A novel hybrid feature extraction and ensemble C3D classification for anomaly detection in surveillance videos

Vishnu Priya Thotakura, Purnachand Nalluri
School of Electronics Engineering, VIT-AP University, Amaravati, India

Article Info

Article history:

Received Jun 24, 2022
Revised Jan 12, 2023
Accepted Jan 14, 2023

Keywords:

Action detection
Convolution neural network
Feature extraction
Support vector machine
Video dataset

ABSTRACT

Anomaly detection in several deep learning frameworks are recently presented on real-time video databases as a challenging task. However, these frameworks have high false positive rate (FPR) and error rate due to various backgrounds, motion appearance and semantic high-level and low-level features for anomaly detection through action classification. Also, extraction of features and classification are the major problems in traditional convolution neural network (CNN) on real-time video databases. The proposed work is a novel action classification framework which is designed and implemented on large video databases with high true positive rate (TPR) and error rate. In this framework, Kalman based incremental principal component analysis (IPCA) feature extraction method; C3D and non-linear support vector machine (SVM) classifier are used to improve the action prediction (anomaly detection) on the large real-time video databases. The proposed frame work shown new results of high computation performance than the traditional deep learning frameworks for action classification.

Corresponding Author:

Purnachand Nalluri
School of Electronics Engineering, VIT-AP University
Amaravati-522237, Andhra Pradesh, India
Email: chanduinece@gmail.com

1. INTRODUCTION

In the field of robotics, human-computer interaction and video monitoring applications, action recognition plays a vital role to find and track the human motions for anomaly detection. Various datasets and benchmarks for action recognition have been used for anomaly detection. The image classification has improved considerably for object detection by Ullah *et al.* [1], scene classification defined by Karpathy *et al.* [2], and feature classification by Schölkopf *et al.* [3]. Traditional convolution neural network (CNN) architectures, from image to video, have made considerable progress in many image and video-based surveillance applications. Detecting anomalies is critical issue which is being explored within distinct research fields and applications. The term anomaly refers to thing or event which is not accommodate to normal behavior. Anomaly recognition is all about finding the real-world things in the given training data which are not accommodate to normal behavior. The non-accommodate events in the data are also termed as exceptions or outliers or discordant conclusions or peculiarities or surprise or contaminants in various application fields described by Nair [4]. In the action anomaly detection, human behavior in videos has gained a lot of attention. Action detection in realistic video datasets such as movies by Zhang *et al.* [5], web videos by Feichtenhofer *et al.* [6] and television shows Draper [7]. Action identification continues to be a challenging issue due to variation in action class shown in Figure 1. In the same action class, there is a large intraclass difference that can be caused by background clutter, change of perspective and different movement speeds and styles.

Supervised learning based anomaly recognition methods are exceptional in performance when correlated with unsupervised techniques. The reason is that, these methods used labeled samples. The learning mechanism in supervised anomaly detection follows the concept of barriers from a set of explained trained data points, since data is trained and labeled, a text data point can be easily classified and named it as normal of abnormal activity class by Draper [7]. They implemented a hybrid multi-class support vector machine (SVM) framework in order to find the anomaly objects in the training data based on the class labels. These multi class detection methods are trying to learn a classifier to differentiate between abnormal classes from the remaining classes. Generally supervised models of anomaly detection have two alternative networks, namely feature extraction and classifier. The computational performance of deep learning based supervised anomaly detection techniques are purely dependent on the input data dimensionality factor and the hidden layer number, which are trained by using back propagation technique. When the input data is a high dimensional then the network needs to maintain more number of hidden layers meaning-meaningful learning of input data features.

Semi-supervised anomaly detection, this category is also known as one-class classifier. Semi-supervised learning (SSL) is a more recent and unsupervised approach. SSL learns from the combination of labeled and unlabeled patterns that may enhance the tasks of classification/clustering. Multiple real-world apps, viz. processing of images and classification of text require experts to label the unlabeled information, which is an expensive process.

Unsupervised learning based anomaly detection, unsupervised anomaly detection is a critical field of research in both industrial applications and basic machine learning (ML) research. Many frameworks of description logics (DL) models addressing the challenging issues of anomalies identification process of all the model architectures. The fundamental architecture is the auto encoder's architecture given by Tran *et al*. [8]. The deep models which are used for unsupervised anomaly detection depends on three assumptions: The reason belongs to the normal events can be separable from the reason of abnormal events in the given original piece of data or frame. The major part of the data comes under normal data points. The detection method generates an outlier score data points depending on inherent properties of the dataset.

The main problems of the traditional models include: i) Problem of detection motion-based features for the video anomaly detection process; ii) Traditional models have high error rate or true negative rate due to noise in the training data classes; and iii) Difficult to find the contextual relationships in the inter and intra features for the anomaly detection process.

The main contribution of the work are summarized: i) Implemented a hybrid feature extraction using Kalman filter measure; ii) Implemented a novel feature ranking using improved principal component analysis (PCA) approach; and iii) Proposed a hybrid C3D boosting classifier for the anomaly detection process.

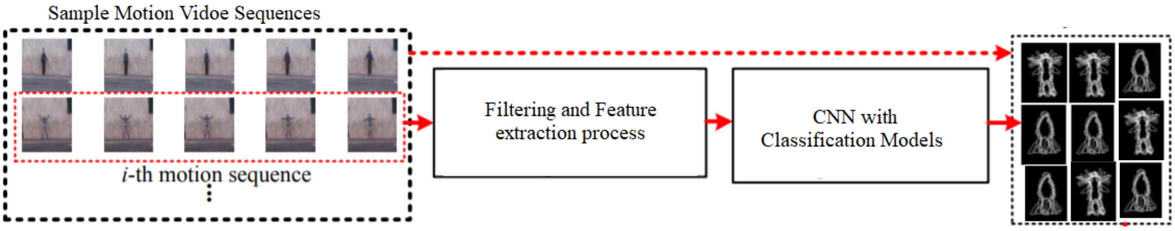

Sample Motion Vidoe Sequences
i-th motion sequence
Filtering and Feature extraction process
CNN with Classification Models

Figure 1. Basic action detection steps

High dimensional anomaly feature extraction: It is the key element of the detection system for human activity. Since human actions appear in the video with different orientations, texture, and intensities; the essential features in the large video datasets are difficult to find. The main purpose of extracting a feature is to provide a perfect correlation between an action sequence that is memory-efficient, true positive rate and error rate computationally defined by Tran *et al*. [9]. In the classification of video action process, selection of feature models are generally divided into two types, wrappers approaches and filter approaches. Wrapper approach estimates feature subset or each feature to enhance the classification accuracy. Filter method estimates each feature independent from the classification algorithm, ranks the action features after assessment and considers the best one. In general, the speediness of wrapper model is relaxed than the filter model because of cross validation and repeated iteration to evaluate the feature subsets. Traditional wrapper model is more efficient because classification technique marks the overall accuracy, although the subset selection is an NP-hard. However, the increment of complex data depends on the amount of features involved; finding new action patterns can become difficult due to the complex relationships among features. The measure for each function is calculated by feature ranking techniques and ranked accordingly. These ranking methods handpicked the highest rank-predicated topmost 'k' features and eliminate those with lower ranks by Tran *et al*. [9]. CNNs are the most effective methods of action classification. Traditional deep learning research used CNNs for the identification of human actions in each frame and then for the development of time-space tubes by Dawn and Shaikh [10]. Simonyan et al. use a two-stream CNN that operates with a single frame and different optical flow frames. Also, if the use of the temporal stream takes advantage of video motion and improves accuracy, separate optical flow calculations are required for each video is given by Georgios Th. Papadopoulos and Daras [11]. 3D CNNs showed a positive extraction of spatio-temporal features for action classification. The 3D kernels allow the CNN to learn details about time and movement directly from the video frames. The baseline action detection model is implemented to extend the action localization by using interactive three-dimensional (I3D) network. As the complexity of these networks increases, the large number of parameters is more difficult to optimize. In addition, high dimensions and low video resolution further increase the difficulty of action detection in various action videos. To aggregate these feature maps into effective descriptors, Wang *et al*. [12] used the CNN framework to learn conventional feature maps and trajectory-constrained pooling. In order to ensure that the characteristics learned are discriminatory, the CNN framework is used in the next stage to find the required features for the problem of classification or prediction. Deep learning (DL) based anomaly detection systems have been to learn complex and ranked feature relations inside high-dimensional unprocessed data in [13]. The quantity of layers utilized in deep learning-based anomaly detection procedures is driven by the dimensionality of the input data; the deep learning systems are appeared to create better execution on high dimensional input data.

Due to high dimensionality and high false positive rates, CNN based static filtering methods are inefficient for processing large video datasets. Therefore, we propose in this paper an effective and optimized C3D CNN-based framework for the extraction and classification process of the action feature. A C3D model is pre-trained and its use in this model is to find and detect the anomaly objects of the action frame for the problem of classification. The creation of C3D network takes place with the intention that the identification process of moving objects and 3D-images are to be improved. With these intentions every C3D network has to possess three important properties; i) The feature extraction of C3D network should be generic. In order to support various types of videos; ii) It should have compact representation which is very much advantageous for processing, repository and retrieval; and iii) The network should show efficient performance in computing features from huge number of videos Liu *et al*. [14] and Xu *et al*. [15].

Next, to learn the temporary modifications in actions to predict the action, an optimized classification model is introduced. Furthermore, non-linear SVM is trained through a non-linear learning approach to classify human actions. The main problems of the traditional models include: i) Problem of detection motion-based features for the video anomaly detection process; ii) Traditional models have high error rate or true negative rate due to noise in the training data classes; and iii) Difficult to find the contextual relationships in the inter and intra features for the anomaly detection process.

The main contribution of the work are summarized: i) Implemented a hybrid feature extraction using Kalman filter measure; ii) Implemented a novel feature ranking using improved PCA approach; and iii) Proposed a hybrid C3D boosting classifier for the anomaly detection process.

2. METHOD

Feature detection is one the basic step of any anomaly detection systems; performance and accuracy can be significantly degraded by poor descriptor choice. In volume length between perpendiculars (VLBP), the binary pattern histogram encodes local volumes by Wang and Schmid [16]. Despite its simplicity, the number of separate patterns generated in neighborhoods' regions by VLBP may become overwhelming. The convolution architecture efficiently uses the image structure by "pooling" and "weight-sharing" to reduce the search space of the network. Pooling and weights initialization help to achieve robustness across differences in scale and space. To optimize this issue, they introduce 3D convolution networks. Traditional models are focused on constructing efficient descriptors or characteristics and then classifying them based on matching features. Here, feature selection measures or filters are used to detect different types of human action classes in anomaly detection process. Global features include silhouette-based descriptors, edge-based features, optical flow-based display and movement history image (MHI). is used in CNN models Azim and Hemayed [17]. Occlusions, changing viewpoints, and noise often create problem in global features. Local characteristics always use image patches separately, and then these patches are combined to create a space-time models such as SURF and histogram of oriented gradients (HOG). Local descriptors, particularly for noise images and partly occluded images can present video action more efficiently. Convolution neural networks (CNNs) have proven to be strong feature extraction model for still image recognition. Most traditional systems are based on movement evaluation and bag-of-words (BoW) Wang *et al*. [18], but BoW 's development is costly in terms of computation and requires high computational runtime. The main objective of OFCM model is to find the flow field and its local neighborhoods, which play a significant role in motions representation. Recently, deep convolutionary neural networks (DCNNs) and long recurrent convolutionary networks (LRCNs) Wei *et al.* [19] are used to improve the action detection rate in many computer vision applications. These methods use back propagation to correctly recognize the hidden pattern in visual information, so characteristics are auto-extracted without manual selection.

They presented LSTM network model for the detection of anomalies under unsupervised category. The advantage of using LSTM is that they can capture the situations or scenes with increase in time slots. Chianucci and Savakis spatial transformer networks which contains deep structures with the combination of LSTM and CNNs. This integration of CNN and LSTM given the flexibility of extracting spatio-temporal features as shown in Figure 2. These features showed a prominent way in detecting anomalies.

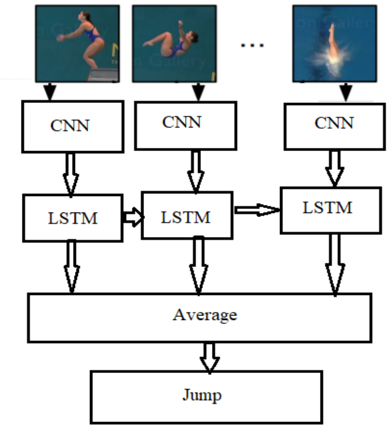


Figure 2. Long short-term memory (LSTM) based action detection

Zhou *et al.* [20] worked on GAN network model. The main advantage with this GAN is that they have the learning ability of input distributions. Because of this ability the generative adversarial network (GAN) dependent models had shown outstanding performance in detecting the anomalies on complex as well as high dimensional data sets. Because the auto encoders are the most general architecture in unsupervised category for detecting anomalies, the problem of optimization is non-convex. Finding and annotation of human classes in static images is a challenging issue in real-time video datasets. Motion vectors are the essential parameters for understanding human action from the video. Most of the existing action detection approaches, which consider the temporal function as a separate source and do not use feature extraction and parameter optimization. 3D CNN is an extended solution to the anomaly detection process.

Dai [21] implemented a C3D CNN-based human motion recognition for segmenting human objects in videos. To recognize large-scale intervention issues, Bilen *et al*. [22] implemented 3D CNN. Henawy *et al*. [23] suggested 3D CNN factorization to detect actions in videos and introduced various levels of convolution kernels. Two-stream region-based convolutional neural network (R-CNNs) are implemented for human activity detection by Laptev [24], RPN and motion RPN are used to find actions in each level. In action recognition, both global template Laptev *et al*. [25] and bag-of-word model Carmona and Climent [26] have been used to classify the essential features and actions. Most of the action detection models are used for multimedia retrieval, video surveillance and human computer interaction applications. Also, these approaches use a low-level discriminative characteristic in order to accurately recognize human actions from videos, including space-temporality points of interest (STIP) Donahue *et al.* [27], shape, optical flow characteristics by Shimazaki and Nagao [28], and trajectory representations in Simonyan and Zisserman [29]. Due to large variations in video backgrounds, feature extraction and classification operations are performed differently. Kang and Wang [30] suggests the dictionary of cuboid prototypes to understand human actions on small video datasets. Laptev *et al.* [25] designed a gradient histogram (HOG) and flow histogram (HOF) features to replace the spatio-temporal pyramid in Singh *et al*. [31] process.

Liu *et al.* [32] used a dense sample taken from object recognition for performing action recognition. Sun *et al.* [33] suggested using Kernel PCA to identify actions and to reduce the dimensions during the pre-processing phase. Subedar *et al.* [34] implemented a new feature selection model to detect the actions in videos by using the linear discriminant analysis (LDA) and PCA. Most of the traditional feature selection models use local descriptors for feature ranking and feature subset selection. Flow trajectory and "STIPs" in Latah [35] are the two popular local descriptors in static activity detection process. "A new motion trajectory method was proposed by Javan *et al.* [36] for the extraction of essential features set using" motion boundary histogram "and" gradient histogram" Chaquet *et al.* [37]. Trajectories are computed here using the vectors of optical flow. A trajectory-based human action detector that captures discriminatory temporal relationships was implemented by Ribeiro *et al.* [38]. The trajectories are extracted based on SIFT descriptors in this model. A "support vector machine (SVM)" classification model is provided with the trajectory points obtained in the feature extraction process. Koperski [39], dense paths were further improved. The camera movement was highlighted and was tried to remove as a main obstacle for extracting objects of interest. The authors first use SURF and dense optical flow descriptors to match the feature points and then estimate homography using the random sample consensus (RANSAC) by Jaouedi *et al.* [40]. This approach explicitly identifies and removes the camera motion. SVM determines the relationships between the frame objects and the videos labels in order to presage the incipient type of action.

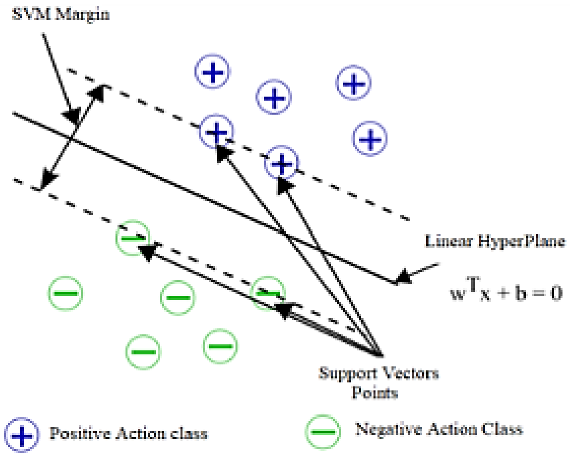


Figure 3. SVM classification model

The spatial branches receive the video frames individually and implement action identification based on still or constant images. That is this wing of two flow architecture behaves like an image classifier. SVM is a technique of supervised learning which is used to classify the anomaly objects in one class or multi-class way. Figure 3 shows a hyper plane with margin and support patterns in a two-dimensional space. SVM classifier is integrated with CNN framework in order to classify the feature sets and to detect the action classes.

3. PROPOSED METHOD

The recommended model is considered and employed in three phases i.e. feature extraction phase, deep learning phase and action classification or prediction phase as shown in Figure 4. In the feature extraction phase, essential motion features in the video sequences are ranked using the Kalman based PCA feature extraction method. These ranked features are used as input to the deep learning phase. A pre-trained C3D framework is used in the deep learning phase, to filter the essential features in each video sequence motion vectors. Finally, a new boosting classifier is realized to predict the action classes in the classification phase. In this stage, a hybrid non-linear SVM and Bayesian classifiers are recycled to train the C3D features for action class prediction.

3.1. Kalman nearest neighbor based PCA feature extraction

In this phase, sequence action frames are taken as input for feature extraction and ranking. Each frame is divided into blocks to find the most proximate neighbor kineticism features along with its variations. Let Bi and Bj, represents the block partitions with mean and covariance matrices of blocks ith and jth blocks of frame. Feature extraction and ranking process are performed in two steps. In the first step, features are extracted block wise using the neighborhood Neigh equation. Each block and its neighboring blocks in this step are to find and mark the motion vectors in each frame. All these nearest motion features (vectors) are used as input to feature ranking process (step 2). An incremental PCA method is implemented to filter the ranked features based on the motion vectors for action detection process.

Phase1: Feature extraction
Input: Sequence action of frames.
Step1: Frame is partitioned in to blocks.

$$Kh_{I,t'}(p_i,p_j) = \sum_{i=0}^{\infty} e^{-E_i t}\ \varphi_i\left(p_j\right) \varphi_i\left(p_i\right) \tag{1}$$

Where$E_i, \ldots \geq 0$ are the Eigen values and φ_i represents the Eigen vectors.
Find the Kalman based nearest neighbour motion features along with its variations.
Let B_i and B_j, represents the block partitions with means and covariance matrices of blocks ith and jth blocks of frame.
Step2: Features are extracted block wise using the neighbourhood Neigh equation.

$$\mathrm{Neigh}_{ij} = \begin{cases} e^{-\min\left\{d_G(B_i, B_j)^2, d_i(B_i, B_j)^2\right\}/Kh_{I,t'}(p_i,p_j)} & \text{if } p_i \text{ and } p_j \text{ neighbors} \\ & 0, \text{otherwise} \end{cases}$$

$$d_G\left(B_{i,}B_j\right) = \left(\overline{B_\imath} - \overline{B_\jmath}\right)^T \left(\frac{\mathrm{Cov}_{B_i} - \mathrm{Cov}_{B_j}}{2}\right)^{-1} \left(\overline{B_\imath} - \overline{B_\jmath}\right) + \frac{1}{2}\mathrm{Xln}\left(\frac{\frac{\mathrm{Cov}_{B_i} - \mathrm{Cov}_{B_j}}{2}}{\sqrt[2]{\left|\mathrm{Cov}_{B_i}\right|\left|\mathrm{Cov}_{B_j}\right|}}\right) \tag{2}$$

$$d_t\left(B_{i,}B_j\right) = \sqrt{\mathrm{trace}\left(\log^2\left(B_i^{-\frac{1}{2}} B_j B_i^{-\frac{1}{2}}\right)\right)} \tag{3}$$

In the Incremental PCA algorithm, motion features F are taken as input for covariation matrix computation. Lines 1-8 in step2 represent the covariance matrix computation on the input motion vectors. The step2 represents the vectors computation of Eigen value and covariance matrix. The optimal Eigen sum of the PCA is used to filter the essential correlated features for principal component scoring. Lastly the algorithm represents the sorted ranked features with high Eigen values. Finally, highest motion Eigen features are marked as positive bag and others as negative bag for C3D framework.

Feature ranking using IPCA:

1. Initialize the motion features F using the Kalman filtering approach. Find the covariance computation between the motion features F.
2. Let motion features MF is represented as {mf[0],mf[1]….mf[k]}. Where k be the size of the features space.
3. The combinational candidate sets of the motion features are represented as CMF={(mf[0],mf[1]),(mf[0],mf[2]),(mf[0],mf[3])……(mf[k],mf[0])….}. In the given features size k, the number of motion feature combinations is computed as $\frac{k!}{(k-2)!2!}$ candidate sets.
4. For each pair of candidate features CF.
5. Do.
6. Compute covariance between features as:

$$\text{Cov (CF\{x,y\})}=\frac{\sum_{i=1}^{n}(CF[x_i]-\mu_{CF[x_i]})(CF[y_i]-\mu_{CF[y_i]})}{(n-1)} \tag{4}$$

7. Done.
8. Compute the Eigen vector and values.
9. Eigenvalues [] = $Det(\lambda I - COV(CF)) = 0$.
10. In this algorithm, I represent the identity matrix of covariance matric COV. The Eigen value vector of the covariance matrix is computed based on the COV matric and it is represented as:

    $(\lambda I - COV(CF))$ v = 0.

11. The optimal Eigen sum score of the covariance matrix is computed to find the best eigen motion features in the anomaly objects by using the following formula.

    $Contingency\_Table(S): C\_T(F,S)$

$$DFCE\left(C_{T(F,S)}\right) = -\frac{(\sum C_i \log (\sum C_i).\sqrt[3]{\sum F_i}).Correlation(F_i,C)}{\chi(F_i,C).\sum P(C_i/F_i)} \tag{5}$$

$$\text{Optimum Eigen} = \frac{\sum \text{Eigenvalues[i]})}{\text{K smallest val(Eigenvalues[i],0.5}*\text{(Maxindex(Eigenvalues[i])+Min index(Eigenvalues[i])}} * DFCE\ Score \tag{6}$$

12. Sort the cumulative Eigen sum score in the step 12.

Select the highest cumulative Eigen score as principal components in the step 12. The highest cumulative Eigen sum scores are labelled as positive bag and lowest cumulative sum scores are labelled as negative bags.

C3D framework:

Input: Positive and negative features.

Output: Filtered features from C3D framework.

Most traditional C3D structures use 3x3x3 kernels to filter the features in the action frames. C3D has 8 turnover layers, 5 pooling layers, 2 fully connected layers, and the softmax output layer in the proposed model. All 3D filters with 1x1x1 steps have 3x3x3 steps with 2x2 steps with 3D layers of 2x2x2 layers. There are 4,096 output units in each fully connected layer. The C3D network has 8 convolution layers, 5 pool layers and 2 fully connected layers, followed by a softmax output layer. All 3D kernels are fitters of three x three x three in space and time. The C3D network has 8 converting layers, five max pooling layers and two fully connected layers, followed by the softmax output layer. Each layer contains the number of filter. An 8frame overlap between the two successive clips divides a video to 16 frame-long videos to extract the C3D features from the rankings shown in Figure 5.

3.4. Classification models Deep CNN-Boosting Classifier

In the proposed classification model, a hybrid SVM classifier is designed and implemented in the deep learning framework C3D. In this section, a hybrid boosting classifier is implemented to test the majority voting of each action class. In this work, two hybrid classifiers i.e non-linear SVM and hybrid Bayesian network are ensemble to predict the action class of the input video. This boosting classifier is used to improve the prediction rate of the anomaly classes on the selected feature sets.

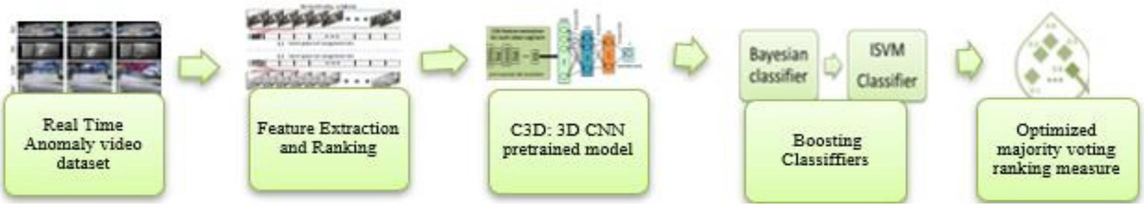

Real Time Anomaly video dataset
Feature Extraction and Ranking
C3D: 3D CNN pretrained model
Bayesian classifier
ISVM Classifier
Boosting Classiffiers
Optimized majority voting ranking measure

Figure 4. Proposed Feature selection-based CNN-ISVM framework

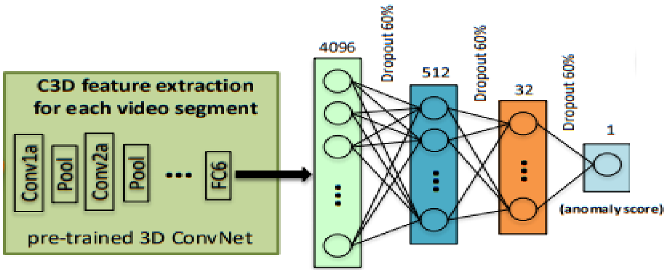


Figure 5. C3D framework

Deep CNN-SVM Classifier:

In the hybrid non-linear SVM approach, an efficient non-linear kernel function is used to transform the feature space and to predict the anomaly class in each video frame.

Input: C3D positive and negative bags and its class labels

Step 1: Initializing the non-linear SVM hyper parameters and kernel function.
Step 2: To each feature set in the feature extraction process, apply a non-linear SVM function and its kernel function on the input video frame.
do
Step 3: Construct a non-linear optimization function for the SVM model. In this proposed model, the optimization function is minimized to improve the error rate of the prediction. The boosted CNN-SVM classifier and its non-linear kernel function are applied on the feature sets for class prediction.
For each feature set do
Apply SVM multi-class optimization models as:

$$min_{W_k,a_k} \frac{1}{2}\|W_k\|_1^2 + \tau_m + \sum_{i=1}^{l} a_i(y_i[ker\langle x,y\rangle.w + b]1 + \xi_i) - \sum_{i=1}^{l} \gamma_i \xi_i \tag{7}$$

$s.t\ ker\langle x,y\rangle.w + b \ \geq 1 - \xi_i^n - \ \tau_m$
$\xi_i^n > 0, \tau_m > 0;\ m = 1 \dots\dots classes$
Here kernel function $\ker\langle x,y\rangle$ represents the kernel functions defined from C3D features space to classes.

$$\ker\langle x,y\rangle = \begin{cases} e^{-\xi_i^n \log(\sum\|x-y\|^2\ if\ x=y} \\ e^{-\xi_i^n \log(\sum\|x-y\|^{\frac{1}{2}}\ if\ x<y} \\ e^{-\xi_i^n \log(\sum\|y\|^2\ if\ x>y} \end{cases} \tag{8}$$

Step 4: Test data is predicted to the class y based on the largest decision values as:

$$argmax\{W_k^T D_i + b_k\}$$

Hybrid Bayesian Network Classifier

Proposed Bayesian network has two steps to predict the action label on the C3D features data. In the first step, both numerical parameters are predicted using the joint probability estimation. In the second step, statistical Bayesian DAG graph structure is learned using the optimized score functions.

Hybrid Bayesian Network Classifier
Step 1: D = Input C3D feature data.
Step 2: For each feature A in D
Step 3: Computing conditional probabilities to each input feature for joint probability estimation.
Step 4: // Parameter estimation step

Discrete parameter estimation in the Bayesian network can be predicted using the following measure.

$$P\left(A_i = \left(I_k \middle| c_j\right)\right) = \frac{N_{ijk}}{N_j}$$

Where N_{ijk} is the number of instances of class c_j having the value I_k in attribute A_i.
Step 5: Continuous parametric estimation in the Bayesian network can be estimated using the following measure.

$$\mathrm{P}(\mathrm{A_i} = \mathrm{I_k}/\mathrm{c_j}) = \mathrm{G}(\mathrm{I_k}, \mu_{ij}, \sigma_{ij})$$

$$\mathrm{G}(\mathrm{I_k}, \mu_{ij}, \sigma_{ij}) = \frac{1}{\sqrt{2\pi\sigma_{ij}}} e^{\frac{(\mathrm{I}_k - \mu_{ij})^2}{2\sigma_{ij}^2}}$$

Here normal distribution is approximation to Gaussian distribution.
Step 6: Estimating Bayesian parameter using the traditional parameter estimation as:

$$\hat{\theta}_{ijk} = \frac{N_{ijk} + \eta_{ijk}}{N_{ij} + \eta_{ij}}$$

where $N_{ij} = \sum N_{ijk}$; $\eta_{ij} = \sum \eta_{ijk}$
Step 7: //Model score estimation
The proposed Bayesian score is computed as:

$$\mathrm{fscore}_k = \sum \Gamma \log(\alpha_i + N_i)$$
$$\mathrm{nsumcnt} = \sum \alpha_i * (\sqrt{\chi(\alpha_i, \mathrm{fscore}_k)}) + N_i$$

$$\mathrm{mfscore}_{\ k} = \mathrm{fscore}_{\ k} - \sqrt{nsumcnt}\Gamma\alpha_i$$

$$\mathrm{BayesScore} = \left(\mathrm{mfscore}_k - |N|^* \Gamma \log\, (\alpha_{i)}\right) + \mathrm{GaussianDistri}\ (|N|^* \alpha_i) + ME(F) \tag{9}$$

Deep feature entropy:

$$\mathrm{Pr_i} = -\mathrm{Pr(F_i)}\,.\,\mathrm{log}\,(\mathrm{Pr(F_i)})$$

$$\mathrm{E(F)} = \mathrm{Pr(F_i/C_m)}\,.\,\sum_{\mathrm{i}} \mathrm{Pr_i}$$

$$ME(F) = Pr + \frac{\chi(F).log\ (E(F))}{\sqrt{E(F)*log(2*\chi(F))}} \tag{10}$$

4. EXPERIMENTAL RESULTS

Experimental results are simulated in windows operating system with 12 GB RAM and 10 GB UCF-101 training data. In this work, UCF-101 action dataset is used as training videos for feature extraction and action prediction. UCF-101 is the larger activity recognition dataset. It has 101 action categories with a total of 13,320 videos. UCF data has different variations of classes with large number of feature space for anomaly detection process. The inter and intra variation of the UCF data are difficult to predict the anomaly using the static feature space. Results are simulated in PyCharm IDE environment with Anaconda installer. Anaconda installer is used to load required packages in PyCharm environment.

In this work, different types of actions are predicted using the pre-trained deep learning C3D framework. This dataset is a publicly available data set to evaluate the anomaly detection and location for crowded scenes. This is used for the performance evaluation of the proposed system in University of California, San Diego (UCSD). A fixed camera with a height and a resolution of 238×158, with 10 fps and a view over pedestrian footpaths, has provided the set of data. The multitude varied from small to crowd in the walkways. The video only contained pedestrians in the normal setting. There are abnormal events: circulation of non-foot passes and anomalous moving patterns in the footpaths, both of which are responsible. The data was divided into two sub-sets, each of which matched the scene. The videos from each scene were divided into several videos with approximately 200 frames. Videos had been divided into two sub-sets, namely Ped 1 and Ped 2. Videos from each scene were divided into different clips with approximately 200 images each. In addition, the generated pixel-level binary masks, which identified anomalous regions, were generated with a subset of 10 Peds 1 clips and twelve Peds 2 clips. The aim is to enable the performance assessment to identify anomalies for the algorithms. In this model, different anomaly video datasets are taken to check the detection rate of the proposed model with different background complex scenes.

Figure 6 illustrate the performance of the present model on the explosion training video datasets for feature extraction. As shown in Figure 6, the number of motion features detected in the proposed Kalman based PCA is lower than the traditional feature selection models. These filtered features are used as positive bag features for C3D network. The proposed model is based on the essential positive and negative bag feature sets to improve the classification rate on the training video datasets.

Figure 7 illustrates the performance of the present model on the road accidents training video datasets for feature extraction. The number of motion features detected in the proposed Kalman based PCA is lower than the traditional feature selection models. These filtered features are used as positive bag features for C3D network. The proposed model is based on the essential positive and negative bag feature sets to improve the classification rate on the training video datasets.

Figure 8 illustrates the performance of the present model on the shooting training video datasets for feature extraction. the number of motion features detected in the proposed Kalman based PCA is lower than the traditional feature selection proposed model is based on the essential positive and negative bag feature sets to improve the classification rate on the training video datasets. Figure 9 illustrates the confusion matrix of the proposed deep learning model to the conventional models for 22 classes video categories in UCF -101 dataset. Proposed model has high computational anomaly detection with high true positive rate on the UCF training video datasets.

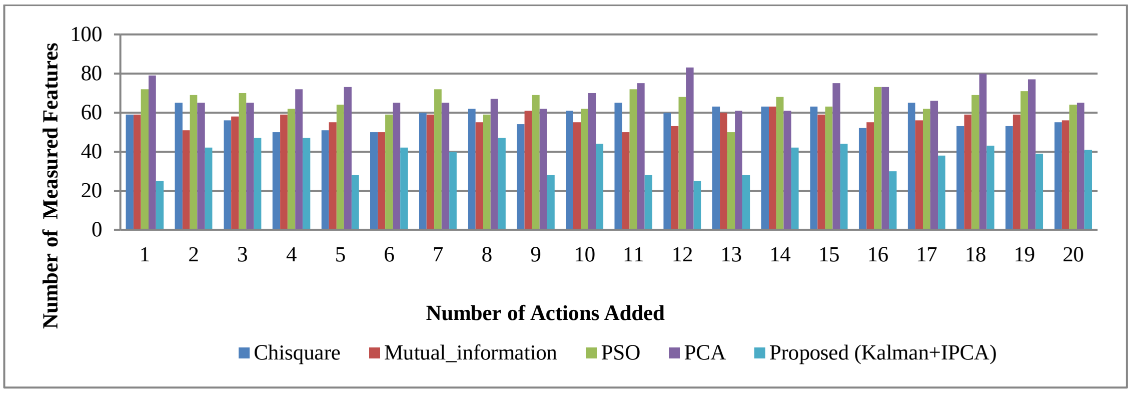


Figure 6. Explosion videos feature selection analysis using Kalman-IPCA

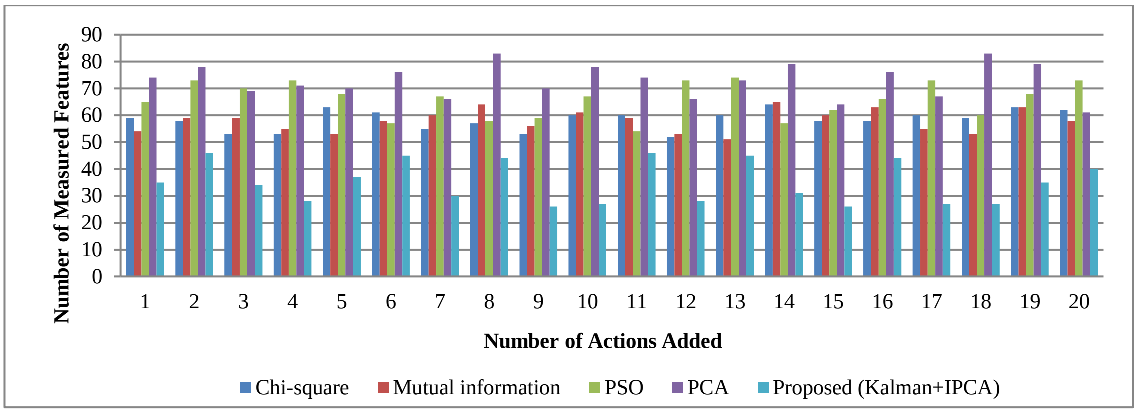


Figure 7. Road accidents videos feature selection analysis using Kalman-IPCA

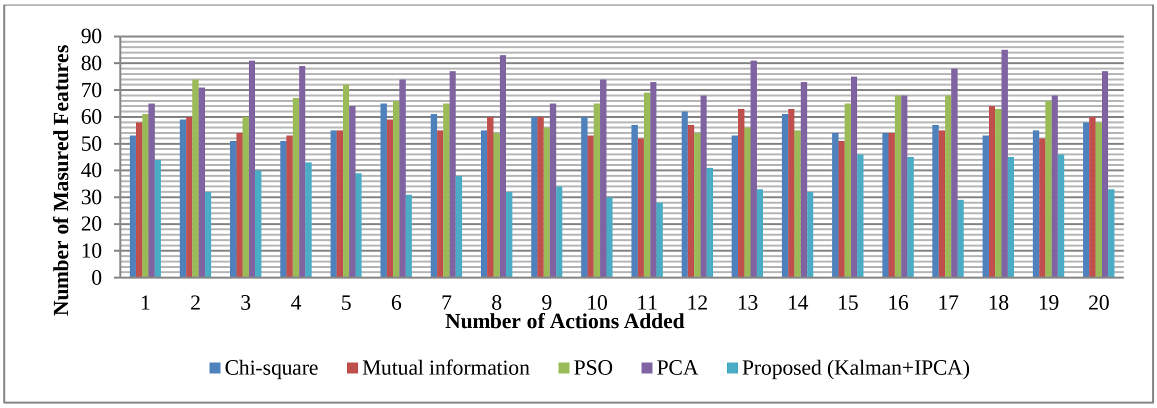
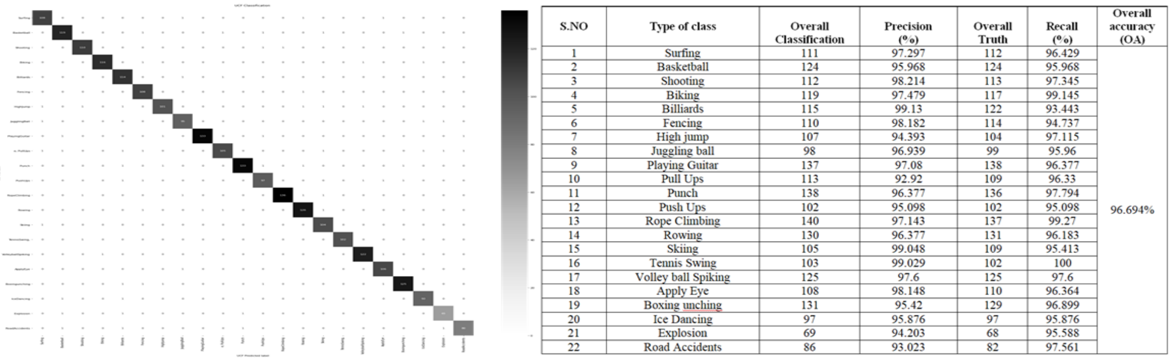


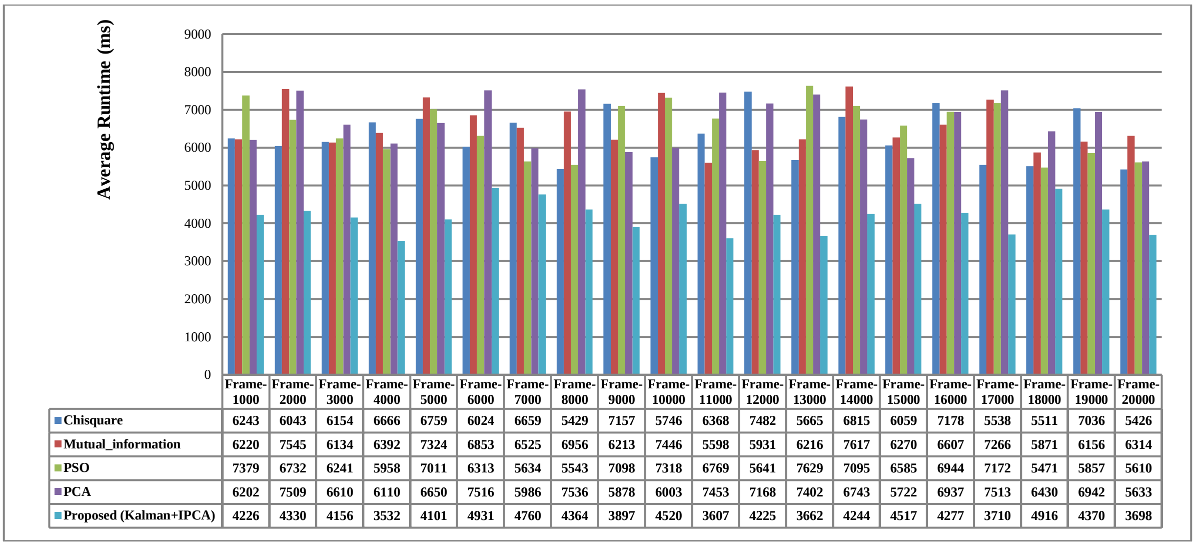
Figure 8. Shooting training videos feature selection analysis using Kalman-IPCA

Figure 10, describe the computational anomaly detection runtime (ms) on all the training UCF datasets. Here, the computational time of each video category is considered to measure the overall average runtime of all training video sets. Also, from Figure 10, it is noted that the proposed model has low computational runtime (ms) than the conventional models on large video datasets. Figure 11 shows the comparison between the present CNN and the boosting classifier to the conventional classifiers using medium accuracy. The positive effect of the action data is 10% optimized compared to the standard action classification model. The current CNN boosting classification is shown to have better average accuracy than conventional CNN classifications as shown in Table 1. Figure 12, represents the performance of the proposed AUC curve using the proposed filter based anomaly prediction model on UCF database. From the curve, it is noted that the present anomaly detection model has better AUC rate than the conventional models on large UCF database.



S.NO	Type of class	Overall Classification	Precision (%)	Overall Truth	Recall (%)	Overall accuracy (OA)
1	Surfing	111	97.297	112	96.429	96.694%
2	Basketball	124	95.968	124	95.968	
3	Shooting	112	98.214	113	97.345	
4	Biking	119	97.479	117	99.145	
5	Billiards	115	99.13	122	93.443	
6	Fencing	110	98.182	114	94.737	
7	High jump	107	94.393	104	97.115	
8	Juggling ball	98	96.939	99	95.96	
9	Playing Guitar	137	97.08	138	96.377	
10	Pull Ups	113	92.92	109	96.33	
11	Punch	138	96.377	136	97.794	
12	Push Ups	102	95.098	102	95.098	
13	Rope Climbing	140	97.143	137	99.27	
14	Rowing	130	96.377	131	96.183	
15	Skiing	105	99.048	109	95.413	
16	Tennis Swing	103	99.029	102	100	
17	Volley ball Spiking	125	97.6	125	97.6	
18	Apply Eye	108	98.148	110	96.364	
19	Boxing unching	131	95.42	129	96.899	
20	Ice Dancing	97	95.876	97	95.876	
21	Explosion	69	94.203	68	95.588	
22	Road Accidents	86	93.023	82	97.561	

Figure 9. Confusion matrix of the proposed model for 22 UCF anomaly categories



	Frame-1000	Frame-2000	Frame-3000	Frame-4000	Frame-5000	Frame-6000	Frame-7000	Frame-8000	Frame-9000	Frame-10000	Frame-11000	Frame-12000	Frame-13000	Frame-14000	Frame-15000	Frame-16000	Frame-17000	Frame-18000	Frame-19000	Frame-20000
Chisquare	6243	6043	6154	6666	6759	6024	6659	5429	7157	5746	6368	7482	5665	6815	6059	7178	5538	5511	7036	5426
Mutual_information	6220	7545	6134	6392	7324	6853	6525	6956	6213	7446	5598	5931	6216	7617	6270	6607	7266	5871	6156	6314
PSO	7379	6732	6241	5958	7011	6313	5634	5543	7098	7318	6769	5641	7629	7095	6585	6944	7172	5471	5857	5610
PCA	6202	7509	6610	6110	6650	7516	5986	7536	5878	6003	7453	7168	7402	6743	5722	6937	7513	6430	6942	5633
Proposed (Kalman+IPCA)	4226	4330	4156	3532	4101	4931	4760	4364	3897	4520	3607	4225	3662	4244	4517	4277	3710	4916	4370	3698

Figure 10. Average runtime for feature selection process of 22 classes in UCF-101 dataset

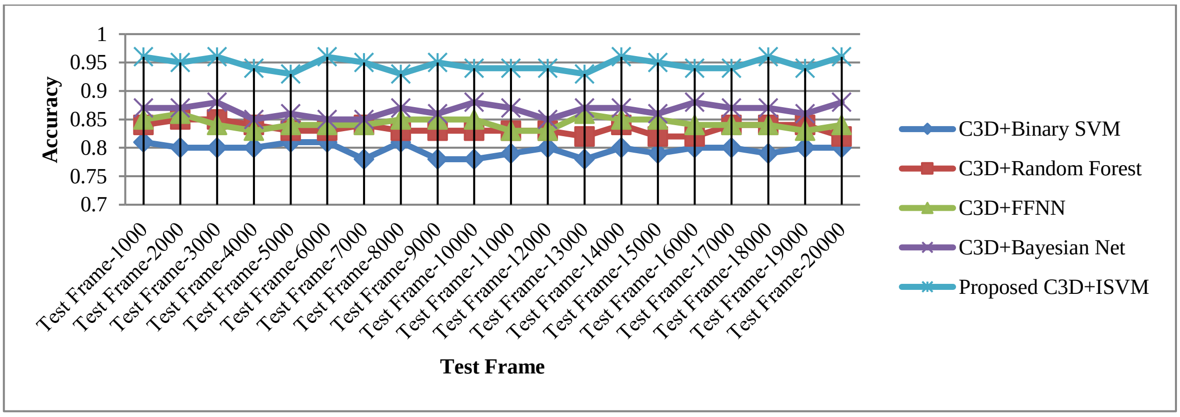


Figure 11. Comparison of present CNN model with different action prediction classifiers

4.1. Results interpretation

This frame work has a hybrid feature extraction-based anomaly detection is designed and implemented on the UCF dataset. In the experimental results, different types of action features are extraction based on the motion features for the anomaly detection process. It is observed from the results that the proposed feature extraction model has better traditional feature extraction measures on different contextual motion vectors. Proposed feature extraction model achieves more than 2% efficiency in terms of features evaluation and contextual relationships using different anomaly classes. A hybrid model for anomaly detection is proposed to improve the error rate and the accuracy on different anomaly classes are implemented in this framework. A hybrid boosting classifier is integrated to the C3D framework for better anomaly detection process. Experimental results proved that the proposed boosting based C3D framework has nearly 1% improvement over the traditional anomaly detection models with different classes.

Table 1. Comparison of C3D CNN Model with different action prediction classifiers

Test frame-ID	C3D+Binary SVM	C3D+Random forest	C3D+FFNN	C3D+Bayesian net	Proposed C3D+ISVM
Test Frame-1000	0.81	0.84	0.85	0.87	0.96
Test Frame-2000	0.8	0.85	0.86	0.87	0.95
Test Frame-3000	0.8	0.85	0.84	0.88	0.96
Test Frame-4000	0.8	0.84	0.83	0.85	0.94
Test Frame-5000	0.81	0.83	0.84	0.86	0.93
Test Frame-6000	0.81	0.83	0.84	0.85	0.96
Test Frame-7000	0.78	0.84	0.84	0.85	0.95
Test Frame-8000	0.81	0.83	0.85	0.87	0.93
Test Frame-9000	0.78	0.83	0.85	0.86	0.95
Test Frame-10000	0.78	0.83	0.85	0.88	0.94
Test Frame-11000	0.79	0.83	0.83	0.87	0.94
Test Frame-12000	0.8	0.83	0.83	0.85	0.94
Test Frame-13000	0.78	0.82	0.86	0.87	0.93
Test Frame-14000	0.8	0.84	0.85	0.87	0.96
Test Frame-15000	0.79	0.82	0.85	0.86	0.95
Test Frame-16000	0.8	0.82	0.84	0.88	0.94
Test Frame-17000	0.8	0.84	0.84	0.87	0.94
Test Frame-18000	0.79	0.84	0.84	0.87	0.96
Test Frame-19000	0.8	0.84	0.83	0.86	0.94
Test Frame-20000	0.8	0.82	0.84	0.88	0.96

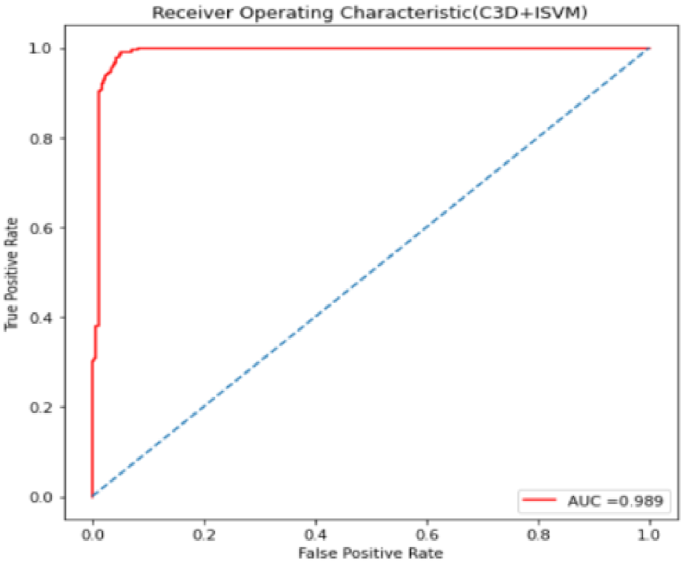


Figure 12. Performance of proposed model for AUC prediction

5. CONCLUSION

Action detection is one of the main applications of surveillance detection systems for real-time scenarios. Traditional models for action detection are dependent on training or dictionary feature sets and pre-trained classification models on large datasets. The C3D and non-linear support vector machine (SVM) classification are used to enhance the anomaly detection in large video databases for real-time applications. Experimental results shown that the current context has high computational performance compared to traditional deep learning frameworks for anomaly detection and these results have demonstrated that the current framework has nearly 3% accuracy and 1% runtime on large action data sets compared to the existing models. In future this work is extended by optimizing the C3D framework with optimized feature extraction based segmentation process on large anomaly databases.

REFERENCES

[1] A. Ullah, K. Muhammad, I. U. Haq, and S. W. Baik, "Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments," *Future Generation Computer Systems*, vol. 96, pp. 386–397, Jul. 2019, doi: 10.1016/j.future.2019.01.029.

[2] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2014, pp. 1725–1732.

[3] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998, doi: 10.1162/089976698300017467.

[4] B. M. Nair, "Deep dive into convolutional 3D features for action and activity recognition (C3D)," *Medium.com blog*, 2018. [Online]. Available: https://medium.com/@nair.binum/quick-overview-of-convolutional-3d-features-for-action-and-activity-recognition-c3d-138f96d58d8f.

[5] B. Zhang, Z. Li, A. Perina, A. D. Bue, V. Murino, and J. Liu, "Adaptive local movement modeling for robust object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 7, pp. 1515–1526, Jul. 2017, doi: 10.1109/TCSVT.2016.2540978.

[6] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal residual networks for video action recognition," *Advances in Neural Information Processing Systems*, vol. 0, pp. 3476–3484, 2016.

[7] D. Draper, "Bayesian modeling, inference and prediction," Thesis, University of California, 2005.

[8] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks," *arXiv preprints*, Oct. 2015, [Online].

[9] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "*C3D: generic features for video Analysis*" *arXiv preprints*, Dec. 2014, [Online]. Available: http://arxiv.org/abs/1412.0767.

[10] D. D. Dawn and S. H. Shaikh, "A comprehensive survey of human action recognition with spatio-temporal interest point (STIP) detector," *The Visual Computer*, vol. 32, no. 3, pp. 289–306, Mar. 2016, doi: 10.1007/s00371-015-1066-2.

[11] G. T. Papadopoulos and P. Daras, "Human action recognition using 3d reconstruction data," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 8, pp. 1807–1823, Aug. 2018, doi: 10.1109/TCSVT.2016.2643161.

[12] H. Wang, A. Kläser, C. Schmid, and C. L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, May 2013, doi: 10.1007/s11263-012-0594-8.

[13] "UCSD Anomaly Detection Dataset," 2013, http://www.svcl.ucsd.edu/projects/anomaly/dataset.html visited on March 2020.

[14] H. Liu, H. Tang, W. Xiao, Z. Guo, L. Tian, and Y. Gao, "Sequential bag-of-words model for human action classification," *CAAI Transactions on Intelligence Technology*, vol. 1, no. 2, pp. 125–136, Apr. 2016, doi: 10.1016/j.trit.2016.10.001.

[15] H. Xu, Q. Tian, Z. Wang, and J. Wu, "A joint evaluation of different dimensionality reduction techniques, fusion and learning methods for action recognition," *Neurocomputing*, vol. 214, pp. 329–339, Nov. 2016, doi: 10.1016/j.neucom.2016.06.017.

[16] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *2013 IEEE International Conference on Computer Visio*n, Dec. 2013, pp. 3551–3558, doi: 10.1109/ICCV.2013.441.

[17] H. A. Abdul-Azim and E. E. Hemayed, "Human action recognition using trajectory-based representation," *Egyptian Informatics Journal*, vol. 16, no. 2, pp. 187–198, Jul. 2015, doi: 10.1016/j.eij.2015.05.002.

[18] F. Chen, N. Sang, X. Kuang, H. Gan, and C. Gao, "Action recognition through discovering distinctive action parts," *Journal of the Optical Society of America*, vol. 32, no. 2, pp. 173-185, 2015, doi: 10.1364/JOSAA.32.000173.

[19] H. Wei, Y. Xiao, R. Li, and X. Liu, "Crowd abnormal detection using two-stream fully convolutional neural networks," in *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, Feb. 2018, vol. 2018-Janua, pp. 332–336, doi: 10.1109/ICMTMA.2018.00087.

[20] H. Zhou, X. Wang, and Y. Zhang, "Feature selection based on weighted conditional mutual information," *Applied Computing and Informatics*, Aug. 2020, doi: 10.1016/j.aci.2019.12.003.

[21] H. Dai, "Research on SVM improved algorithm for large data classification," in *2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA)*, Mar. 2018, pp. 181–185, doi: 10.1109/ICBDA.2018.8367673.

[22] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, "Dynamic Image Networks for Action Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Jun. 2016, vol. 2016-Decem, pp. 3034–3042, doi: 10.1109/CVPR.2016.331.

[23] I. M. EL-Henawy, H. A. Mahmoud, and K. Ahmed, "Sequential-based action recognition technique based on homography of interested SIFT keypoints," in *2016 11th International Conference on Computer Engineering & Systems (ICCES)*, Dec. 2016, pp. 161–166, doi: 10.1109/ICCES.2016.7821993.

[24] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2–3, pp. 107–123, Sep. 2005, doi: 10.1007/s11263-005-1838-7.

[25] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2008, pp. 1–8, doi: 10.1109/CVPR.2008.4587756.

[26] J. M. Carmona and J. Climent, "Human action recognition by means of subtensor projections and dense trajectories," *Pattern Recognition*, vol. 81, pp. 443–455, Sep. 2018, doi: 10.1016/j.patcog.2018.04.015.

[27] J. Donahue *et al*., "Long-term recurrent convolutional networks for visual recognition and description," *in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, vol. 07-12-June, pp. 2625–2634, doi: 10.1109/CVPR.2015.7298878.

[28] K. Shimazaki and T. Nagao, "Scene classification using color and structure-based features," in *2013 IEEE 6th International Workshop on Computational Intelligence and Applications, IWCIA 2013 - Proceedings*, Jul. 2013, pp. 211–216, doi: 10.1109/IWCIA.2013.6624817.

[29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprints*, Sep. 2015, [Online]. Available: http://arxiv.org/abs/1409.1556.

[30] K. Kang and X. Wang, "Fully convolutional neural networks for crowd segmentation," *arXiv preprints*, 2014, [Online]. Available: http://arxiv.org/abs/1411.4464.

[31] K. Singh, G. Gupta, L. Vig, G. Shroff, and P. Agarwal, "Deep convolutional neural networks for pairwise causality," *arXiv preprints*, Jan. 2017, [Online]. Available: http://arxiv.org/abs/1701.00597.

[32] K. Liu, W. Liu, C. Gan, M. Tan, and H. Ma, "T-C3D: Temporal convolutional 3D network for real-time action recognition," *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, vol. 32, no. 1, pp. 7138–7145, Apr. 2018, doi: 10.1609/aaai.v32i1.12333.

[33] L. Sun, K. Jia, D. Y. Yeung, and B. E. Shi, "Human action recognition using factorized spatio-temporal convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision, Dec. 2015, vol. 2015 International Conference on Computer Vision, ICCV 2015*, pp. 4597–4605, doi: 10.1109/ICCV.2015.522.

[34] M. Subedar, R. Krishnan, P. L. Meyer, O. Tickoo, and J. Huang, "Uncertainty aware audiovisual activity recognition using deep Bayesian variational inference," *arXiv preprints*, Nov. 2018, [Online]. Available: http://arxiv.org/abs/1811.10811.

[35] M. Latah, "Human action recognition using support vector machines and 3D convolutional neural networks," *International Journal of Advances in Intelligent Informatics*, vol. 3, no. 1, p. 47, Mar. 2017, doi: 10.26555/ijain.v3i1.89.

[36] M. J. Roshtkhari and M. D. Levine, "An on-line, real-time learning method for detecting anomalies in videos using spatio-temporal compositions," *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1436–1452, Oct. 2013, doi: 10.1016/j.cviu.2013.06.007.

[37] J. M. Chaquet, E. J. Carmona, and A. Fernández-Caballero, "A survey of video datasets for human action and activity recognition," *Computer Vision and Image Understanding*, vol. 117, no. 6, pp. 633–659, Jun. 2013, doi: 10.1016/j.cviu.2013.01.013.

[38] M. Ribeiro, A. E. Lazzaretti, and H. S. Lopes, "A study of deep convolutional auto-encoders for anomaly detection in videos," *Pattern Recognition Letters*, vol. 105, pp. 13–22, Apr. 2018, doi: 10.1016/j.patrec.2017.07.016.

[39] M. Koperski, "Human action recognition in videos with local representation," Thesis, de l'Université Cote d'Azur, 2017.

[40] N. Jaouedi, N. Boujnah, and M. S. Bouhlel, "Human action recognition using wavelets of derived beta distributions," in *2017 18th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, Dec. 2017, vol. 2017-Decem, pp. 516–520, doi: 10.1109/PDCAT.2017.00088.

BIOGRAPHIES OF AUTHORS

Vishnu Priya Thotakura received M.Tech. Degree in Electronics and Communication Engineering from Acharya Nagarjuna University, Guntur, India in 2008. She is currently pursuing Ph.D. degree with VIT–AP University, Amaravati, India. Her research interest includes image processing, surveillance video analytics, video compression, image processing, energy-efficient scheme for wireless sensor network. She can be contacted at email: vishnupriya000@gmail.com.

Purnachand Nalluri received his M.Tech degree from VIT University, Vellore, India, and Ph.D. degree from University of Aveiro, Aveiro, Portugal. He is currently working as Associate Professor at VIT-AP University, India. His areas of research include image processing, video processing, pattern recognition, FPGA, and ASIC architectures for video processing applications. He can be contacted at email: chanduinece@gmail.com.