



Detection of Moving Object in Dynamic Visual Sequences Based on Partial Least Squares Classifier

Shyamala Balakumar¹ · Selvaperumal Sundaramoorthy² · Ramasubramanian Bhoopalan³ · G. Prabhakar⁴

Received: 6 April 2019 / Accepted: 10 June 2019 / Published online: 1 July 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

Detection of moving object from a visual sequence plays a vital role for the tracking of object. The main objective of this proposed work is to detect and classify the various video sequences with the help of different classification algorithms. The input video sequences from the publicly available datasets are collected and the individual frames are extracted. These frames are pre-processed and then applied to the novel background subtraction process. Important features based on the Local Binary Pattern (LBP) and grey level co-efficient are extracted. Finally these features are classified by three different classifiers like SVM, PLS, and PNN. The performance of these different classifiers are evaluated and compared. It is found that PLS classifier produces more classification accuracy but with more computation time.

Keywords Adaptive histogram equalization · Local binary pattern · PLS · PNN · SVM

Introduction

The detection and classification of moving object in a video sequence is important for tracking of object, activity recognition and video surveillance. The aim of any motion detection technique is to separate the foreground region of the moving object in a video from the background region. The conventional method proposed in [1] used optical flow technique for the detection of moving object in real time environment. The method reported a good accuracy with a cost of high complexity and time. These limitations can be overcome with the help of background

subtraction and temporal differencing approaches. Background subtraction takes the input video sequence and detects the moving objects in a frame by finding the difference of current pixel of the frame with the pixel of the background reference frame [2]. Usually the first frame is selected as the first reference frame and it is then updated periodically.

Whereas, the temporal differencing find the difference of pixel features in a consecutive frame obtained from the video.

Optical flow method takes the sequence of images assigns to every pixel a 2D velocity vector. Based on the attributes of these velocity vectors, the moving objects are segmented and the edge shapes of these objects are detected.

This proposed approach uses background subtraction technique for detecting the moving objects and compares the performance of different classifiers. This developed system will answers to many problems that develops while applying background subtraction techniques. Some of these problems are sudden or gradual variation in the lighting condition, movement of multiple objects in the scene, noise present in the frame because of poor and low quality visual source and the shadow regions which are projected by the foreground objects as moving objects.

An example for background reference frame and a frame with a moving objects in a video are illustrated in Fig. 1a and b.

This article is part of the Topical Collection on *Image & Signal Processing*

✉ Shyamala Balakumar
shyamalabalakumar@yahoo.com

¹ Department of ECE, Mohamed Sathak Engineering College, Kilakarai, Ramanathapuram, India

² Department of EEE, Syed Ammal Engineering College, Lanthai, Ramanathapuram, TamilNadu, India

³ Department of ECE, SRM TRP Engineering College, Irungalur, Truchirappalli, TamilNadu, India

⁴ Department of EEE, V.S.B. Engineering College, Karur, Tamilnadu, India



(a)



(b)

Fig. 1 a static environmental frame b dynamic environmental frame

The main objective of this work is to detect the moving object in the video and classify them using different classification algorithms. The organization of the paper is as follows. The overview of state of art is classified in section 2. The proposed approach of object detection and classification is discussed in section 3, section 4 reports. The experimental results and validation. Finally the solutions are summarized and the conclusion is given in section 5.

Overview of State of Art

Several studies have investigated the automatic detection of moving objects in a video sequence [3–10]. Y Yang et al. in [3] applied spatio-temporal modelling for

segmenting and detecting the presence of moving objects in video sequences. Temporal image features were extracted from the background frame obtained from the video. In order to segment the moving objects, dynamic background algorithm was applied and achieved a recall rate of 72%, false positive rate of 27.73% and false negative rate of 27.04%.

A methodology for detecting the moving object using codebook algorithm was proposed by S Li et al. in [4]. Initially, codebook algorithm was employed for segmenting the foreground and background. Three frame difference was used in detecting the foreground frame. For eliminating the cavities introduced by the three frame differences, the author applied log edge detection and component filling in order to optimize the foreground. Finally, logical ‘AND’ operation was applied to relate the foreground object obtained by the improved three frame difference algorithm with the codebook algorithm.

A system for person segmentation, tracking and interpretation was developed by Wren et al. in [5]. The author named the system as pfinder and modelled each pixel of the background with a simple Gaussian distribution. But the estimation of the Gaussian parameter using Expectation Maximization (EM) for each pixel is computationally complex.

Bilodeau et al. in [6] proposed an efficient approach for detecting the moving objects. They applied modified local binary similarity patterns for segmenting the background from the input video sequences. They also tested the effectiveness of their proposed method on various real time video sequences.

Elgammal et al. in [7] introduced a non-parametric background modelling. This approach estimates the probability of observing a pixel grey level based on a sample of grey levels for each pixel. The author employed colour information to suppress the shadows of the target object.

A training free method for the detection of moving object in video sequence is presented by Zhang et al. in [8]. For each frames of the sequence, dense optical flow between itself and its previous frame is measured. A novel clustering method is applied for each region whose optical flow is high. This helps to segment the different objects in motion. This approach achieved a recall rate of 87.2% with a precision of 93.5%.

Wu et al. in [9] proposed a method to use ratio images as the fundamental step for the motion detection. The effects of poor illumination are smoothed out. Usually the problem in the selection of method is related to the difference image. In this method, it is shifted to ratio image. This problem was addressed automatically by the author based on histogram technique.

A Non-parametric kernel density estimation for the detection and tracking of moving object was proposed by Ianasi

et al. in [10]. The author developed a fast and robust algorithm by employing multiresolution and recursive density estimation with mean shift based tracking.

D. Kollar et al. in [11] introduced a model using kalman filter to model the background pixel based on the effects of the weather and the time on the intensity values. Whereas S Jabri et al. in [12] used colour and edge information for modelling the background and for subtraction. The author used confidence maps to fuse intermediate results. This method cannot produce good results when there is a sudden change or multiple moving objects in the scene.

Proposed Method

The purpose of this proposed approach is to detect and classify the input video sequences by different classification algorithms. The required input video sequences are collected from CIPR database. The input video sequence is first converted into frames and it is then pre-processed to improve the quality of the frame and to remove the noise. The enhanced frames are then applied to the background subtraction process. Then, the feature vectors are extracted. These features obtained from the different video sequences are trained and tested by three different classifiers. The overall schematic representation of the proposed approach is shown in Fig. 2.

Databank Used

To compare the performance of the proposed work, the input video sequences required for processing are collected from CIPR databank (<http://www.cipr.rpi.edu/resources/sequences/qcif.html>). This databank contains different video sequences which are captured under various environmental location by a high resolution camera. The input video sequence in this databank includes both indoor and outdoor environmental condition. Out of the different sequences, fountain Sequence, airport sequence, meeting room sequence and lobby sequences are chosen to validate and compare the performance of this proposed work.

Pre-Processing

The input video sequence obtained from the CIPR databank is first converted into frames. Each frame is then pre-processed in order to enhance the quality of the frame. The main purpose of pre-processing is to improve the accuracy of the work by removing the noise. Each frame obtained from the video will be in RGB form. This RGB frame is converted into HSI form and the I-Component alone is extracted for further processing. The reason for extracting the I - Component is that the noise

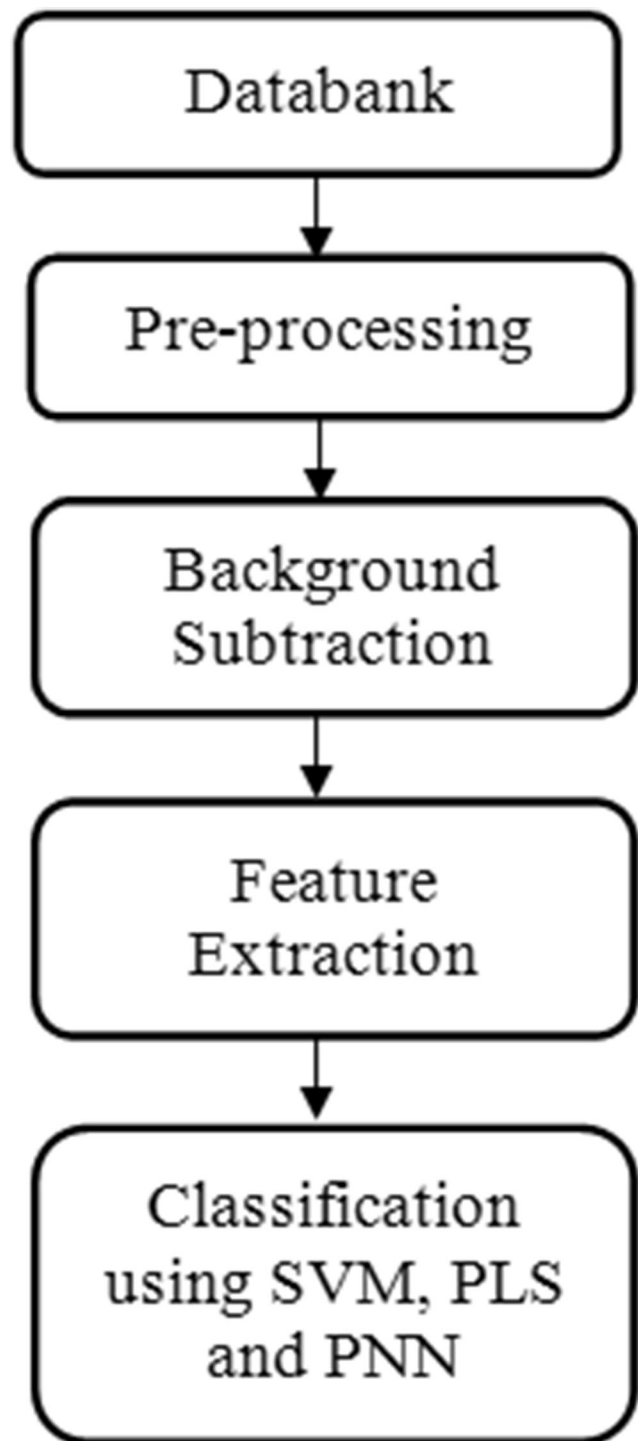
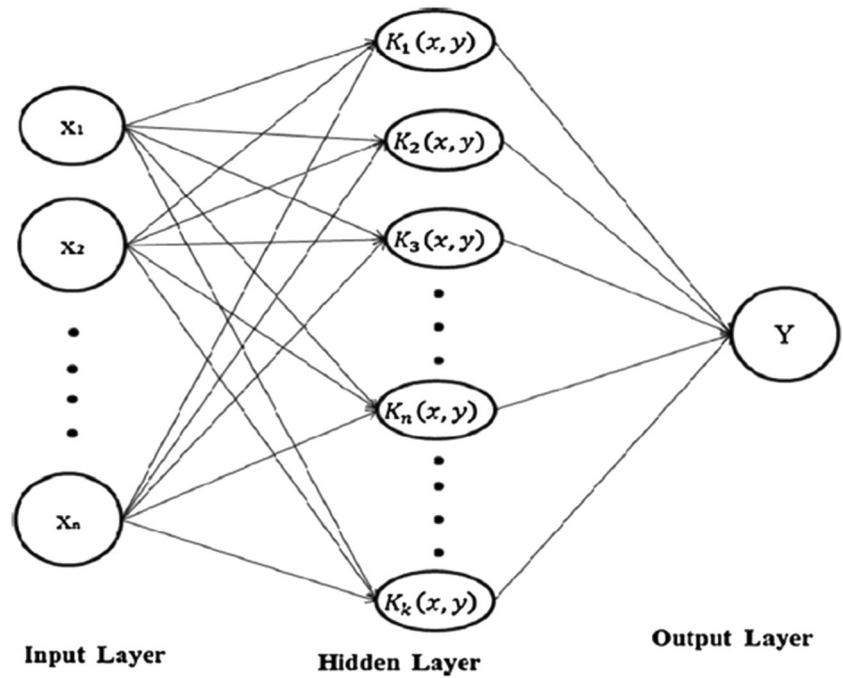


Fig. 2 Schematic diagram of the proposed work

will mostly affect only the I - Component compared to Hue and Saturation component. The extracted I - Component is then applied to a median filter for removing the noise.

In order to improve the contrast, the output of the median filter is given to the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm [13].

Fig. 3 Architecture of SVM



Background Subtraction

The next important step is the extraction of foreground object from the background. This background extraction algorithm should be able to answer a number of critical problems as discussed in section 1. In this proposed work, the extraction of the foreground region is accomplished by the combination

of temporal image analysis and a background subtraction process. Initially, a temporal analysis is performed by comparing at each time ‘t’ two consecutive frames and it provides an image IM^t . This image IM^t is used for the background subtraction process.

In the background subtraction process the image IM^t is compared with the reference background image IB^t at each

Fig. 4 Architecture of PNN classifier

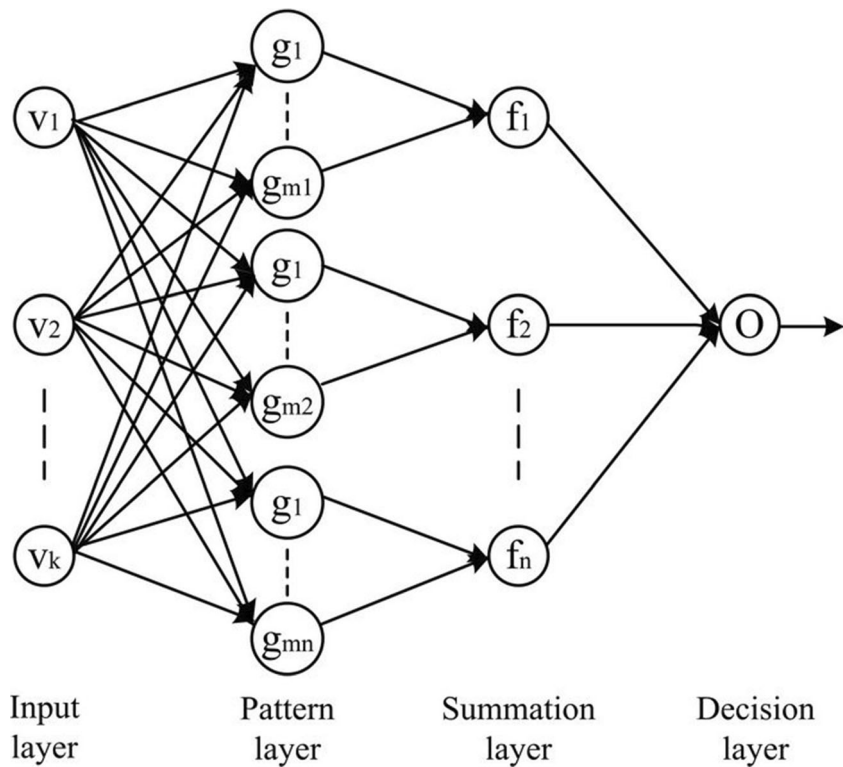


Table 1 Comparison of proposed method with other state of art method

Methodology	Accuracy (%)
Proposed Work	97.29
Patel et al. (2014)	91.50
Barnich et al. (2011)	89.60
St. Charles et al. (2014)	87.60
Our previous work	94.90

time t . To detect the foreground object, the Radiometric Similarity (RS) is calculated and it is given by

$$RS(I^t(x, y), I^{t-1}(x, y)) = \frac{m[W(I^t(x, y)W(I^{t-1}(x, y))] - m[W(I^t(x, y))]m[W(I^{t-1}(x, y))]}{(v[W(I^t(x, y))] \cdot v[W(I^{t-1}(x, y))])} \quad (1)$$

Where $m[W]$ and $v[W]$ represent the mean and variance of the intensity of the pixel present in the window W .

Feature Extraction

The objective of the feature extraction process is to represent a pixel in terms of some quantifiable information which will be useful for the classification process. In this proposed work, the following set of feature vectors were selected.

- i. Texture Feature using LBP: Features based on texture is extracted using Local Binary Pattern (LBP) algorithm.
- ii. Grey Level Features: Five different features based on the grey level of the foreground object is extracted.

Texture Feature Using LBP

With the motivation of the literature work in [14], 24 features based on texture is extracted using Local Binary Pattern (LBP). LBP is one of the powerful feature descriptor used in image processing and machine learning. As compared to other texture descriptor, the computational complexity of LBP is very less.

The primary key in this algorithm is to place a label for each pixel in the obtained foreground region. This is obtained

by calculating the number of points ‘P’ and radius ‘r’ in the local neighbourhood of the pixel. The intensity value of the centre pixel is calculated and this value is chosen as a reference. Based on this reference value, the neighbourhood pixels are threshold to form a binary pattern. Finally, the LBP labels are calculated by adding the binary pattern of every pixel and weighting scaling it with a power of two.

$$FLBP = \sum(I_p - I_c) 2^p \quad (2)$$

$$f(x) = \begin{cases} 1; & x \\ 0; & x \end{cases} \quad (3)$$

Where I_p and I_c are the intensity value of the neighbourhood pixel and centre pixel respectively and P is the number of samples on the circle of radius ‘r’.

Six statistical features like mean, Standard Deviation, median, entropy, skewness and Kurtosis are calculated from each LBP patterns. This procedure is performed for four different radius like $r = 1, 2, 3$ and 4 , thereby a total of 24 features were obtained.

Grey Level Features

The grey level of the background object provides more meaningful feature for the classification of input sequences. Considering this information into account, a set of grey level features were derived from the foreground object [15]. Let $S_{x,y}$ denotes the set of coordinates in a $w \times w$ square window centered on the described pixel (x,y) . These features are given as follows.

$$FI_1(x, y) = HI(x, y) \quad (4)$$

$$FI_2(x, y) = HI(x, y) - \min_{(s, t) \in S_{x,y}^9} \{HI(s, t)\} \quad (5)$$

$$FI_3(x, y) = \text{std}_{(s, t) \in S_{x,y}^9} \{HI(s, t)\} \quad (6)$$

$$FI_4(x, y) = HI(x, y) - \text{mean}_{(s, t) \in S_{x,y}^9} \{HI(s, t)\} \quad (7)$$

$$FI_5(x, y) = \max_{(s, t) \in S_{x,y}^9} \{HI(s, t) - HI(x, y)\} \quad (8)$$

Table 2 Experimental results obtained by the SVM Classifiers on input sequences

Video Sequence	Accuracy (in %)	Detection Rate (DR in %)	False Alarm Rate (FAR in %)	Computational Time (seconds)
Fountain Sequence	93.50	94.20	15.60	56.50
Airport Sequence	92.40	93.41	14.20	55.43
Meeting Room Sequence	95.53	96.21	11.10	35.40
Lobby Sequence	95.45	95.91	10.20	32.60
Average Value	94.22	94.93	12.77	44.98

Table 3 Experimental results obtained by the PLS Classifiers on input sequences

Video Sequence	Accuracy (in %)	Detection Rate (DR in %)	False Alarm Rate (FAR in %)	Computational Time (seconds)
Fountain Sequence	96.50	97.32	8.2	67.2
Airport Sequence	95.90	96.98	9.3	68.18
Meeting Room Sequence	98.42	98.22	7.2	45.3
Lobby Sequence	98.34	98.46	7.3	43.4
Average Value	97.29	97.74	8	56.02

Scene Classification

The extracted LBP and grey level features are combined and formed as a feature vector. To classify the input video sequences into different classes, these feature vectors are applied to the classifier algorithm. In this work, three different classifiers like SVM, PLS and PNN are chosen and their performances are compared.

SVM Classifier

The extracted feature vector from different moving objects of the inputs video sequences are applied to the SVM classifiers [16]. This classifier tries to minimize the empirical risk and prevents the overfitting problem. The architecture of this classifier is shown in Fig. 3.

This classifier consists of three different layers such as input layer, hidden layer and outer layer. The classification is performed in two different phases a) Training Phase b) Testing Phase.

In the training phase, the features extracted from the different sequences like fountain, airport, meeting room and lobby are trained by the SVM classifiers. Nearly, 60% of the extracted features are used for this process. Then, in the testing phase, the remaining features are applied and tested for the classification process.

The RBF (Radial basis function) kernel used for the process is given by

$$K_f(x, y) = \sum_{f=1}^k \beta_f e^{-\gamma f x^2} (x^f) \quad (9)$$

Table 4 Experimental results obtained by the PNN classifiers on input sequences

Video Sequence	Accuracy (in %)	Detection Rate (DR in %)	False Alarm Rate (FAR in %)	Computational Time (seconds)
Fountain Sequence	91.23	90.2	17.10	54.30
Airport Sequence	90.62	89.14	15.22	53.92
Meeting Room Sequence	93.47	92.14	13.42	38.23
Lobby Sequence	94.10	93.26	11.89	35.20
Average Value	92.35	91.18	14.40	45.41

Where γf is a kernel Parameter.

PLS Classifier

The second type of classifiers that helps to classify the input sequence is Partial Least Square (PLS) classifier.

This classifier have low bias and high variance between the different classes. In this work, a linear regression PLS classifier with adjustable threshold is employed [17–19]. The main reason for choosing this classifier is that it provides high accuracy and avoids over-fitting problem. Usually, this classifier is formulated as

$$A = B \cdot \beta + \varepsilon \quad (10)$$

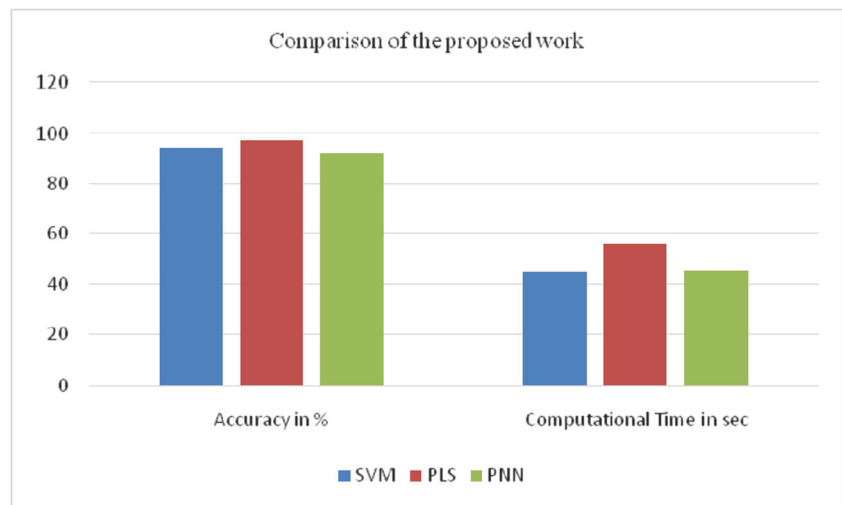
Where, A is the vector having the classification metric and B is the extracted feature vector. β is the linear regression coefficient and ε is a residual Vector.

The extracted feature vector B (in section 3.4) is applied to the PLS classifier for training and an optimum linear regression coefficient is found. This optimum value is applied to the testing phase to classify the input sequences.

PNN Classifiers

The next classifiers were experimented for classification process is probabilistic Neural Network (PNN) classifier. This classifier is one of the multilayer feed forward neural network classifier and it is derived from the Bayesian network [20]. This classifier consists of four different layers like input layer, pattern layer, summation layer and output layer and it is shown in Fig. 4.

Fig. 5 Comparison of Performance of three different classifiers



The step by step process of this PNN is given as follows.

Step 1: Feed the extracted input features into the input layer.

Step 2: Feed the trained features into next layer i.e. pattern layer. The kernel used is given by

$$Fk, i(x) = \frac{1}{(2\pi\sigma^2)} e^{-\frac{\|X-X_{ki}\|^2}{2\sigma^2}} \quad (11)$$

Where X_{ki} is the centre of the kernel and σ is the smoothing parameter. This sigma determines the speed of the kernel function.

Step 3: The next step is to compare the input features for each groups of patterns. Using this comparison, the summation layer computes the conditional probability by using a combination of the previously computed densities

$$Gk(X) = \sum_{i=1}^{Mk} Wki Fki(x) K\epsilon\{1\dots k\} \quad (12)$$

Where Wki represents the positive coefficients satisfying

$$\sum_{i=1}^{Mk} Wki = 1.$$

Step 4: At each class output node, add all the frames with similar features to those of the input.

Step 5: Calculate the maximum value of all of the added functional values at the output nodes using the following equation.

$$M(x) = \arg \max_{1 \leq k \leq k} Gk \quad (13)$$

The output layer gives the different classification results.

Results and Discussion

The performance of the proposed approach has been tested on a number of sequences obtained from both Indoor and Outdoor Environment. In particular, fountain sequence, airport sequence, meeting room sequence and lobby sequences are chosen to validate and compare the performance. Indoor Environment sequences with varying illumination are also considered.

The algorithm was developed using MATLAB (R2013a) on a Pentium IV 2.0 GHz processor. The quantitative performance of the proposed work is obtained by calculating accuracy, Detection rate (DR), False Alarm Rate (FAR) and computational time.

$$Accuracy = \frac{1}{N} \sum_i^N TP \quad (14)$$

$$Detection Rate (DR) = TP / (TP + FN) \quad (15)$$

$$False Alarm Rate (FAR) = FP / (TP + FP) \quad (16)$$

Where

TP = True Positive

FP = False Positive

FN = False Negative

True positive represents the number of detected pixels that corresponds to moving objects. False positive denotes the number of detected pixels that do not correspond to a moving object and False Negative represents the pixels of moving objects that are not detected.

In Table 1, the comparison of performance of the proposed work with other state of art work is shown. The accuracy in Table 1 for the proposed work is the average accuracy of the different video sequence obtained using PLS classifier. From

this table, it is clear that this work achieved a high accuracy as compared to other work discussed in the literature.

Table 2 shows the experimental results obtained by the SVM classifiers on different sequences.

In Table 3, the performance of the proposed work using PLS classifier is shown.

In Table 4, the experimental results of the work obtained by PNN classifier is employed.

Figure 5 shows the comparison of performance of three different classifiers. For this comparison process, accuracy and computational time of the different classifiers are chosen.

Conclusions

In this proposed work, a comparative analysis of different classifiers for the detection of moving object is presented. This work can be very useful in tracking of objects, video surveillance, activity recognition and so on. A robust algorithm for the detection of moving object in a video sequence is developed. This method takes the input video and extracts the image sequences initially. These sequences are pre-processed and the foreground object alone is segmented by novel background subtraction techniques. Different useful information based on LBP and grey level co-efficient are extracted and the different scenes are classified using three different classifiers.

The results obtained by these three different classifiers in both indoor and outdoor environment show the robustness and reliability of this approach. One of the drawbacks of this work is that it is not able to completely eliminate the shadows present in the scene. This happens mainly because of the high contrast in the background.

References

- Barnich, and Van Droogenbroeck, M., ViBe: A universal background subtraction algorithm for video sequences. *IEEE Trans. Image Process.* 20(6):1709–1724, 2011.
- Patel, C. I., Garg, S., Zaveri, T., and Banerjee, A., Top-down and bottom-up cues based moving object detection for varied background video sequences. *Advances in Multimedia* 2014:13, 2014.
- Yang, Y., Zhang, Q., Wang, P., Hu, X., and Wu, N.: 'Moving object detection for dynamic background scenes based on spatiotemporal model', *Advances in Multimedia*, 2017.
- Li, S., Liu, P. and Han, G.: 'Moving object detection based on codebook algorithm and threeframe difference', *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 2017, 10(3), pp. 23–32.
- Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A., Pfinder: Real-time tracking of the human body. *IEEE Trans. Pattern Anal. Mach. Intell.* 19(7):780–785, 1997.
- Bilodeau, G. A., Jodoin, J. P., and Saunier, N., 'Change detection in feature space using local binary similarity patterns,' in Proceedings of the international conference on Computer & Robot Vision, 10(1), pp. 106–112, , 2013.
- Elgammal, A., Harwood, D., Davis, L.: 'Non-parametric model for background subtraction' IEEE frame rate workshop, 1999, 21 September, Corfu, Greece.
- Zhang, Y., Li, G., Xie, X. and Wang, Z.: 'A new algorithm for fast and accurate moving object detection based on motion segmentation by clustering', 2017 fifteenth IAPR international conference on machine vision applications (MVA), Nagoya, pp. 444–447, , 2017.
- Wu, Q. Z., Cheng, H. Y., and Jeng, B. S., Motion detection via change-point detection for cumulative histograms of ratio images. *Pattern Recogn. Lett.* 26(5):555–563, 2005.
- Ianasi, C., Gui, V., Toma, C. I., and Pescaru, D., A fast algorithm for background tracking in video surveillance, using nonparametric kernel density estimation. *FactaUniversitatis Ser.: Elect. Energy* 18(1):127–144, 2005.
- Koller, D., Weber, J., Huang, T., Malik, J., Ogasawara, G., Rao, B., Russel, S.: 'Towards robust automatic traffic scene analysis in real-time' Proceedings of the international conference on pattern recognition, Israel, 1994.
- Jabri, S., Duric, Z., Wechsler, H., Rosenfeld, A.: ' Detection and location of people in video images using adaptive fusion of colour and edge information', ICPR 2000, 3–8 September, Barcellona, Spain, pp. 627–630.
- Ramasubramanian, B., and Selvaperumal, S., A novel efficient approach for the screening of new abnormal blood vessels in color fundus images. *Appl. Mech. Mater.* 573:808–813, 2014.
- Lu, X. et al: 'Feature extraction and fusion using deep convolutional neural networks for face detection', *Hindawi Mathematical Problems in Engineering*, 2017.
- Marín, D., Aquino, A., Gegúndez-Arias, M. E., and Bravo, J. M., A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features. *IEEE Trans. Med. Imaging* 30(1):146–158, 2011.
- Ramasubramanian, B., An efficient integrated approach for the detection of exudates and diabetic maculopathy in colour fundus images. *Advanced Computing: An International Journal (ACLJ)* 3(5): 83–91, 2012.
- Agurto, C. et al., A multiscale optimization approach to detect exudates in the macula. *IEEE Journal of Biomedical and Health Informatics* 18(4):1328–1336, 2014.
- de Jong, S., SIMPLS: An alternative approach to partial least squares regression. *ChemometricsIntell. Lab. Syst.* 18:251–263, 1993.
- Ramasubramanian, B., Selvaperumal, S. : 'Efficient approach for the automatic detection of haemorrhages in colour retinal images', *IET Image Processing*, Online ISSN 1751-966.
- Vl, G., Pavlidis, N. G., Parsopoulos, K. E. et al., New self-adaptive probabilistic neural network in bio-informatics and medical tasks. *International Journal on Artificial Intelligence Tool* 15(3):371–396, 2006.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.