

Moving Object Extraction through a Real-World Variable-Bandwidth Network using KFDA-based RBF

Ramakant Verma
Assistant Professor, CSE Dept.
BIT Meerut

Maitreyee Dutta, Ph.D
Professor & Head, CSE Dept.
NITTTR, Chandigarh

ABSTRACT

Motion detection has become one of the most important applications in traffic monitoring systems. Video communication in traffic monitoring systems may suffer network congestion or unstable bandwidth over real-world networks with definite bandwidth, which is dangerous in motion detection in video streams of variable-bit-rate. In this paper, we propose a unique Kernel Fisher's linear discriminant (KFLD)-based radial basis function (RBF) network for motion detection approach for accurate and complete detection of moving objects in video streams of both high and low bit rates. The proposed method will be accomplished through a combination of two stages: pattern generation (PG) and motion extraction (ME). In the PG stage, the variable - bit- rate video stream properties will be accommodated by this new technique, which subsequently distinguishes the moving objects within the segmented regions belonging to the moving object class by using two devised procedures that is Background Discriminant Procedure and Object Extraction Procedure during the ME stage. The accuracy result evaluations can show that the new method exhibits superior when compared to the old methods.

General Terms

Motion detection, Spatiotemporal, Gaussian Function

Keywords

PCA, RBF, KFDA, KDA

1. INTRODUCTION

Identifying moving objects from a video sequence is a fundamental and critical task in many computer-vision applications. A common approach is to perform background subtraction [1], [2], which identifies moving objects from the portion of a video frame that differs significantly from a background model. There are many challenges in developing a good background subtraction algorithm. First, it must be robust against changes in illumination. Second, it should avoid detecting non-stationary background objects such as moving leaves, rain, snow, and shadows cast by moving objects.

For example, by spatial and temporal smoothing, we can remove snow from a video. Small moving objects, such as moving leaves on a tree, can be removed by morphological processing of the frames after the identification of the moving objects.

The problem with background subtraction is to automatically update the background from the incoming video frame and it should be able to overcome the following problems:

- **Motion in the background:** Non-stationary background regions, like branches and leaves of trees, a flag waving in the

wind, or flowing water in background, can be identified as part of the background.

- **Illumination changes:** The background model should be able to tolerate, gradual changes in illumination over a period of time. Because since morning to evening the sun light change.

- **Memory:** To detect moving object the background module should not use much resource, in terms of computing power and memory.

- **Shadows:** Shadows cast by moving object should be identified as part of the background and not foreground, so to remove shadows extra logic and processor cycle is required.

- **Camouflage:** Moving object should be detected even if pixel characteristics are similar to those of the background.

- **Bootstrapping:** The background model should be able to maintain background even in the absence of training background.

2. PCA

In PCA [3], the main concept is to express the available dataset to extract the relevant information by reducing the duplicate attributes and minimize the noise. PCA didn't concern about whether this dataset represent features from one or more classes, i.e. the discrimination power was not taken into consideration in PCA. In PCA, it had a dataset matrix X with dimensions $m \times n$, where columns represent different data samples. Zero mean dataset is created by subtracting the mean from each sample of dataset matrix X , after that covariance matrix $S_x = XX^T$ is created. Now Eigen values and Eigen vectors were then computed from S_x . So the new basis vectors are those Eigen vectors with highest Eigen values. Eigen vectors having lower Eigen values are eliminated. Thus, using the new basis, algorithm can project the dataset onto a less dimensional space with more powerful data representation.

This paper presents concepts and working program to detect motion in a video sequence using Kernel Fisher Discriminant Analysis (KFDA). Prior to using KFDA for motion detection, a histogram-based [6], Sigma Difference Estimation (SDE) [4], MultiSigma Difference Estimation (MSDE) [5], Mean based [6], Median and Histogram based [7] and PCA and many more technique were used. These techniques did not provide as good results as the KFDA based system where speeds of computation and sensitivity to motion were of primary concern. Using KFDA to provide the feature descriptor of consecutive video frames along with dynamic thresholding provided very good results in detecting object larger than the video segmentation block used. The PCA algorithm presented here used two different PCA matrix generation methods: global and local; and two thresholding methods: static and dynamic. Collected data shows that the local PCA generation with dynamic thresholding may be the best combination of

speed and sensitivity. Matlab code along with video showing detected motion is provided as part of this project.

3. KFDA

Kernel fisher discriminant analysis (KFDA) [8] is a generalized version of linear discriminant analysis (LDA). LDA which is a frequently-used categorization technology is to change from input space to Eigen-space to achieve classified distinguishing. The core idea of KDA which combines kernel function and linear discriminant analysis methods is to map the samples of input space to high dimensional feature space through nonlinear transformation, and then to classify and distinguish in Eigen-space. The use of kernel function transform nonlinear distinguish in original input space into linear distinguish in high dimensional feature space. Besides KDA can maximize between-class classified difference and minimum intra-class classified difference.

The key idea of kernel fisher discriminant analysis (KFDA) algorithm enables homogeneous data close each other and heterogeneous data remote each other by projecting data to high dimensional space by kernel mapping.

Mapping of input X_i is simply noted $\phi(X_i)=\phi_i$ as total mapping input set and No. i mapping class are given by

$$\begin{aligned}\phi(X) &= \{ \phi(X_1), \phi(X_2), \dots, \phi(X_C) \} \\ \phi(X_i) &= \{ \phi(X_1^i), \phi(X_2^i), \dots, \phi(X_C^i) \}\end{aligned}$$

Mean of mapping signal class is given by $\phi(X_j)$

$$\mu_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \phi(X_j) \quad (1)$$

And global mean of total mapping samples is given by

$$\mu_0 = \frac{1}{N} \sum_{i=1}^C \sum_{j=1}^{N_i} \phi(X_j^i) \quad (2)$$

By constructing matrixes within-class scatter matrix S_W and between-class scatter matrix S_B in F . Given kernel matrix K , original data P is mapped to feature space F : $P \xrightarrow{K} X$, so new data sample can be obtained in feature space. Matrix S_W and S_B are defined as:

$$S_W = \sum_{i=1}^C \sum_{j=1}^{N_i} (\phi(X_j^i) - \mu_i) (\phi(X_j^i) - \mu_i)^T, \phi(X_j^i) \in C^i \quad (3)$$

$$S_B = \frac{1}{N} \sum_{i=1}^C N_i (\mu_i - \mu_0) (\mu_i - \mu_0)^T \quad (4)$$

Performing FDA in feature space F means maximizing between-class scatter matrix S_B and minimizing the within-class scatter matrix S_W . The following function is defined as fisher discriminate criterion:

$$J(W) = \arg \max_W \frac{|W^T S_B W|}{|W^T S_W W|} \quad (5)$$

where w is a non-zero column vector.

The aim of kernel Fisher discriminant criterions is to find optimal projection discriminant vector W . Although kernel function can solve nonlinear problem of data, but by projection, data space dimensionality becomes very huge. A new kernel function, called the Gaussian kernel, is proposed to increase the discriminating capability of the original polynomial kernel function.

4. RBF

A Radial Basis Function (RBF) [9], [10] neural network has an input layer, a hidden layer and an output layer. The neurons in the hidden layer contain Gaussian transfer functions whose outputs are inversely proportional to the distance from the center of the neuron.

1. Input layer–For each block, it consists of one neuron in input layer. Input neuron exist the range of neuron values given by the median. After that input neurons enter the values to each neuron in hidden layer.

2. Hidden Layer– A set functions given by the hidden layer constitute for input patterns on arbitrary basis. The hidden layers are also called as radialcenters and it represented by a vector. The conversion from input space to hidden space is non-linear, whereas conversion from hidden space to output space is linear.

3. Output layer– The value provided by the neurons of the hidden layer is multiplied by the weight along with output of the neurons (W_1, W_2, \dots, W_n) and passed to the summation layer that adds the value of weight and output neuron.

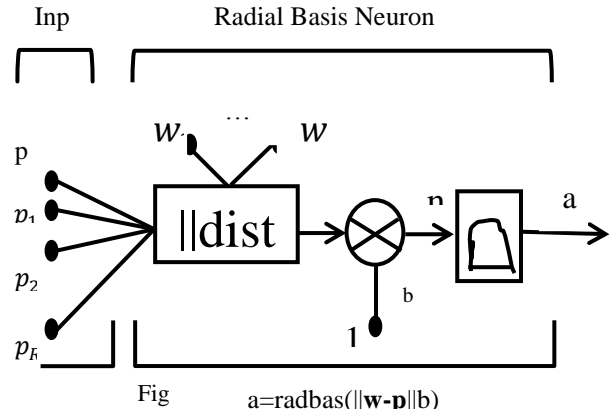


Fig a=radbas(||w-p||b)

5. PROPOSED METHODOLOGY

The key operation of our approach is performed in two proposed stages as follows.

1) The lower dimensional discriminant patterns of tolerant video streams of variable bit rate and a reliable background model can be generated effectively via the KFDA based RBF network. This occurs during the proposed a pattern generation (PG) stage so as to accommodate the properties of video streams of either low or high bit rate.

2) After the PG stage, the KFDA-based RBF network is mapped onto the incoming pixels during the proposed moving extraction (ME) stage for accurate and complete motion detection in video streams of variable bit rates.

Algorithm Step

Stage 1 Pattern Generation

1. Read the video sequence of traffic.
2. Convert rgb frame into gray scale
3. Spilt grayscale frame into 8x8 blocks
4. Apply KFDA to create lower dimensional discriminant pattern Z_{dp} of block
5. Compute KFDA by projecting blocks from each incoming frame onto that block's 3-PCA projection.
6. Compute Eigen Value matrix with window size 5 for each block.
7. Generate global threshold based on all blocks and frame.
8. Generate local dynamic threshold for each block/frame with window size 5.
9. Generate motion matrix based on local and global dynamic threshold for all blocks-frames. Those blocks which consist motion store 1 otherwise 0.

Stage 2: Motion Extractions (ME)

When construction of motion matrix that is spatiotemporal matrix is over we split incoming frame in 8x8 blocks and use spatiotemporal matrix to create sample videos showing blocks where motion was detected.

Once the unnecessary blocks of background area have been eliminated via the proposed background discriminate procedure, moving objects are accurately detected within only those blocks, which are regarded as belonging to the moving object class. This is accomplished by the proposed motion extraction procedure [11].

To this end, each pixel of each motion block is propagated to the hidden layer neurons as the input vector. The combination of the basis function and the Euclidean distance is subsequently used to generate the output of each neuron in the hidden layer, which can be expressed as follows:

$$Z_j(I_t) = \Phi(\|I_t - C_j\|) \quad (6)$$

where $j = 1$ to M , M is the number of hidden neurons, C_j is the j^{th} hidden neuron, I_t is the incoming pixel in the block belonging to the moving object class, $\Phi(\cdot)$ is the basis function, and $\|I_t - C_j\|$ represents the Euclidean distance between p_t and C_j . Note that, in our approach, the Gaussian function.

This representative function is as follows:

$$\Phi(\|I_t - C_j\|) = e^{-\frac{\|I_t - C_j\|^2}{2\sigma^2}} \quad (7)$$

where σ is the variance of motion block.

Finally, the output value can be yielded in the output layer of the RBF network and can be further separated into the binary motion detection mask. This is performed using the weighted linear combination of the hidden and output layers as follows:































$$F(I_t) = \sum_{j=1}^M W_j \cdot Z_j(I_t) \quad (8)$$

where Z_j represents the output value of the j^{th} hidden neuron and W_j is the weight between the j^{th} hidden neuron and the output layer. The decision rule can be used to obtain the binary mask of moving objects as follows:

$$Y(I_t) = \begin{cases} 1, & \text{if } F(I_t) < \delta \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

where $Y(I_t)$ represents a motion pixel when it is labelled as "1," which means that it is part of a moving object. Otherwise, the pixel is labelled as "0" to indicate a pixel belonging to the background. The experiential threshold value is represented by δ .

Fig2. Detection result of sequence traffic_1 in video stream of differing bit rates

Original Frame			
Frame	62	112	87
Bitrate	256	1200	16000
Ground Truth			
SBS			
SDE			
MSDE			
Histogram based			
Mean Based			
PCA			
KFDA			
KLDA-RBF			

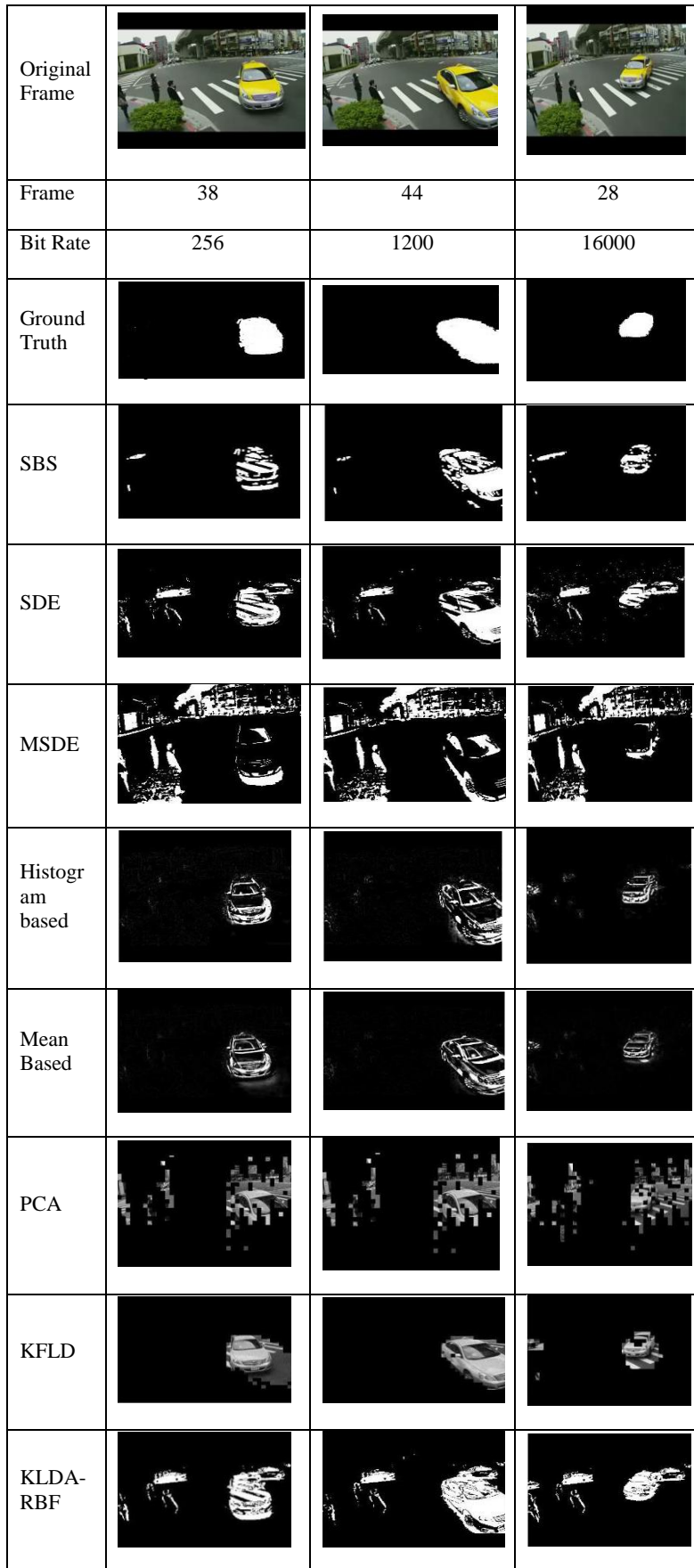


Fig 3. Detection results of traffic_2 in video streams of differing bit rates

6. PERFORMANCE EVALUATION

Table1 Classification matrix

		Classified	
Actual		Foreground	Background
	Foreground	TP	FN
	Background	FP	TN

We assessed and compared the performance of the proposed KFDA-based RBF network method to the respective performances of traditional methods through quantitative evaluation. For this evaluation, we applied *Recall*, *Precision*, *F1*, and *Similarity* metrics to video sequences of differing bit rates[11].

The *Recall* metric provides the percentage of detected true positives as compared with the total amount of true positives in the ground truth. The *Precision* metric provides the percentage of detected true positives as compared with the total amount of items detected in the detected binary object mask. They can be described as follows:

$$\text{Recall} = \frac{tp}{(tp+fn)} \quad (10)$$

$$\text{Precision} = \frac{tp}{(tp+fp)} \quad (11)$$

where tp, fn, and fp indicate total amount of true positive pixels, false negative pixels, and false positive pixels, respectively.

Sometime Recall and Precision selectively measure the incorrect association of lost true positive pixels and external true positive pixels, we cannot obtain satisfactory comparison results through the use of Recall and Precision alone. Therefore, we also applied two other accuracy metrics—F1 and Similarity—which are derived from Recall and Precision and are considered a significant measurement of accuracy. They can be expressed as

$$F1 = 2 \frac{(\text{Recall})(\text{Precision})}{(\text{Recall} + \text{Precision})} \quad (12)$$

$$\text{Similarity} = \frac{tp}{(tp + fp + fn)} \quad (13)$$

$$\text{Accuracy} = \frac{(tp+tn)}{(tp + fp + fn + tn)} \quad (14)$$

Accuracy indicates the ability of the classifier for predict unknown instances

To compare two classifiers,the points need to be compared.

The receiver operating characteristic (ROC) graph is an effective tool for visualization of classifier performance as well as comparing the performance of many classifiers.

Table 2: Performance calculation of traffic_1 video frame 62 at bit rate 256

	Frame Rate 256							
	KFDA-RBF	KFDA	PCA	Mean Based	Histogram Based	MSDE	SDE	SBS
Precision	0.191449	0.208378	0.204449	0.619619	0.646705	0.238328	0.137717	0.777882
Recall	0.523197	0.169568	0.387501	0.378798	0.270029	0.312299	0.43711	0.504003
F1	0.280322	0.18698	0.267672	0.470165	0.38098	0.270345	0.209445	0.611685
Similarity	0.217292	0.128299	0.18488	0.318502	0.245441	0.197415	0.142125	0.435698
Accuracy	0.934948	0.898073	0.870625	0.925365	0.901094	0.903281	0.841667	0.928646

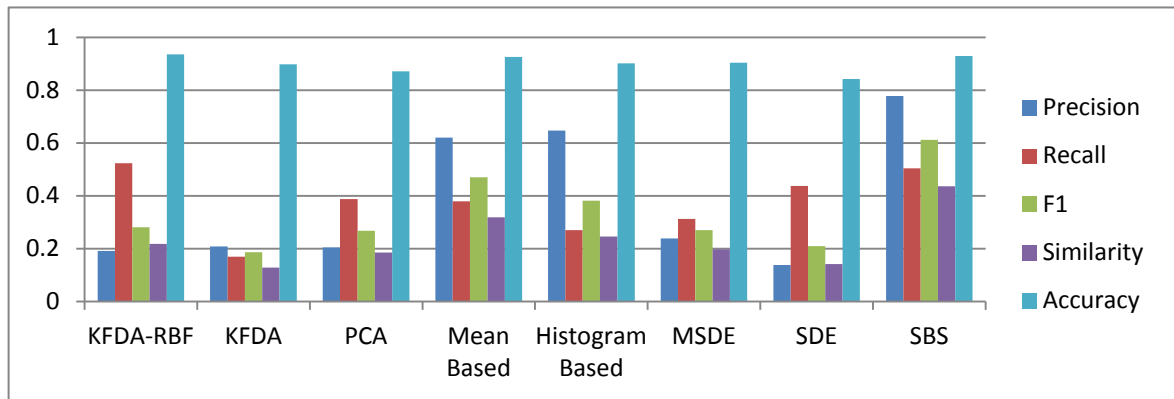


Fig 4. Graphical representation of performance of traffic_1 video frame 62 at bit rate 256

Table 3: Performance calculation of traffic_1 video frame112 at bit rate 1200

	Frame Rate 1200							
	KFDA-RBF	KFDA	PCA	Mean Based	Histogram Based	MSDE	SDE	SBS
Precision	0.60744	0.374793	0.367309	0.499516	0.41013	0.352973	0.346483	0.521581
Recall	0.598669	0.534668	0.537222	0.491827	0.333693	0.600182	0.346236	0.847929
F1	0.603022	0.440678	0.436307	0.495641	0.367984	0.444519	0.34636	0.645872
Similarity	0.454163	0.315582	0.31144	0.366957	0.275493	0.315703	0.249225	0.486327
Accuracy	0.915885	0.866328	0.864245	0.918398	0.914036	0.855638	0.898906	0.903568

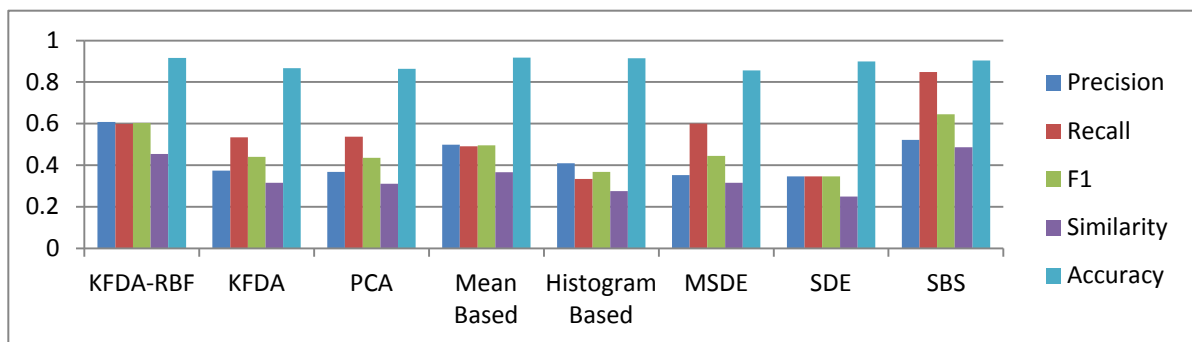


Fig 5. Graphical representation of performance of traffic_1 video frame 112 at bit rate1200

Table 4: Performance calculation of traffic_1 video frame 87 at bit rate 16000

	Frame Rate 16000							
	KFDA-RBF	KFDA	PCA	Mean Based	Histogram Based	MSDE	SDE	SBS
Precision	0.76675	0.452422	0.380642	0.741037	0.479846	0.206518	0.337599	0.536006
Recall	0.594153	0.450631	0.565343	0.431635	0.343559	0.385773	0.551644	0.731888
F1	0.669507	0.451525	0.454961	0.545519	0.400424	0.26902	0.418861	0.618816
Similarity	0.517098	0.289276	0.30966	0.374746	0.298153	0.180663	0.285539	0.45915
Accuracy	0.876198	0.78276	0.745	0.854844	0.846979	0.702396	0.76901	0.839271

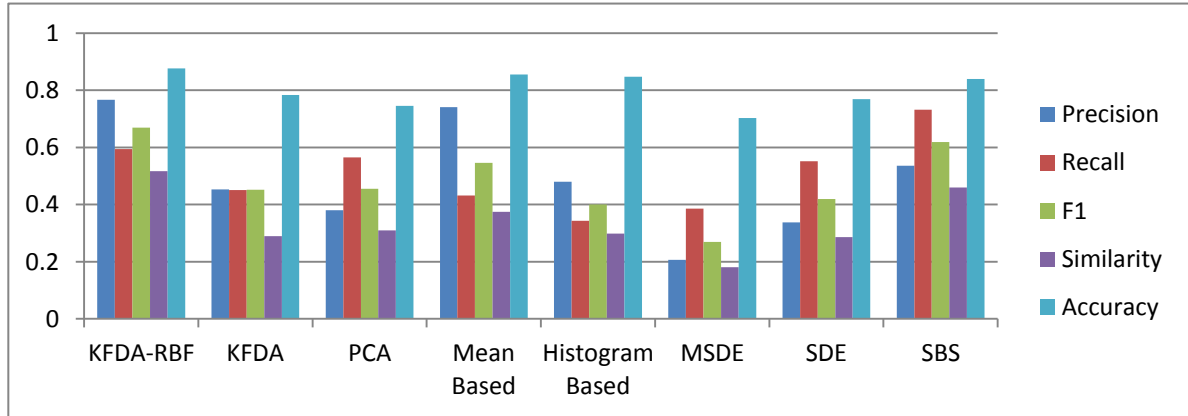


Fig 6. Graphical representation of performance of traffic_1 video frame 87 at bit rate16000

Table 5: Performance calculation of traffic_2 video frame38 at bit rate 256

	Frame Rate 256							
	KFDA-RBF	KFDA	PCA	Mean Based	Histogram Based	MSDE	SDE	SBS
Precision	0.822778	0.425565	0.592342	0.884492	0.974068	0.100741	0.838751	0.990947
Recall	0.523488	0.384715	0.5193	0.279472	0.3645	0.136598	0.423875	0.385016
F1	0.639865	0.40411	0.553421	0.424739	0.530489	0.115961	0.563153	0.554564
Similarity	0.471048	0.281654	0.405064	0.272888	0.359818	0.081682	0.392709	0.383457
Accuracy	0.921076	0.841247	0.895254	0.900143	0.905511	0.724948	0.895202	0.918011

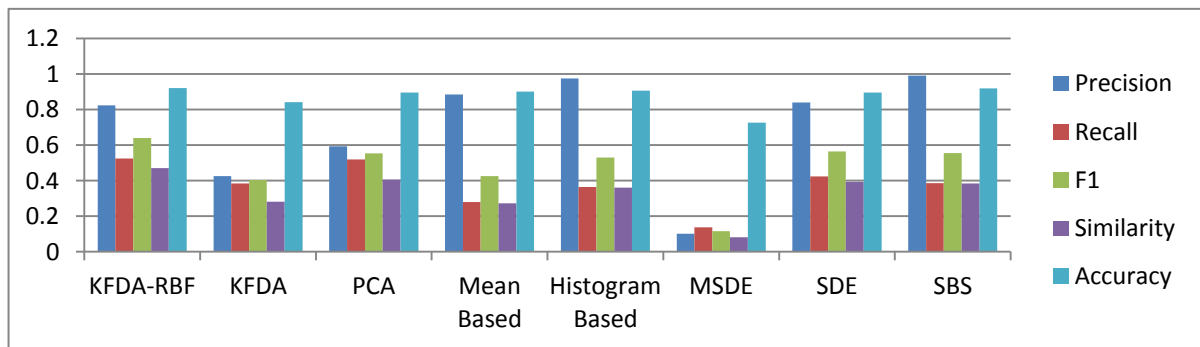


Fig 7. Graphical representation of performance of traffic_2 video frame 38 at bit rate 256

Table 6: Performance calculation of traffic_2 video frame 44 at bit rate 1200

	Frame Rate 1200							
	KFDA-RBF	KFDA	PCA	Mean Based	Histogram Based	MSDE	SDE	SBS
Precision	0.848915	0.41513	0.959478	0.742275	0.543788	0.110526	0.846048	0.629679
Recall	0.807765	0.313898	0.726843	0.457681	0.373944	0.185901	0.6072	0.592777
F1	0.827829	0.357486	0.827114	0.566229	0.443149	0.13863	0.706996	0.610671
Similarity	0.711343	0.224388	0.707259	0.430056	0.326785	0.103832	0.554877	0.48709
Accuracy	0.955977	0.82535	0.934583	0.915794	0.895234	0.697852	0.908359	0.923802

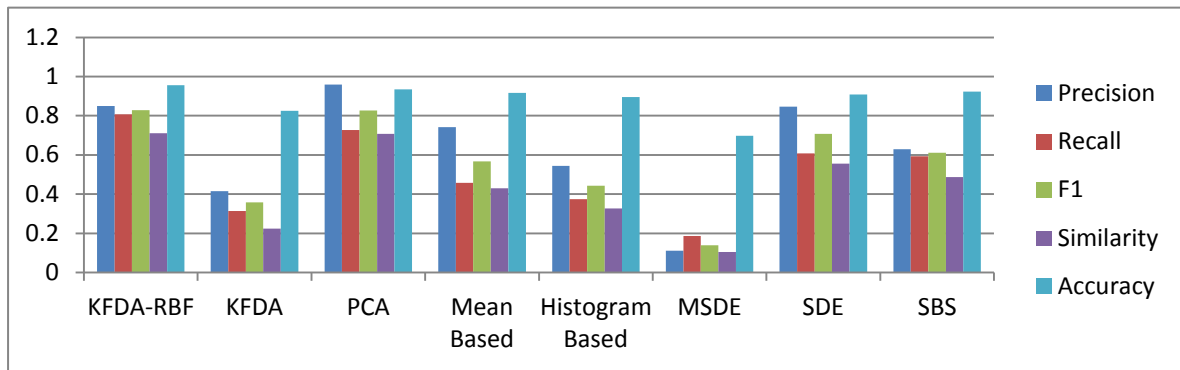


Fig 8. Graphical representation of performance of traffic_2 video frame 44 at bit rate 1200

Table 7: Performance calculation of traffic_2 video frame 28 at bit rate 16000

	Frame Rate 16000							
	KFDA-RBF	KFDA	PCA	Mean Based	Histogram Based	MSDE	SDE	SBS
Precision	0.344675	0.172225	0.444785	0.783524	0.731351	0.026071	0.383036	0.779171
Recall	0.822313	0.57484	0.62734	0.364492	0.393417	0.133235	0.627659	0.504637
F1	0.485747	0.265042	0.52052	0.497534	0.511618	0.043609	0.475744	0.61255
Similarity	0.349115	0.173009	0.367285	0.335706	0.358954	0.031307	0.366629	0.448872
Accuracy	0.96431	0.862591	0.952721	0.936901	0.95707	0.793906	0.936302	0.961719

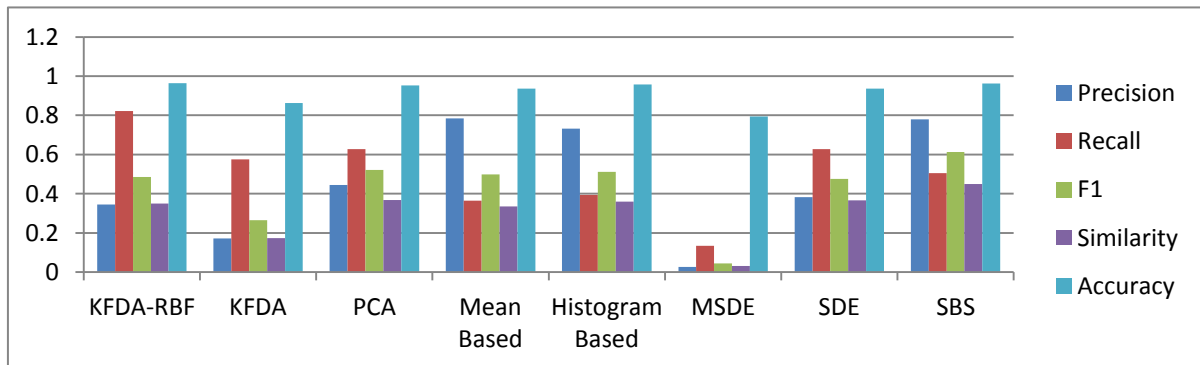


Fig 9. Graphical representation of performance of traffic_2 video frame 28 at bit rate 16000

7. CONCLUSION

The goal of KFDA based RBF technique to detect and extract moving object over variable bit rate network. This technique work well when incoming signal is noisy and classification pattern are non-linear, also when the real network bandwidth is not constant means sometimes signal stream is good and sometime signal stream is not good. When signal stream is more than normal bandwidth then algorithm detect background as foreground and when signal stream is lesser than normal bandwidth algorithm detect foreground as background. While this new technique may be some computationally complex but it gives good result in moving object extraction in all situation. This technique can be useful in traffic surveillance systems, motion tracking, traffic counting, and many more.

8. FUTURE WORK

A better thresholding algorithm may be used to eliminate sporadic detection of motion where there is no motion. This appears to happen when the gray level values are very high. Future work will include generating KFDA matrix per each block of the video sequence and processing images with variation in size of the blocks.

9. REFERENCES

- [1] K Toyama, J Krumm, B Brumitt, and B Meyers. Wallflower: principles and practice of background maintenance. Proceedings of the Seventh IEEE International Conference on Computer Vision, 1(c):255–261, 1999.
- [2] L. Maddalena and A. Petrosino. A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications. IEEE Transactions on Image Processing, 17(7):1168–1177, 2008.
- [3] http://en.wikipedia.org/wiki/Principal_components_analysis
- [4] A. Manzanera and J. Richefeu, “A robust and computationally efficient motion detection algorithm based on Σ - Δ background estimation,” in Proceedings. Indian Conference Computing Vis., pp. 46–51, December 2004.
- [5] José Manuel Milla, Sergio L. Toral, Manuel Vargas and Federico Barrero, “Computer Vision Techniques for Background Modelling in Urban Traffic Monitoring,” University of Seville Spain.
- [6] Vrunda A. Mahamuni, Madhuri Khambete, “Background Subtraction Techniques for Moving Object Detection in Video Frames,” in International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-3, Issue-1, October 2013.
- [7] Rupali Patharkar, Sonali Bodkhe, Dr. A.R Mahajan, “Background Detection by Two Way Technique,” in International Journal of Computer Science and Information Technologies, Vol. 4 (3) pp. 289 – 291, 2013.
- [8] Xiaoqiang Zhao & Shiyuan Li, “A Modified Kernel Fisher Discriminant Analysis Algorithm for Fault Diagnosis,” in International Journal of Advanced Computer Science, Vol. 2, No. 1, Pp. 33-36, January. 2012.
- [9] Shih-Chia Huang and Ben-Hsiang Do “Radial Basis Function Based Neural Network for Motion Detection in Dynamic Scenes,” in IEEE Transactions on Cybernetics, vol. 44, no. 1, January 2014.
- [10] Shweta Lawanya Rao and Dolley Shukla, “Image Segmentation Using a RBF Approach of Neural Network,” in International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, Volume 4, Issue 4, April 2014.
- [11] Shih-Chia Huang and Bo-Hao Chen, “An Advanced Moving Object Detection Algorithm for Automatic Traffic Monitoring in Real-World Limited Bandwidth Networks,” in IEEE Transactions on Multimedia, vol. 16, no. 3, April 2014.
- [12] S. C. Huang, “An advanced motion detection algorithm with video quality analysis for video surveillance systems,” in IEEE Transactions Circuits System Video Technology, vol. 21, no. 1, pp. 1–14, January 2011.
- [13] Pankaj Kumar, Member, IEEE, Surendra Ranganath, Huang Weimin, and Kuntal Sengupta, “Framework for Real-Time Behavior Interpretation From Traffic Video,” in IEEE Transactions on Intelligent Transportation Systems, vol. 6, no. 1, March 2005.
- [14] B. H. Dou and S. C. Huang, “Dynamic background modelling based on radial basis function neural networks for moving object detection,” in Proceedings IEEE International Conference Multimedia Expo, pp. 1–4, July 2011.
- [15] W. Hu, T. Tan, L. Wang, and S. Maybank, “A survey on visual surveillance of object motion and behaviors,” in IEEE Transactions System, Man, Cybernetics C, Applied Rev., vol. 34, no. 3, pp. 334–352, August 2004.