and their corresponding stages using proposed approach.

Table 1

Confusion Matrix for Weizmann Dataset Action Classification

	Bend	Jump	Run	P Jump	Skip	Side	Jack	Walk	Wave 1	Wave 2
Bend	0.9669	-	-	0.0331	-	-	-	-	-	-
Jump	-	0.9602	-	-	0.0398	-	-	-	-	-
Run	-	0.0301	0.9699	-	-	-	-	-	-	-
P Jump	-	-	-	0.98988	-	-	0.01012	-	-	-
Skip	-	-	-	-	0.9877	-	-	0.0123	-	-
Side	-	-	-	-	0.0088	0.9912	-	-	-	-
Jack	0.01288	-	-	-	-	-	0.98712	-	-	-
Walk	-	-	-	-	-	-	-	0.9877	-	0.0123
Wave 1	-	-	0.0178	-	-	-	-	-	0.9822	-
Wave 2	-	-	-	-	-	-	0.0197	-	-	0.9803

From the obtained confusion matrix, the overall classification accuracy obtained was 98.03% which was comparatively better when compared with other techniques. Misclassification of few actions were due the similarity in features of the actions. The *side* movement action achieved the highest accuracy of classification while the *jump* activity had less accuracy. A comparative classification performance analysis is shown in Table 2.

Table 2

Classification Performance Comparison for Weizmann Dataset

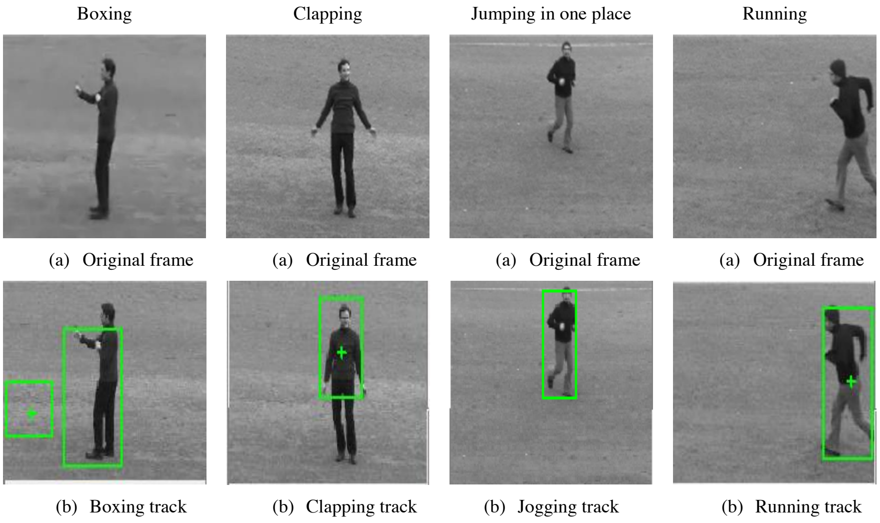
Reference Paper	Technique used	Classification Accuracy
Cai et al. (2017)	Discriminative two-phase dictionary learning framework for classifying human action by sparse shape representations	97.85%
Cai et al. (2017)	Discriminative two-phase dictionary learning framework for classifying human action by sparse shape representations	95.70%

(continued)

Reference Paper	Technique used	Classification Accuracy
Chaaraoui et al. (2013)	Pose representation by means of a distance signal feature with model learning approach	92.77%
Cheema et al. (2011)	Scale invariant contour-based pose feature extraction, weighted voting scheme for classification	91.6%
Wang et al. (2012)	Continuous motion segment descriptor and modified sparse model for classification	96.7%
Cheng et al. (2015)	Supervised temporal t-stochastic neighbor embedding (ST-tSNE) and incremental learning	94.44%
Proposed Method	**SIFT Feature Extraction, Optical Flow Computation, Convolutional Neural Networks Classifier**	**98.03%**

Experimental Study for KTH Dataset

The proposed approach tracking, optical flow, SIFT feature extraction and classification for KTH dataset is shown in Figure 7.



Boxing

Clapping

Jumping in one place

Running

(a) Original frame

(a) Original frame

(a) Original frame

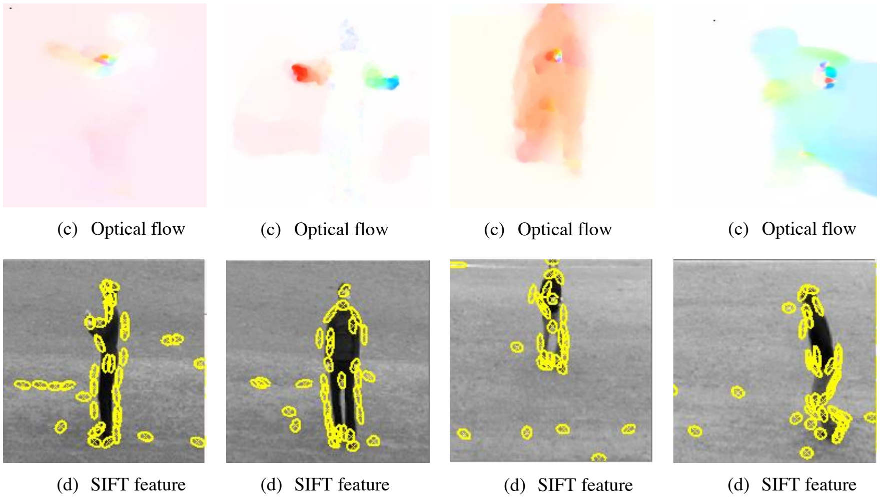
(a) Original frame

(b) Boxing track

(b) Clapping track

(b) Jogging track

(b) Running track



(c) Optical flow (c) Optical flow (c) Optical flow (c) Optical flow

(d) SIFT feature (d) SIFT feature (d) SIFT feature (d) SIFT feature

Figure 7. Action and their corresponding stages using proposed approach.

A similar process of tracking, optical flow, SIFT feature extraction and classification were applied on the remaining video clips. These feature sets were considered for training and testing using CNN. Based on CNN classification study, the obtained confusion matrix for the dataset is shown in Table 3.

Table 3

Confusion Matrix for KTH Database Action Classification

	Walking	**Jogging**	**Running**	**Boxing**	**Hand clapping**	**Hand waving**
Walking	0.9242	0.0758	-	-	-	-
Jogging	0.056	0.944	-	-	-	-
Running	-	-	0.987	-	0.013	-
Boxing	-	-	-	0.941	0.059	-
Hand clapping	0.02	-	-	-	0.98	-
Hand Waving	-	0.08	-	-	-	0.92

Based on this study, the proposed approach achieved a classification accuracy of 94.96% which was comparatively good with respect to other techniques. The comparative analysis of classification accuracy is given in Table 4. In the activity recognition process, walking, jogging and running were misclassified because of the moderate similarity among these activities. However, the proposed feature extraction process helped to segregate these features distinctly. According to the proposed approach, hand clapping activity achieved the highest accuracy because of less movements involved. Running and boxing also achieved better performance in terms of activity recognition.

Table 4

Classification Accuracy Comparison for KTH Dataset

Reference Paper	Technique Used	Classification Accuracy
Kamiński et al. (2017)	MPEG-7 Compact descriptors for visual search (CDVS) to describe a human pose and distance based ranking system for classification.	81.80%
Niebles et al. (2008)	Probabilistic Latent Semantic Analysis (pLSA) model and Latent Dirichlet Allocation (LDA).	83.33%
Klaser et al. (2008)	Histograms of oriented 3D spatio-temporal gradients, 3D orientation quantization	91.40%
Lu and Zhang (2014)	HOG3D, K-means clustering algorithm, Support Vector Machine (SVM)	91.5%
Liu et al. (2017)	3D SIFT, Principal Component Analysis (PCA), Support Vector Machine (SVM) and AdaBoost-SVM Classifiers	94.92%
Proposed Method	**SIFT feature extraction, Optical flowcomputation,Convolutional Neural Networks Classifier**	**94.96%**

Misclassification of actions occurred due to background and feature similarity. The quantitative comparison of the proposed approach was made with other reviewed techiques for the two publicly available datasets KTH and Weizmann. The computational cost of the 2D approaches was comparatively less than the 3D and currently, is still a better choice for activity recognition.

CONCLUSION

This paper focuses mainly on human action recognition and classification using machine learning techniques. A novel approach in feature extraction where previously, input video stream is converted into multiple frames and processed through feature extraction techniques. Feature extraction in this improvized cascaded approach is presented where optical flow computation and SIFT features are extracted and combined to avoid misclassification due to pose variations. Moreover, a distance based similarity scheme is also incorporated to avoid videos or frames which are dissimilar to the input; therefore reducing time taken for computation. Finally, a CNN based classifier model is utilized to classify the activities. The proposed approach is implemented on two publicly available open-source datasets: Weizmann and KTH datasets. Moreover, a comparative analysis is also presented to highlight that the proposed approach is capable of delivering a promising classification performance for action recognition as compared to state-of-the-art techniques. In future, this work can be extended to own datasets and comparative analysis conducted to evaluate the performance of the method used.

ACKNOWLEGEMENT

The authors would like to express their gratitude to the Research Center, Department of Electronics and Communication Engineering, Vidyavardhaka College of Engineering, Mysuru, India which is recognized by the Visvesvaraya Technological University, Belagavi, India for providing the support and facilities to carry out the research work. This research received no specific grant from any funding agency in the public, commercial, or not-for profit sectors.

REFERENCES

Aggarwal, J. K., & Ryoo, M. S. (2011). Human activity analysis. *ACM computing surveys*, *43*(3), 1–43. https://doi.org/10.1145/1922649.1922653

Burghouts, G. J., & Schutte, K. (2013). Spatio-temporal layout of human actions for improved bag-of-words action detection. *Pattern Recognition Letters*, *34*(15), 1861–1869. https://doi.org/10.1016/j.patrec.2013.01.024

Cai, J., Tang, X., Zhang, L., & Feng, G. (2017). Learning zeroth class dictionary for human action recognition. *Communications in Computer and Information Science*, vol. 773, 651–666. https://doi.org/10.1007/978-981-10-7305-2_55

Chaaraoui, A. A., Climent-Pérez, P., & Flórez-Revuelta, F. (2013). Silhouette-based human action recognition using sequences of key poses. *Pattern*

Recognition Letters, *34*(15), 1799–1807. https://doi.org/10.1016/j.patrec.2013.01.021

Cheema, S., Eweiwi, A., Thurau, C., & Bauckhage, C. (2011). *Action recognition by learning discriminative key poses*. Paper presented at the IEEE International Conference on Computer Vision Workshops (ICCV Workshops). Retrieved from https://doi.org/10.1109/iccvw.2011.6130402

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*. Paper presented at the IEEE Transactions on Image Processing, *24*(10), 3203–3217. https://doi.org/10.1109/tip.2015.2441634

Dollar, P., Rabaud, V., Cottrell, G., & Belongie, S. (2019). *Behavior recognition via sparse spatio-temporal features*. Paper presented at the IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. https://doi.org/10.1109/vspets.2005.1570899

Farhadi, A., & Tabrizi, M. K. (2008). Learning to recognize activities from the wrong view point. *Lecture Notes in Computer Science*, vol. 5302, 154–166. https://doi.org/10.1007/978-3-540-88682-2_13

Gorelick, L., Blank, M., Shechtman, E., Irani, M., & Basri, R. (2005). *Actions as space-time shapes*. Paper presented at the IEEE Transactions on Pattern Analysis and Machine Intelligence, *29*(12), 2247–2253. https://doi.org/10.1109/tpami.2007.70711

Ji, S., Xu, W., Yang, M., & Yu, K. (2013). *3D convolutional neural networks for human action recognition*. Paper presented at the IEEE Transactions on Pattern Analysis and Machine Intelligence, *35*(1), 221–231. https://doi.org/10.1109/tpami.2012.59

Ji, X.-F., Wu, Q.-Q., Ju, Z.-J., & Wang, Y.-Y. (2014). Study of human action recognition based on improved spatio-temporal features. *International Journal of Automation and Computing*, *11*(5), 500–509. https://doi.org/10.1007/s11633-014-0831-4

Jiang, H., Drew, M. S., & Ze-Nian Li. (2010). *Action detection in cluttered video with successive convex matching*. Paper presented at the IEEE Transactions on Circuits and Systems for Video Technology, *20*(1), 50–64. https://doi.org/10.1109/tcsvt.2009.2026947

Kaminski, L., Mackowiak, S., & Domanski, M. (2017). *Human activity recognition using standard descriptors of MPEG CDVS.* Paper presented at the IEEE International Conference on Multimedia & Expo Workshops (ICMEW). https://doi.org/10.1109/icmew.2017.8026248

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). *Large-scale video classification with convolutional neural* *networks*. Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/cvpr.2014.223

Klaser, A., Marszałek, M., & Schmid, C. (2008). A spatio-temporal descriptor based on 3D-gradients. *Proceedings of the British Machine Vision Conference*, 99.1-99.10. https://doi.org/10.5244/C.22.99

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90. https://doi.org/10.1145/3065386

Laptev, I., Marszalek, M., Schmid, C., & Rozenfeld, B. (2008). Learning realistic human actions from movies. Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/cvpr.2008.4587756

Lee, S., Le, H. X., Ngo, H. Q., Kim, H. I., Han, M., & Lee, Y. K. (2011). Semi-Markov conditional random fields for accelerometer-based activity recognition. *Applied Intelligence, 35*(2), 226-241.

Li, X., Zhang, Y., & Liao, D. (2017). Mining Key Skeleton Poses with Latent SVM for Action Recognition. *Applied Computational Intelligence and Soft Computing*, 1–11. https://doi.org/10.1155/2017/5861435

Liu, M., Chen, C., & Liu, H. (2017). Time-ordered spatial-temporal interest points for human action classification. Paper presented at the IEEE International Conference on Multimedia and Expo (ICME), 655–660. https://doi.org/10.1109/icme.2017.8019477

Lu, M., & Zhang, L. (2014). Action recognition by fusing spatial-temporal appearance and the local distribution of interest points. *Proceedings of the 2014 International Conference on Future Computer and Communication Engineering*. https://doi.org/10.2991/icfcce-14.2014.19

Meng, B., Liu, X., & Wang, X. (2018). Human action recognition based on quaternion spatial-temporal convolutional neural network and LSTM in RGB videos. *Multimedia Tools and Applications, 77*(20), 26901–26918. https://doi.org/10.1007/s11042-018-5893-9

Moghaddam, Z., & Piccardi, M. (2014). *Training initialization of hidden Markov models in human action recognition*. Paper presented at the IEEE Transactions on Automation Science and Engineering, *11*(2), 394–408. https://doi.org/10.1109/tase.2013.2262940

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. https://doi.org/10.1007/s11263-007-0122-4

Ramasso, E., Panagiotakis, C., Pellerin, D., & Rombaut, M. (2007). Human action recognition in videos based on the transferable belief model. *Pattern Analysis and Applications, 11*(1), 1–19. https://doi.org/10.1007/s10044-007-0073-y

Schuldt, C., Laptev, I., & Caputo, B. (2004). Recognizing human actions: A local SVM approach. *Proceedings of the 17th International Conference on Pattern Recognition*, Vol.3, 32–36. https://doi.org/10.1109/icpr.2004.1334462

Shao, L., Zhen, X., Tao, D., & Li, X. (2014). *Spatio-temporal laplacian pyramid coding for action recognition*. Paper presented at the IEEE Transactions on Cybernetics, *44*(6), 817–827. https://doi.org/10.1109/tcyb.2013.2273174

Taylor, G. W., Fergus, R., LeCun, Y., & Bregler, C. (2010). Convolutional Learning of Spatio-temporal Features. *Computer Vision – ECCV*, 140–153. https://doi.org/10.1007/978-3-642-15567-3_11

Varol, G., Laptev, I., & Schmid, C. (2018). *Long-term temporal convolutions for action recognition*. Paper presented at the IEEE Transactions on Pattern Analysis and Machine Intelligence, *40*(6), 1510–1517. https://doi.org/10.1109/tpami.2017.2712608

Wang, Haoran, Yuan, C., Hu, W., & Sun, C. (2012). Supervised class-specific dictionary learning for sparse modeling in action recognition. *Pattern Recognition, 45*(11), 3902–3911. https://doi.org/10.1016/j.patcog.2012.04.024

Wang, Heng, Kläser, A., Schmid, C., & Liu, C.-L. (2013). Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision, 103*(1), 60–79. https://doi.org/10.1007/s11263-012-0594-8

Wang, Lei, Xu, Y., Cheng, J., Xia, H., Yin, J., & Wu, J. (2018). *Human action recognition by learning spatio-temporal features with deep neural networks*. IEEE Access, *6*, 17913–17922. https://doi.org/10.1109/access.2018.2817253

Wang, Lei, Xu, Y., Cheng, J., Xia, H., Yin, J., & Wu, J. (2018). Human action recognition by learning spatio-temporal features with deep neural networks. IEEE Access, *6*, 17913–17922. https://doi.org/10.1109/access.2018.2817253

Wang, Limin, Qiao, Y., & Tang, X. (2015). *Action recognition with trajectory-pooled deep-convolutional descriptors*. Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4305–4314. https://doi.org/10.1109/cvpr.2015.7299059

Wang, Limin, Qiao, Y., & Tang, X. (2015). *Action recognition with trajectory-pooled deep-convolutional descriptors*. Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4305–4314. https://doi.org/10.1109/cvpr.2015.7299059

Wong, S.-F., & Cipolla, R. (2007). *Extracting spatiotemporal interest points using global information*. Paper presented at the IEEE 11th International Conference on Computer Vision, 1–8. https://doi.org/10.1109/iccv.2007.4408923

Wong, S.-F., & Cipolla, R. (2007). *Extracting spatiotemporal interest points using global information*. Paper presented at the IEEE 11th International Conference on Computer Vision, 1–8. https://doi.org/10.1109/iccv.2007.4408923

Wu, S., Oreifej, O., & Shah, M. (2011). *Action recognition in videos acquired by a moving camera using motion decomposition of Lagrangian particle trajectories*. Paper presented at the International Conference on Computer Vision, 1419-1426. https://doi.org/10.1109/iccv.2011.6126397

Wu, S., Oreifej, O., & Shah, M. (2011). Action recognition in videos acquired by a moving camera using motion decomposition of Lagrangian particle trajectories. *International Conference on Computer Vision*, 1419-1426. https://doi.org/10.1109/iccv.2011.6126397

Yao, L., Liu, Y., & Huang, S. (2016). Spatio-temporal information for human action recognition. *EURASIP Journal on Image and Video Processing, 2016*(1). https://doi.org/10.1186/s13640-016-0145-2

Yao, L., Liu, Y., & Huang, S. (2016). Spatio-temporal information for human action recognition. *EURASIP Journal on Image and Video Processing, 2016*(1). https://doi.org/10.1186/s13640-016-0145-2

Zhang, B., Yang, Y., Chen, C., Yang, L., Han, J., & Shao, L. (2017). *Action recognition using 3D histograms of texture and a multi-class boosting classifier*. Paper presented at the IEEE Transactions on Image Processing, 26(10), 4648–4660. https://doi.org/10.1109/tip.2017.2718189

Zhang, B., Yang, Y., Chen, C., Yang, L., Han, J., & Shao, L. (2017). Action recognition using 3D histograms of texture and a multi-class boosting classifier. Paper presented at the IEEE Transactions on Image Processing, *26*(10), 4648–4660. https://doi.org/10.1109/tip.2017.2718189

Zhen, X., Zheng, F., Shao, L., Cao, X., & Xu, D. (2017). Supervised local descriptor learning for human action recognition. Paper presented at the IEEE Transactions on Multimedia, *19*(9), 2056–2065. https://doi.org/10.1109/tmm.2017.2700204

Zhen, X., Zheng, F., Shao, L., Cao, X., & Xu, D. (2017). Supervised local descriptor learning for human action recognition. Paper presented at the IEEE Transactions on Multimedia, *19*(9), 2056–2065. https://doi.org/10.1109/tmm.2017.2700204

Zhu, S., & Xia, L. (2015). Human action recognition based on fusion features extraction of adaptive background subtraction and optical flow model. *Mathematical Problems in Engineering*, 1–11. https://doi.org/10.1155/2015/387464