Deep Learning based Intelligent Surveillance System

Muhammad Ishtiaq[1], Rashid Amin[3]
Department of Computer Science
University of Engineering and Technology
Taxila, Pakistan

Sultan H. Almotiri[2], Mohammed A. Al Ghamdi[4]
Computer Science Department
Umm Al-Qura University
Makkah City, Saudi Arabia

Hamza Aldabbas[5]
Prince Abdullah bin Ghazi Faculty of Information and Communication Technology
Al-Balqa Applied University
Al-Salt- Jordan

***Abstract*—In the field of developing innovation, pictures are assuming as an important entity. Almost in all fields, picture base data is considered very beneficial, like in the field of security, facial acknowledgment, or therapeutic imaging, pictures make the existence simple for people. In this paper, an approach for both human detection and classification of single human activity recognition is proposed. We implement the pre-processing technique which is the fusion of the different methods. In the first step, we select the channel, apply the top hat filter, adjust the intensity values, and contrast stretching by threshold values applied to enhance the quality of the image. After pre-processing a weight-based segmentation approach is implemented to detect and compute the frame difference using cumulative mean. A hybrid feature extraction technique is used for the recognition of human action. The extracted features are fused based on serial-based fusion and later on fused features are utilized for classification. To validate the proposed algorithm 4 datasets as HOLLYWOOD, UCF101, HMDB51, and WEIZMANN are used for action recognition. The proposed technique performs better than the existing one.**

***Keywords*—HMG; ALMD; PBoW; DPNs LOP; BoF; CT; LDA; EBT**

I. Introduction

In the last few years, there is a significant increase in image, video and multimedia content that is increasing day by day. Surveillance means supervision or keeping an eye on some gatherings or particular events. So, it is considered that video surveillance is the best option for monitoring and observing. Manual surveillance is too hectic and time-consuming. By using this surveillance system, we can easily detect what is happening at a particular place, and remotely we can monitor many places in the meantime [1]. Also, there has been remarkable progress in the video analyzing techniques and algorithms. Many researchers have attempted to develop good intelligent surveillance systems that can recognize any human activity through different approaches. Accuracy and Efficiency are the main concerns as 100% accuracy is not achieved [2]. There are a lot of Video Surveillance Systems and the focus of each of them is to take its place in the market. Video surveillance involves analysis that contains a list of steps like video preprocessing, object detection, action detection, recognition, and categorization of actions [3]. The videos and images collected from cameras require large memory for storing and processing them.

Deep Learning approaches are more suitable for handling and analyzing such large data sets [4]. These approaches can perform an analysis of the image and video data sets that are available publically. These trained models of deep learning can achieve an accuracy of more than 95 % in some cases.

In this paper, we use the following human activities dataset for the training of the system i.e., HMDB51, UCF50, Weizmann, Hollywood Movies. HMDB51 is the actions database that includes the actions of a cartwheel, catch, draw the sword, jump, kick, and laugh, pick, sit up, smile, turn, walk and wave, etc. UCF50 is also human action database that includes the actions of Basketball, clean and jerk, diving, horse riding, kayaking, mixing, nun chucks, playing the violin, skateboarding, tennis swing, and yoyo, etc. Hollywood movies dataset consist of 8 actions which include getting out of the car, answer phone calls, handshake, hug, and kiss, sit down, sit up and stand up. In this paper, we used the SVM classifier for classification. Support Vector Machine (SVM) is the model of supervised learning. SVMs use very cleverly a large number of features for learning without using additional computational power. These are capable to represent nonlinear functions and able to use efficient algorithms for learning. The system deals with static images and live camera video of human activities. There are 6 main phases of our proposed system e.g., Image Acquisition, Image Pre-processing, Feature Extraction, Training, Testing, Classification. Because of the new features and image processing used, we claim that this system can be used for large databases with an accuracy of more than 90%. Accuracy depends on the number and type of features used. Here we use local, global, and geometric features for classification and selection.

The paper is organized as the Section II presents the related work. Section III proposed system model is discussed and data sets used in this research are elaborated in Section IV. Feature extraction and classification are discussed in Sections V and VI, respectively. Performance evaluation is presented in Section VIII and Section IX concludes the paper.

II. Related work

A lot of work has already been done on intelligent surveillance systems; many researchers have attempted to develop good intelligent surveillance systems that can recognize any human activity through different approaches. The following are the different techniques used.

A. Human Action Recognition (HAR) Techniques

Many researchers worked in the domain of HAR. In the area of computer vision, researchers are using a distributed

and dynamic environment for the performance evaluation of multimedia data. Ionut et al. [5] proposed a scheme for encoding of features and their extraction to get real-time processing of frame rate for action recognition systems. An approach is proposed to get the motion information within the captured video. The descriptor which is proposed, Histogram of Motion Gradient (HMG) is Spatio-temporal derivation based. For encoding step Vector of Locally Aggregated Descriptors (VLAD) method is applied. Challenging data sets namely UCF101, HMDB51, and UCF50 are used for validation purposes. The proposed method has improved accuracy and computational cost.

B. Shape Features based Methods

Azher et al. [6] introduced a novel technique to recognize human actions. A novel feature descriptor is introduced namely, Adaptive Local Motion Descriptor (ALMD) by considering motion and appearance. This technique is an extension of the Local Ternary Pattern (LTP), which is used for static text analysis. Spark MLlib (Machine Learning Library) Random Forest method is employed to recognize human actions. KTH dataset of 600 videos including six human action classes is used for testing purposes. UCF sports action and UCF-50 data sets are also used for result analysis.

Chen et al. [7] presented a Spatio-temporal descriptor to recognize human actions from depths of video sequences. In the research, improved Depth Motion Maps (DMM) [8-10] are compared to previously introduced DMMs. Fisher Kernel [11] method is applied for generating bunched features representation, afterward, they are associated with Kernel-Based Extreme Learning Machine (ELM) [12] classifier. The developed solution is implemented on MSR Action 3D, Depth-included Human Action (DHA), MSR Gesture 3D, MSR action data sets.

Luvizon et al. [13] propose a new technique for HAR, from the sequences of the skeleton. The research proposes the extraction of Spatio-temporal sets of local features. Aggregation of extracted features is done through VLAD algorithm. K-NN classifier is used to bring accuracy in results. MSR-Action 3D, UT-Kinect Action 3D, and the Florence 3D Actions data sets are used for the evaluation of the proposed methodology. The aim of the research is an improvement in accuracy and computational time.

Liu et al. [14] presented a space-time approach for the analysis of hidden sources of action and activity-related information. To handle noise and occlusion in 3D skeleton data, a mechanism within LSTM is introduced. 3D human action datasets are used namely; SBU interaction, Berkely MHAD, and UT-Kinect datasets are used to test the proposed method. Improvement in accuracy and decrement in noise are the main achievements of the research.

Veenendaal et al. [15] examine the use of Dynamic Probabilistic Networks (DPNs) for human action recognition. The actions of lifting objects, walking, sitting, and neutral standing are used to test classification. The recognition accuracy performance between indoor (controlled lighting conditions) is compared with the outdoor lighting conditions. The results inferred that accuracy in outdoor scenes was lower than the controlled environment.

Zhong et al. [16] introduced a novel technique to recognize cross-view actions performed by using virtual paths. Virtual View Kernel (VVK) is used for finding similarities between multidimensional features. For simulations and analysis purposes IXMAS and MuHAVi datasets are used. VVK makes use of virtual views created by virtual paths and the results show that this proposed cross-view action recognition methodology brings enhanced system performance.

C. Point Features based Techniques

Gao et al. [17] introduced a technique for the recognition of multiple fused human actions. At the initial stage, STIP's features are extracted and then the Multi-View Bag of Words (MVBoW) model is deployed. Alongside the extraction of STIP's features and implementation of MVBoW's model, the graph model is also applied which removes the intersecting/overlapping points of interest from the data. Experimentations are done on a large scale on two famous datasets namely IXMAS and CVS-MV-RGB datasets. Simulation results proved the competent results of the proposed methodology.

Yang et al. [18] presented a technique for HAR in real-world multimedia. For the extraction of foreground motion, background motion is compensated by using a motion model. The needed foreground patch is over segmented in Spatio-temporal patches using trajectory spectral clustering. Three features are extracted namely HOG, HOF and Motion Boundary Histogram, and K-means are applied to construct a visual dictionary of each extracted feature. Competent results are obtained after simulations done on YouTube, UCF Sports, and Hollywood dataset.

Zhang et al. [19] presented a coding scheme/model which can learn more accurate and discriminative representations then other existing models. HOG, HOF, and HOG 3D descriptor features are extracted. Manifold-constrained Sparse Representation based Recognition (MSRR) [20] method is applied to the extracted features. The proposed methodology brings robust results against occlusion and multi-views. For classification of features, many classifiers like SVM, K-NN, HMM, AdaBoost: AdaBoost, and Sparse Representation-based classification (SRC) are used. Weizmann, KTH, UCF, and Facial expression datasets are used for testing purposes of the proposed methodology.

Guo et al. [21] proposed a novel HAR technique using normalized multi-task learning. To overcome the problem of the human body features to recognize human actions a 3 stage Part Bag of Words (PBoW) [22] approach is introduced to describe extracted features. PBoW approach divides the human body into 7 parts and HOG / HOF features of each part are extracted. Then K-means are applied to each feature and seven PBoW's are obtained for each sample. For experimentation TJU multi-action dataset is used with depth, skeleton and RGB data. Results show that good accuracy is gained by the proposed methodology.

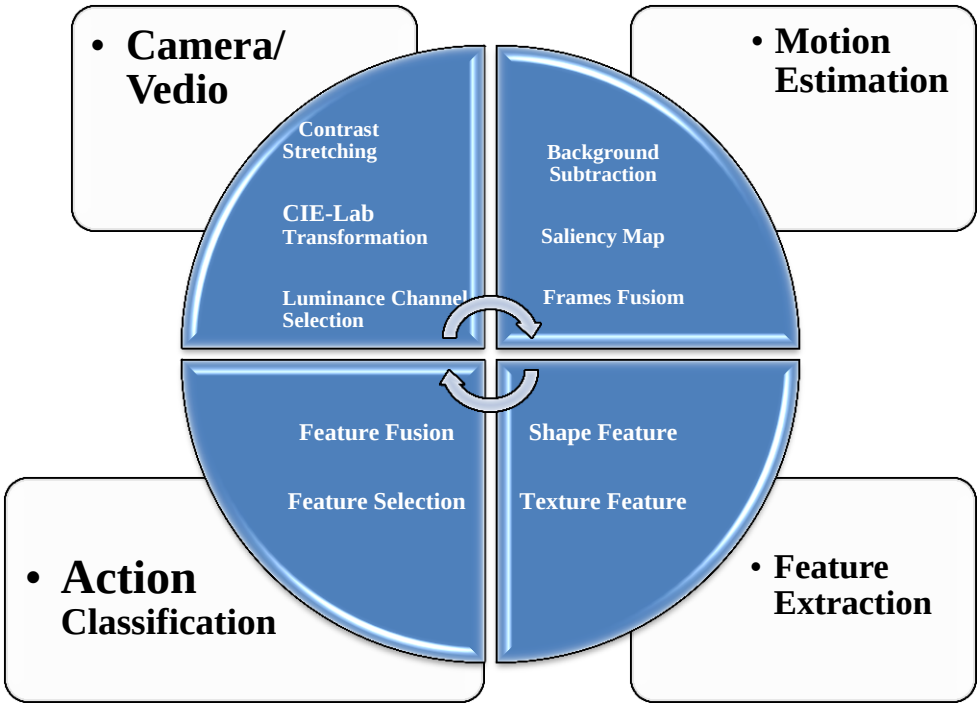

Fig 1. Block Diagram of the Proposed System.

Wang et al. [23] propose a new technique to overcome the problem of intra-class variance present in different human actions due to occlusion. The proposed method is applied to well-known datasets namely MSR daily activity and CMU Motion Capture dataset and evaluated results show that the proposed method achieved better results. Features that are robust against noise and invariant to translate are proposed in this research. Based on 3D body joints positions and depth features Local Occupancy Pattern (LOP) [23] is proposed. Fourier Temporal Pyramid [24] technique is applied for the illustration of temporal sequences.

D. Classifier based Techniques

Huang et al. [25] proposed a neural network approach for HAR. The proposed method is based on Self Organizing Map (SOM) scheme which is used to combine feature vectors and reduces dimensionality in data. After the trajectories of SOM, the stage of action recognition starts. Actions are mapped on the trajectories obtained, as a result, a similar trajectory is produced which causes trajectory matching problems. This problem is overcome in the proposed methodology by using the Longest Common Sequence (LCS) method. The Weizmann dataset is used for experimentation and promising results are obtained from the proposed method.

Shikhar et al. [26] proposed a model for action recognition in multimedia. Long Short-Term Memory (LSTM) unit along with a multi-layered Recurrent Neural Network technique, is used to implement the proposed method. The proposed model is capable of classifying the content of multimedia and learns the more important parts of the available frame. The proposed model is evaluated on YouTube action, HMDB-51, and Hollywood-2 datasets. This model brings good results because it focuses on the frames on the content very deeply.

Hashemi et al. [27] used an entropy-based method of silhouette extraction for the representation of view-reliant HAR and trajectory of feature vector points of the human body for the representation of view-unaided HAR. For classification, Bag of Features (BoF) technique is used and then K-means is applied over each feature. The clustering approach is also used in this research which is applied to scale down the search space of feature vectors by cutting down the count of labels of each action class. The proposed method is tested on WVU and INRIA XMAS datasets. The experimental results proved the respective methodology along with the low computational cost.

In our proposed method we used a neural network model named densenet and also alexnet separately but we didn't use their classifier we used the SVM classifier instead of the model's classifier also we fused features to get more accuracy. Previously this approach was not adopted.

III. Proposed System

In our proposed approach, first, we implement the pre-processing technique, which is the fusion of different techniques. In the first step, we select the channel, apply the top hat filter, adjusting the intensity values and contrast stretching by threshold values applied to enhance the quality of the image. After the pre-processing a weight-based segmentation approach is implemented for detection to calculate frame difference using cumulative mean and to update the background by using weights and also to identify the foreground regions and further a hybrid feature extraction technique is used for recognition of human action. The extracted features are fused based on serial-based fusion and later on the fused feature is utilized for classification. To validate the proposed algorithm 4 datasets as HOLLYWOOD,

UCF101, HMDB51 and WEIZMANN dataset are used for action recognition. Overall system operation is shown in Fig. 1.

- Image Acquisition: The human action images are scanned through the scanner to bring the image in a digital format. The scanned images are cropped so that it only contains the required object area.
- Image Binarization: Image binarization is a fundamental research theme in image processing and an important pre-processing method in image recognition and edge/boundary detection. It is challenging to select the corresponding threshold for each image in different application domains.
- Image Enhancement: Image enhancement is the process of adjusting digital images so that the results are more suitable for display or further image analysis. For example, you can remove noise, sharpen, or brighten an image, making it easier to identify key features.
- Feature Extraction: The system extracts the key features from the images and forms a feature vector.
- Training: The system is trained using the feature vector.
- Classification: The result of classification is the human activity label and part of the image highlighted.
- Operating Environment: The system is developed using MATLAB version 2019a. The system works only on the Microsoft Windows platform. There is no such extra software is required to run and use the system.
- General Constraints: One of the major limitations of our system is that we need to train our system both on normal and suspicious activities. Then we train our system by inputting the normal and suspicious activities. After training, we can test any new activity. Second major problem is that if the images are too noisy, then we may lose some of the features and our system may classify the activity incorrectly.
- User: To assist the user in using the system, we prepare a user manual that is delivered along with the software.

IV. Datasets

Different data sets are discussed in this section along with brief details. The data sets are given along with their number of action classes, total video slips, number of actors performed in the respective data set, and resolution of frames.

A. Weizmann Action Dataset

Weizmann Institute of Science [28] provided Weizmann Event-Based (2001) and Actions as Space-Time Shapes (2005) Analysis data sets. Weizmann Event-Based data set comprises 4 action classes namely running, waving, jogging in place, and walking, as shown in Fig. 2. Actions in the data set are performed by various persons wearing varying clothes. The data set is comprised of 6000 frames long video sequences, which display different actions. Weizmann Space-Time Actions data set is comprised of 10 action classes namely walk, bend, skip, jump, jumping jack, gallop, one hand waving, two hands waving, run and jump on the place [29], as shown in Fig. 2. Just one person acted in each frame. Frames of 90 videos with the resolution of 180*144 and static cameras are taken for evaluation. Homogeneous backgrounds with robust actions like the addition of dog or bag etc. are part of the data set.

Fig 2. Frames from Weizmann dataset [31].

V. Feature Extraction

The feature is meant to be information required for solving computational problems in different fields of science and technology especially computer vision. An image, the feature can be an edge, shape, point, or object in a scene. Sometimes only one type of feature is not enough so two or more features are extracted from the given data set which results in two or more feature descriptors of each point in an image. Information extracted from these features descriptors is organized in the form of a single feature matrix/vector and concatenation of these vectors results in a space of features. In image processing, the process of feature extraction starts from collecting a set of derived features that are found informative, non-redundant, and are in the form that is interpreted. This process is done to reduce dimensionality in feature space. To reduce such huge dimensionality in feature space feature extraction techniques are discussed in this section such as Histogram of Oriented Gradient (HOG), color, Gabor filtration, Scale Invariant Feature Transform (SIFT), Speed Up Robust Features (SURF), and Local Binary Pattern (LBP).

A. Histogram of Oriented Gradient

An image or part of an image is represented by a feature descriptor [30] and helps in the extraction of useful information from that image and discards the irrelevant information of an image. It converts an image into 3 portions namely width*height*3 channels. The size of the input sample is 64*128*3 in HOG descriptor and it gives output feature vector of length 3780 dimensions. In general, these descriptors are not useful for viewing an image but very useful in the process of actions or activities recognition. When the extracted feature vector is fed into any classifier like Cubic – Support Vector Machine (C-SVM) or Complex tree (CT), they give very good recognition rates. In HOG feature descriptor, distributions of orientation (direction) gradient features are used to obtain feature vector. X gradient and Y gradient of the sample are measured because higher magnitude at edges of the

histogram is observed. HOG features are calculated by following 6 steps, namely, pre-processing, calculation of image gradient, histogram calculation, 16*16 block normalization, HOG feature vector calculation and visualization of HOG.

B. Pre-Processing

HOG features are calculated on a part of an image of size 64*128. A constraint for HOG is a fixed aspect ratio of images under analysis. In a pre-processing step, firstly, an image of any size is re-sized to 64*128 and makes it ready for HOG feature descriptors calculations.

C. Calculation of Image Gradient

In the second step, vertical and horizontal gradients of the sample are needed to be measured, it is done for the histogram of the gradient. It is done by applying the filter as shown in Fig. 3. In this step irrelevant information is almost separated from useful information and discarded, image boundaries are also specified in this step.

D. Histogram Calculation in 8*8 Cells

In this step, the given sample is converted into 8*8 blocks and again a histogram is measured for every block. These 8*8 blocks of the sample contain 8*8*3 = 192-pixel values. Two values magnitude and direction are obtained for 8*8 blocked patch of samples gradient which is added to 8*8*2 = 128 numbers. This 128 numeric value is divided into 9 bins of a histogram. The bins represent angles ranging from 0, 20, 40… 160.

E. 16*16 Block Normalization

Generally, histograms of images are sensitive to light intensity. To make histogram less affected by light, we need to normalize the image gradient obtained so far. If the image is colored, the process of normalization includes; finding the magnitude of RGB values and dividing each value by the magnitude.

F. HOG Feature Vector Calculation

For the calculation of the final feature, vector obtains from HOG feature descriptor, we need to concatenate all the feature vectors obtained so far to make a single large vector. In a 16*16 block of the image there exist 7 horizontal and 15 vertical positions i.e., 7*15 = 105 positions. 36*1 vectors are used to represent each block of 16*16 matrix-es. After concatenation 36*105 = 3780 dimensional vector is obtained.

G. Visualization of HOG

The HOG features can be visualized by plotting a normalized histogram of each 8*8 blocks. HOG features see a slightly different visual world than what humans see, and by visualizing this space, we can gain a more intuitive understanding of our object detectors.

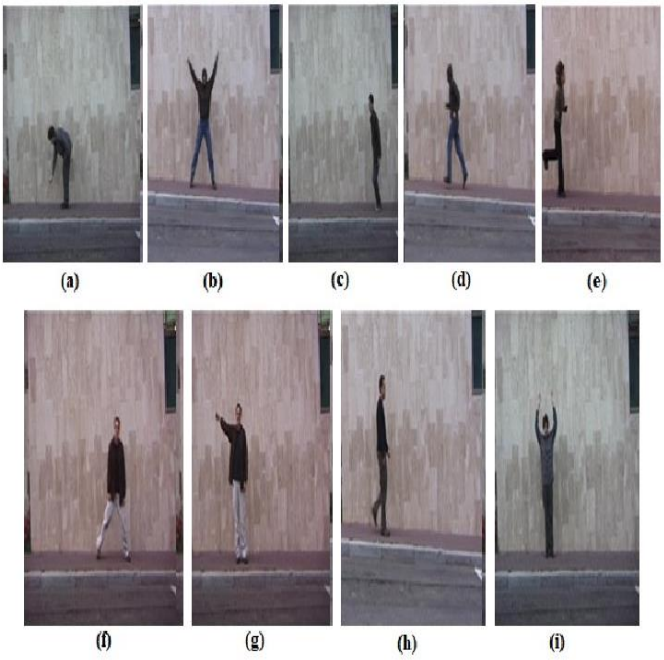

Fig 3. Frames from Weizmann Space Time Data set [29].

H. Color Feature

The color feature is useful in the classification and detection of an object. Colored images are mostly a combination of 3 color planes red, green, and blue (RGB planes). Each color plane is considered as a separate grayscale image or sample. Many factors contribute to the color feature like optical filtration, lightning, and printing effect on the photograph. We can obtain the relation between colors by colors transform like HSL, SCT, Luv, and YCbCr. There are three basic characteristics of color including hugeness, brightness (luminance of an object) and saturation. For the representation of color compactness, a color histogram is used. Colors are divided into K number of groups or bins according to the color intensity. Image is then transformed into K-dimensional points and distance matrix-like Euclidean distance measure is used to find similarity between K numbers of points.

I. Gabor Filtration

In image processing texture means regular repetition of some pattern on a surface. It is commonly used to separate non-textured (non-repetitive part) from textured (repetitive part). It is used for the classification of texture in a sample and to extract boundaries between large textured regions. Gabor Filter is used for texture analysis in image processing. Its purpose is to classify specific frequency components in the given region of an image and also classification in a specific orientation. Texture classification flowchart of the proposed system is shown in Fig. 4.

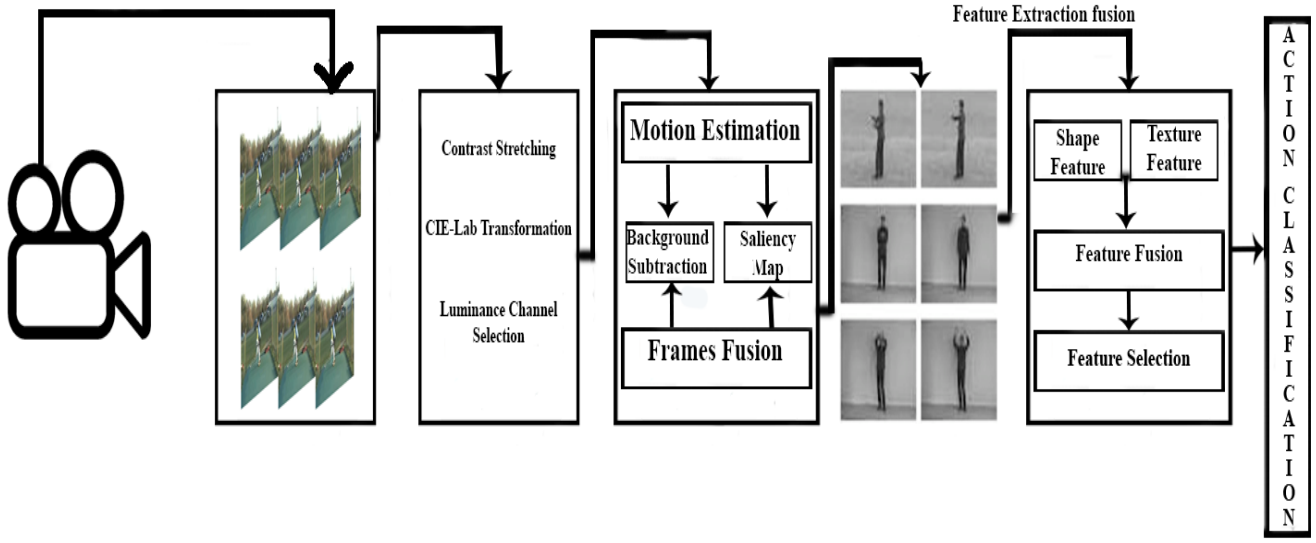

Fig 4. Flowchart of Texture Classification of the Proposed System.

VI. Classification

Classification in the image is assigning pixels to categories and classes of interest. In the process of classification, symbols are mapped with numbers. Before assigning data or pixels to certain classes and categories the relation between both data and class must be known. To achieve this purpose, we need to train the machine with data. Training is the basic theme of the classification process. First of all, classification techniques were proposed for pattern recognition. In the latest researches, computer-aided classification is taking place. It has several advantages like the processing of a large number of images; it gives better consistency in results, it can be applicable for large application datasets and gives an insight into the relationship between data and categories to the researchers. Computer-aided classification is divided into three types of classification namely supervised, unsupervised and hybrid classification. In supervised classification, the system is first trained with a known set of multi-dimensional features space. In unsupervised classification, the system is allowed to allocate pixels to data itself based on labels assigned to spectral clusters given by the user. Hybrid classification uses both supervised and unsupervised methods for data learning. It collects training data sets, uses the unsupervised classification for identification of spectral clusters, assigns image with clusters and in the end rejoin the clusters to make a single class/category. In this section, different classifiers and models are discussed which are used for classification purposes namely SVM, Decision Tree (DT), Linear Discriminant Analysis (LDA), Ensemble Bagged Tree (EBT), K-Nearest Neighbors (KNN) and AdaBoost.

A. Support Vector Machine

Support Vector Machine (SVM) is the model of supervised learning. SVMs use very cleverly a large number of features for learning without using additional computational power. They are capable to represent nonlinear functions and able to use efficient algorithms for learning. SVM constructs a hyperplane or sets of hyperplanes in a finite or infinite-dimensional sample space for classification purposes without intermixing of feature vectors. Good separation between classes and data is achieved by a hyperplane whose distance is large to the data point in the nearest training class if the gap is large; the classifier is less likely of generalization error. SVM is used in text and hypertext categorization because it reduces the number of training label instances. It is also used in image classification and language characters recognition etc. The main purpose of the SVM classifier is to find the maximum hyperplane separation in a given data set.

VII. System Operations

There are six main features of our system:

A. Image Acquisition

Human activity images are scanned through a scanner to bring the image in a digital format. The scanned images are cropped so that it only contains the required subject area. The image will be saved with 96 dpi.

B. Image Pre-Processing (that Includes Binarization, Enhancement, etc.)

The third requirement includes the image enhancement process; which includes removing noise, sharpening, brightening, thickening, thinning of images, etc. These operations are applied to images to make it easier to identify the key features.

C. Feature Extraction

The most important requirement of our system is feature extraction. This includes extracting features from the images and then selecting the key features that are used to classify the images. A feature vector fill is formed after the selection of the best features. If the image enhancement process is not done correctly, it would be difficult to extract the best features. The features extracted may classify the images incorrectly.

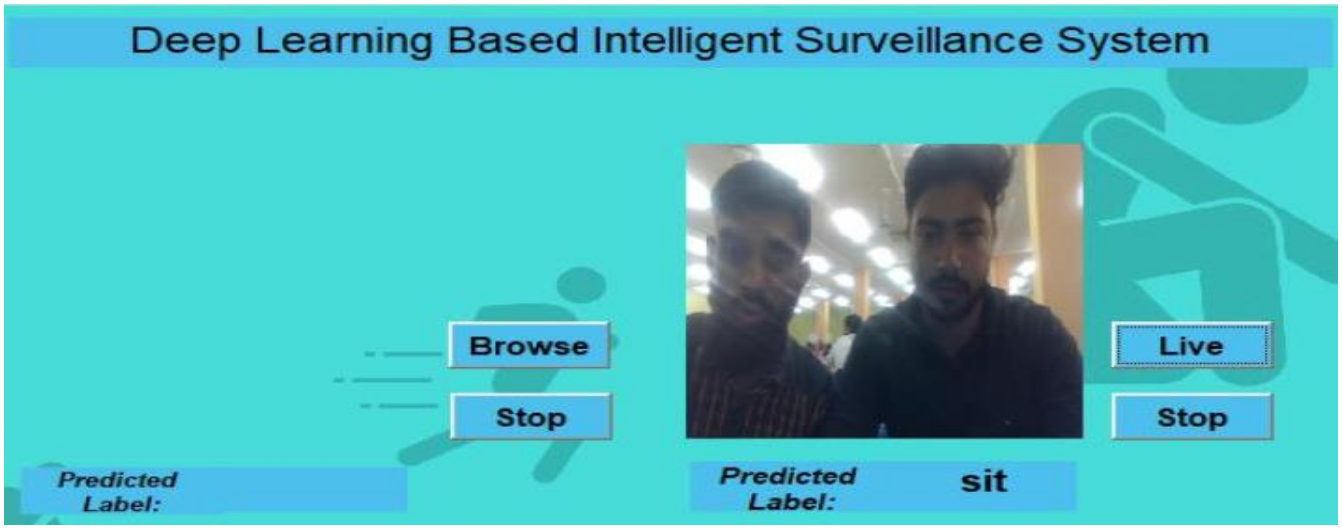

Fig 5. Live Video Capturing Proposed Method (Classification).

D. Training

The system is trained for the classification of images. The system is trained by using the feature vector. The feature vector should be labeled correctly; if the Human Activity is Suspicious then the row containing the features must be labeled Suspicious. If the signature image is normal, then the row containing the features must be labeled Normal. Otherwise, the system is trained for the wrong data. As a result, the classification may be false.

E. Testing

This is an image enhancement process; which includes removing noise, sharpening, brightening, thickening, thinning of images, etc. These operations are applied to images to make it easier to identify the key features.

F. Classification

This requirement is mandatory. The system needs to give output in the form of an Activity label, and it will also suggest if any appropriate action needed. The system can only classify the images if it is trained. The result of classification relies on the best feature selection. Live video capturing proposed method is shown in Fig. 5.

A sequence diagram shows in Fig. 6 the workflow or sequential execution of our system. The figure represents that the user will use the graphical user interface terminal to communicate with the system. First of all, users click on upload training data then terminal call the function of preprocess the data. When the data is pre-processed then the system will verify the data and on the next step, the system will store the trained features and display the message to the terminal that data is stored successfully. After that user can upload testing data and then the system will accept the data and convert it into the gray scale and in the pre-processing step features will be extracted. After feature extraction data is passed to the database for the sake of identifying the label. After that, the predicted name is displayed on the terminal for user verification.

VIII. Performance Evaluation

In order to validate the system, an HP envy machine is used with Intel core i5, RAM 4 GB DDR2-RAM, Digital Camera 16 Mega Pixel, Disk space 8 GB for MATLAB R2018a. Some image processing software such as Photoshop is used to convert the images which are not BMP or jpg format to BMP or jpg images so that they become readable by the system. There would be needed to resize and set the resolution of scanned images. With the MSVM method and feature type Original FC gives us an accuracy of 68.5% and FNR% is 31.9 and for this purpose time consumed is 121.6 sec.

With the MSVM method and feature type Original Pooling gives us an accuracy of 59.3 and FNR% is 40.7 and the time consumed is 198.6 sec. With the MSVM method and feature, type Fusion gives us an accuracy of 94.4 and FNR% is 5.1 and the time consumed is 251.5 sec. With the MSVM method and feature, type Reduction gives us an accuracy of 97.1 and FNR% is 2.9 and the time consumed is 104.3 sec. With the Ensemble method and feature type Original FC gives us an accuracy of 90.6% and FNR% is 9.4 and for this purpose time consumed is 127.9 sec. With the Ensemble method and feature type Original Pooling gives us an accuracy of 79.5% and FNR% is 20.5 and the time consumed is 182.4 sec. With the Ensemble method and feature, type Fusion gives us the accuracy of 94.6 and FNR% is 5.4 and time consumed is 294.9 sec. With Ensemble method and feature, type Reduction gives us an accuracy of 97.2 and FNR% is 2.8 and time consumed is 107.5 sec as shown in Table I.

With Naive Bayes method and feature type Original FC gives us an accuracy of 92.5% and FNR% is 7.5 and for this purpose time consumed is 111.5 sec. With Naive Bayes method and feature type Original Pooling gives us an accuracy of 89.0% and FNR% is 11.0 and time consumed is 188.9 sec. With Naive Bayes method and feature type gives us an accuracy of 96.3% and FNR% is 3.7 and time consumed is 240.5 sec. With Naive Bayes method and feature type Reduction gives us an accuracy of 98% and FNR% is 2 and the time consumed is 92.48 sec.

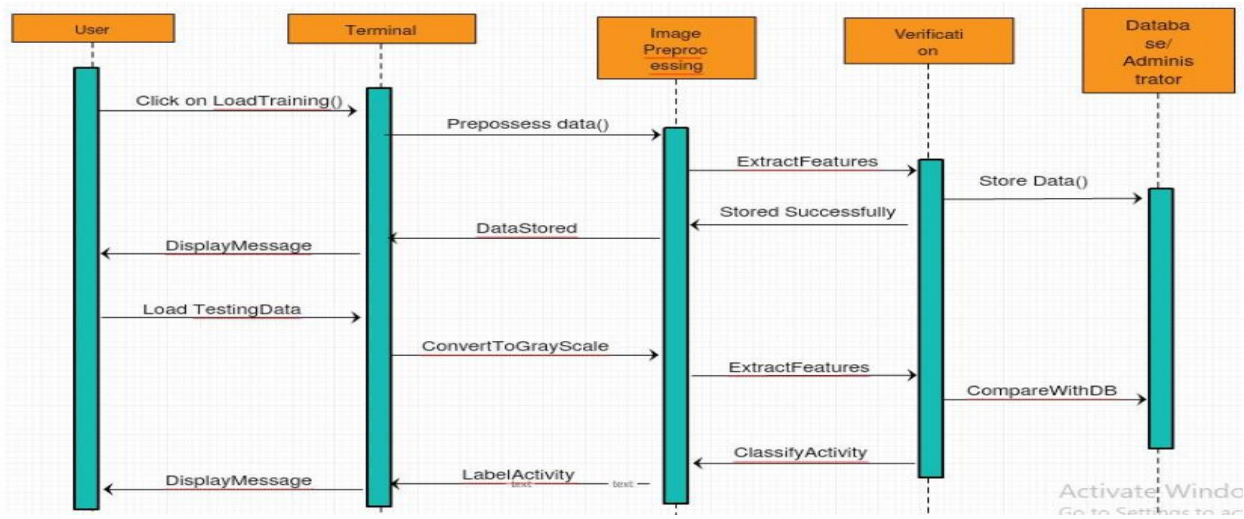

Fig 6. The Sequence of Different Opearation of the System.

TABLE I. HMDB51 DATASET

Method	Features Type				Evaluation Metrics		
	Original FC	Original Pooling	Fusion	Reduction	Time (sec)	FNR (%)	Accuracy (%)
Naive Bayes	Y				111.5	7.5	92.5
		Y			188.9	11	89.0
			Y		240.5	3.7	96.3
				Y	92.48	2	98
MSVM	Y				121.6	31.9	68.5
		Y			198.6	40.7	59.3
			Y		251.5	5.1	94.4
				Y	104.3	2.9	97.1
Ensemble	Y				127.9	9.4	90.6
		Y			182.4	20.5	79.5
			Y		294.9	5.4	94.6
				Y	107.5	2.8	97.2

TABLE II. UCF101 DATASET

Method	Features Type				Evaluation Metrics		
	Original FC	Original Pooling	Fusion	Reduction	Time (sec)	FNR (%)	Accuracy (%)
Naive Bayes	Y				109.54	22.2	77.8
		Y			113.6	36	64
			Y		194.2	18.9	81.1
				Y	69.84	5.0	95
MSVM	Y				107.96	28.9	71.1
		Y			156.6	33.5	66.5
			Y		211.9	23.1	76.9
				Y	89.6	12.5	87.5
Ensemble	Y				116.04	30.5	69.5
		Y			145.5	32.8	67.2
			Y		201.5	26.4	73.6
				Y	74.8	15.7	84.3

TABLE III. HOLLYWOOD

Method	Features Type				Evaluation Metrics		
	Original FC	Original Pooling	Fusion	Reduction	Time (sec)	FNR (%)	Accuracy (%)
Naive Bayes	Y				72.5	4.9	95.1
		Y			68.9	20.2	79.8
			Y		81.3	4.3	95.7
				Y	33.1	3	97
MSVM	Y				76.2	6.9	93.1
		Y			79.7	42	58
			Y		89.2	6	94
				Y	36.3	4.1	95.9
Ensemble	Y				89.4	10.4	89.6
		Y			81.2	31.9	68.1
			Y		93.5	5.6	94.4
				Y	47.4	3.4	96.6

TABLE IV. WEIZZMAN DATA

Method	Features Type				Evaluation Metrics		
	Original FC	Original Pooling	Fusion	Reduction	Time (sec)	FNR (%)	Accuracy (%)
Naive Bayes	Y				104.9	1.9	98.1
		Y			112.6	37.7	62.3
			Y		129.9	1.6	98.4
				Y	59.4	0.6	99.4
MSVM	Y				122.4	5.8	94.2
		Y			137.1	23.1	76.9
			Y		158.9	4.2	95.8
				Y	69.4	1.3	98.7
Ensemble	Y				107.5	9.1	90.9
		Y			119.42	18.33	81.67
			Y		148.9	6.71	93.29
				Y	76.5	4.4	95.6

Fig 7. Proposed Method (Classification).

With MSVM method and feature type Original FC gives us an accuracy of 71.1% and FNR% is 28.9 and for this purpose time consumed is 10796 sec. With MSVM method and feature type Original Pooling gives us an accuracy of 66.5% and FNR% is 33.5 and time consumed is 156.6 sec. With MSVM method and feature type Fusion gives us an accuracy of 76.9% and FNR% is 23.1 and time consumed is 211.9 sec. With MSVM method and feature type Reduction gives us an accuracy of 87.5% and FNR% is 12.5 and time consumed is 89.6 sec in Table II. Also see Fig. 7 for the proposed method.

With Ensemble method and feature type Original FC gives us an accuracy of 69.5% and FNR% is 30.5 and for this purpose time consumed is 116.04 sec. With Ensemble method and feature type Original Pooling gives us an accuracy of 67.2% and FNR% is 32.8 and time consumed is 145.5 sec. With Ensemble method and feature, type Fusion gives us an accuracy of 73.6% and FNR% is 26.4 and time consumed is 201.5 sec. With the Ensemble method and feature type Reduction gives us an accuracy of 84.3% and FNR% is 15.7 and time consumed is 74.8 sec. With Naive Bayes method and feature type Original FC gives an accuracy of 77.8% and FNR% is 22.2 and for this purpose time consumed is 109.54 sec. With Naive Bayes method and feature type Original Pooling gives us an accuracy of 64% and FNR% is 36.0 and time consumed is 113.6 sec. With Naive Bayes method and feature, type Fusion gives us an accuracy of 81.1% and FNR% is 18.9 and time consumed is 194.2 sec. With Naive Bayes method and feature, type Reduction gives us an accuracy of 95% and FNR% is 5.0 and time consumed is 69.84 sec.

With MSVM method and feature type Original FC gives us an accuracy of 93.1% and FNR% is 6.9 and for this purpose time 42 and time consumed is 79.7 sec. With MSVM method and feature type Fusion gives us an accuracy of 94% and FNR% is 6 and time consumed is 89.2 sec. With MSVM method and feature type Reduction gives us an accuracy of 95.4% and FNR% is 4.1 and time consumed is 36.3 sec shown in Table III.

With Ensemble method and feature type Original FC gives us time consumed is 89.4 sec. With Ensemble method and feature type Original Pooling gives us an accuracy of 68.1% and FNR% is 31.9 and time consumed is 81.2 sec. With Ensemble method and feature type Fusion gives us an accuracy of 94.4% and FNR% is 5.6 and time consumed is 93.5 sec. With Ensemble method and feature type Reduction gives us an accuracy of 96.6% and FNR% is 3.4 and time consumed is 47.4 sec.

With Naive Bayes method and feature type Original FC gives us an accuracy of 95.1% and FNR% is 4.9 and for this purpose time consumed is 72.5 sec. With Naive Bayes method and feature type Original Pooling gives us an accuracy of 79.8% and FNR% is 20.2 and time consumed is 68.9 sec. With Naive Bayes method and feature type Fusion gives us an accuracy of 95.7% and FNR% is 4.3 and time consumed is 81.3 sec. With Naive Bayes method and feature type Reduction gives us an accuracy of 97% and FNR% is 3.0 and time consumed is 33.1 sec. With MSVM method and feature type Original FC gives us an accuracy of 94.2% and FNR% is 5.8 and for this purpose time consumed is 122.4 sec. With MSVM method and feature type Original Pooling gives us an accuracy of 76.9% and FNR% is 23.1 and time consumed is 137.7 sec.

With MSVM method and feature type Fusion gives us an accuracy of 95.8% and FNR% is 4.2 and time consumed is 158.9 sec. With MSVM method and feature type Reduction gives us an accuracy of 98.7% and FNR% is 1.3 and time consumed is 69.4 sec as shown in Table IV.

With Ensemble method and feature type Original FC gives us an accuracy of 90.9% and FNR% is 9.1 and for this purpose time consumed is 107.5 sec. With Ensemble method and feature type Original Pooling gives us an accuracy of 81.67% and FNR% is 18.33 and the time consumed is 119.42 sec. With Ensemble method and feature, type Fusion gives us an accuracy of 93.29% and FNR% is 6.71 and time consumed is 148.9 sec. With the Ensemble method and feature, type Reduction gives us an accuracy of 95.6% and FNR% is 4.4 and the time consumed is 76.5 sec.

With Naive Bayes method and feature type Original FC gives us an accuracy of 98.1% and FNR% is 1.9 and for this purpose time consumed is 104.9 sec. With Naive Bayes method and feature type Original Pooling gives us an accuracy of 62.3% and FNR% is 37.7 and the time consumed is 112.6 sec. With Naive Bayes method and feature, type Fusion gives us an accuracy of 98.4% and FNR% is 1.6 and the time consumed is 129.9 sec. With Naive Bayes method and feature,

type Reduction gives us an accuracy of 99.4% and FNR% is 0.6 and the time consumed is 59.4 sec.

IX. Conclusion

In this work, we proposed an approach for both human detection and classification of a single person activity recognition. For this purpose, we implemented the pre-processing techniques. In the first step, we apply the top hat filter by adjusting intensity values to improve the quality of images. In the next step, for edge detection, a weighted based segmentation technique is used and then hybrid feature extraction methods are applied to identify the human actions. These extracted features are fused on serial-base fusion used for classification. To verify our proposed techniques four data sets are considered i.e., HOLLYWOOD, UCF101, HMDB51, and WEIZMANN. These datasets are used for action recognition and feature extraction. The proposed techniques performed significantly better as compared to existing technologies and achieve the following accuracy rates 94.1%, 96.9% and 98.1%, respectively.

References

[1] Janakiramaiah, B., Kalyani, G. & Jayalakshmi, A. Automatic alert generation in a surveillance systems for smart city environment using deep learning algorithm. Evol. Intel. (2020).

[2] I. C. Duta, J. R. Uijlings, B. Ionescu, K. Aizawa, A. G. Hauptmann, and N. Sebe, "Efficient human action recognition using histograms of motion gradients and VLAD with descriptor shape information," Multimedia Tools and Applications, pp. 1-28, 2017.

[3] Sreenu, G., Saleem Durai, M.A. Intelligent video surveillance: a review through deep learning techniques for crowd analysis. J Big Data 6, 48 (2019).

[4] Baba, M.; Gui, V.; Cernazanu, C.; Pescaru, D. A Sensor Network Approach for Violence Detection in Smart Cities Using Deep Learning. Sensors 2019.

[5] R. Slama, H. Wannous, M. Daoudi, and A. Srivastava, "Accurate 3D action recognition using learning on the Grassmann manifold," Pattern Recognition, vol. 48, pp. 556-567, 2015.

[6] M. A. Uddin, J. B. Joolee, A. Alam, and Y.-K. Lee, "Human Action Recognition Using Adaptive Local Motion Descriptor in Spark," IEEE Access, vol. 5, pp. 21157-21167, 2017.

[7] L. Xia, C.-C. Chen, and J. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on, 2012, pp. 20-27.

[8] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, 2005, pp. 1395-1402.

[9] 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, 2004, pp. 32-36.

[10] S. Singh, S. A. Velastin, and H. Ragheb, "Muhavi: A multicamera human action video dataset for the evaluation of action recognition methods," in Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on, 2010, pp. 48-55.

[11] Kulathumani, "WVU Multi-view action recognition dataset available on: http://csee.wvu.edu/~vkkulathumani/wvu-action.html#download2."

[12] J. Yuan, Z. Liu, and Y. Wu, "Discriminative video pattern search for efficient action detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 1728-1743, 2011.

[13] J. Yuan, Z. Liu, and Y. Wu, "Discriminative video pattern search for efficient action detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 1728-1743, 2011.

[14] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on, 2010, pp. 9-14.

[15] J. Liu, Y. Yang, I. Saleemi, and M. Shah, "Learning semantic features for action recognition via diffusion maps," Computer Vision and Image Understanding, vol. 116, pp. 361-377, 2012.

[16] J. Liu, Y. Yang, and M. Shah, "Learning semantic visual vocabularies using diffusion distance," in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, 2009, pp. 461-468.

[17] M. S. Ryoo and J. K. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in Computer vision, 2009 ieee 12th international conference on, 2009, pp. 1593-1600.

[18] U. Gaur, Y. Zhu, B. Song, and A. Roy-Chowdhury, "A "string of feature graphs" model for recognition of complex activities in natural videos," in Computer Vision (ICCV), 2011 IEEE International Conference on, 2011, pp. 2595-2602.

[19] S.MALLICK, "Histogram of Oriented Gradients; https://www.learnopencv.com/histogram-of-oriented-gradients/," December, 2016.

[20] Y. Coque, E. Touraud, and O. Thomas, "On line spectrophotometric method for the monitoring of colour removal processes," Dyes and pigments, vol. 54, pp. 17-23, 2002.

[21] A. Ramakrishnan, S. K. Raja, and H. R. Ram, "Neural network-based segmentation of textures using Gabor features," in Neural Networks for Signal Processing, 2002. Proceedings of the 2002 12th IEEE Workshop on, 2002, pp. 365-374.

[22] C. Cortes and V. Vapnik, "Support-vector networks," Machine learning, vol. 20, pp. 273-297, 1995.

[23] I. C. Duta, J. R. Uijlings, B. Ionescu, K. Aizawa, A. G. Hauptmann, and N. Sebe, "Efficient human action recognition using histograms of motion gradients and VLAD with descriptor shape information," Multimedia Tools and Applications, pp. 1-28, 2017.

[24] C. Chen, M. Liu, H. Liu, B. Zhang, J. Han, and N. Kehtarnavaz, "Multi-Temporal Depth Motion Maps-Based Local Binary Patterns for 3-D Human Action Recognition," IEEE Access, vol. 5, pp. 22590-22604, 2017.

[25] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," Computer Vision–ECCV 2010, pp. 143-156, 2010.

[26] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," Neurocomputing, vol. 70, pp. 489-501, 2006.

[27] D. C. Luvizon, H. Tabia, and D. Picard, "Learning features combination for human action recognition from skeleton sequences," Pattern Recognition Letters, 2017.

[28] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal LSTM with trust gates for 3D human action recognition," in European Conference on Computer Vision, 2016, pp. 816-833.

[29] A. Veenendaal, E. Jones, Z. Gang, E. Daly, S. Vartak, and R. Patwardhan, "Dynamic Probabilistic Network Based Human Action Recognition," arXiv preprint arXiv:1610.06395, 2016.

[30] Z. Zhang, S. Liu, C. Wang, B. Xiao, and W. Zhou, "Multiple Continuous Virtual Paths Based Cross-View Action Recognition," International Journal of Pattern Recognition and Artificial Intelligence, vol. 30, p. 1655014, 2016.

[31] Z. Gao, H. Zhang, G. Xu, Y. Xue, and A. G. Hauptmann, "Multi-view discriminative and structured dictionary learning with group sparsity for human action recognition," Signal Processing, vol. 112, pp. 83-97, 2015.